

# **Development of High-Order CENO Finite-Volume Schemes with Block-Based Adaptive Mesh Refinement**

Doctor of Philosophy

Lucian Ivan

A thesis submitted in conformity with the requirements  
for the degree of Doctor of Philosophy  
Graduate Department of Aerospace Science and Engineering  
University of Toronto

Copyright © 2010 by Lucian Ivan

# Abstract

## Development of High-Order CENO Finite-Volume Schemes with Block-Based Adaptive Mesh Refinement

Lucian Ivan

<lucian.ivan@utoronto.ca>

Doctor of Philosophy

Graduate Department of Aerospace Science and Engineering

University of Toronto

2010

A high-order central essentially non-oscillatory (CENO) finite-volume scheme in combination with a block-based adaptive mesh refinement (AMR) algorithm is proposed for solution of hyperbolic and elliptic systems of conservation laws on body-fitted multi-block mesh. The spatial discretization of the hyperbolic (inviscid) terms is based on a hybrid solution reconstruction procedure that combines an unlimited high-order  $k$ -exact least-squares reconstruction technique following from a fixed central stencil with a monotonicity preserving limited piecewise linear reconstruction algorithm. The limited reconstruction is applied to computational cells with under-resolved solution content and the unlimited  $k$ -exact reconstruction procedure is used for cells in which the solution is fully resolved. Switching in the hybrid procedure is determined by a solution smoothness indicator. The hybrid approach avoids the complexity associated with other ENO schemes that require reconstruction on multiple stencils and therefore, would seem very well suited for extension to unstructured meshes. The high-order elliptic (viscous) fluxes are computed based on a  $k$ -order accurate average gradient derived from a  $(k+1)$ -order accurate reconstruction. A novel  $h$ -refinement criterion based on the solution smoothness indicator is used to direct the steady and unsteady refinement of the AMR mesh. The predictive capabilities of the proposed high-order AMR scheme are demonstrated for the Euler and Navier-Stokes equations governing two-dimensional compressible gaseous flows as well as for advection-diffusion problems characterized by the full range of Péclet numbers,  $Pe$ . The ability of the scheme to accurately represent solutions with smooth extrema and yet robustly handle under-resolved and/or non-smooth solution content (i.e., shocks and other discontinuities) is shown for a range of problems. Moreover, the ability to perform mesh refinement in regions of smooth but under-resolved and/or non-smooth solution content to achieve the desired resolution is also demonstrated.

# Abstract

## Dezvoltarea Metodei Volumelor Finite CENO de Ordin Superior cu Adaptarea Discretizării Bazată pe Blocuri

Lucian Ivan

<lucian.ivan@utoronto.ca>

Doctor în Filosofie

Graduate Department of Aerospace Science and Engineering

University of Toronto

2010

O schemă numerică de ordin superior, centrată și esențial neoscilantă (CENO) de tipul volumelor finite, în combinație cu un algoritm de adaptare a rețelei de discretizare bazat pe blocuri (AMR) este propusă pentru obținerea soluțiilor de sisteme hiperbolice și eliptice ce provin din principii de conservare, pe rețele de discretizare multibloc ce urmăresc frontierele domeniului de calcul. Discretizarea spațială a termenilor de tip hiperbolic (nevâscos) este bazată pe o procedură hibridă de reconstrucție a soluției. Astfel, o reconstrucție nelimitată de ordin superior de tipul  $k$ -exact în sensul celor mai mici pătrate, care este derivată pe baza unui șablon centrat și fixat de puncte, este combinată cu un algoritm de reconstrucție pe porțiuni care este linear și limitat, și care păstrează monotonia soluției. Reconstrucția limitată este aplicată în celulele de calcul în care conținutul soluției nu este total rezolvat, iar procedura de reconstrucție nelimitată  $k$ -exact este utilizată pentru celulele în care soluția este complet reprodusă. Alegerea metodei în procedura hibridă este determinată automat de către un indicator ce caracterizează netezimea soluției. Abordarea hibridă evită complexitățile asociate cu alte scheme ENO care necesită reconstrucția pe multiple șabloane de puncte și, de aceea, pare foarte indicată pentru extinderea la rețele de discretizare nestructurate. Fluxurile eliptice (viscoase) de ordin superior sunt calculate pe baza unui gradient de ordin  $k$  obținut printr-o medie aritmetică și derivat din reconstrucții cu ordin de acuratețe  $(k+1)$ . Un nou criteriu pentru rafinarea  $h$  a rețelei, derivat pe baza indicatorului de netezime al soluției, este utilizat pentru a direcționa adaptarea rețelei de discretizare pentru cazuri staționare și nestaționare. Capabilitățile de precizie ale schemei propuse de ordin superior și cu rafinarea rețelei de calcul sunt demonstrate pentru ecuațiile Euler și Navier-Stokes care guvernează curgerea gazelor compresibile în două dimensiuni precum și pentru probleme de convecție-difuzie caracterizate de un întreg domeniu de numere Péclet,  $Pe$ . Abilitatea schemei de a

reprezenta soluții cu extreme netede și, în același timp, de a manipula în mod robust soluții cu conținut nerezolvat și/sau discontinuu (adică, unde de șoc și alte discontinuități) este demonstrată pentru o gamă largă de probleme. Mai mult decât atât, abilitatea metodei de a rafina rețeaua de discretizare pentru atingerea rezoluției dorite în regiuni în care conținutul soluției este fie neted dar nerezolvat, fie discontinuu este dovedită.

*To my beloved parents,  
Loving children and wife,  
And dear sisters.*

# Acknowledgements

I would like to thank a number of people without whom I would not have been able to complete this thesis research.

I feel great honour to express my deepest sense of appreciation and gratitude to my thesis supervisor Prof. C.P.T. Groth for his inspiring dedication, teaching passion, skillful suggestions and keen interest in the project over the whole period of my studies. His encouragement and support have provided me with the opportunity to explore and deeply appreciate the amazing world of computational fluid dynamics. I am sure that his unforgettable attitude and dedicated time will prove valuable sources of inspiration throughout my life.

I am thankful to the other members of my doctoral examination committee, Prof. D.W. Zingg and Prof. Ö. L. Gülder, for their assistance, guidance and invaluable suggestions provided at all levels of this research project. I also wish to thank Prof. Z. J. Wang and Prof. M. Bussmann, who served as the external and internal thesis examiners, respectively, for their extensive comments and suggestions which contributed significantly to the quality of the final document. Many thanks go to Prof. H. De Sterck for his support, patience and understanding during my extended transition to postdoctoral studies.

I would also like to thank my colleagues at UTIAS for their invaluable help and support throughout this work. I wish to acknowledge and thank my CFD lab fellows Jai Sachdev, James McDonald, Scott Northrup, Eric Li, Francisco H. Perez, Markus Rumpfkeil and Jason Hicken for enlightening discussions, suggested solutions and memorable friendship. An extended thank-you goes to my very good friend James who was always ready to help and to Scott, the computer guru, for his invaluable assistance on any computer issues. I am also very thankful to Eric Li and Kalvin Tsang for their help with my settlement in Canada.

It is a great pleasure to pay my sincerest thanks to the people in Romania who provided knowledge, guidance and inspiration for pursuing graduate studies overseas. In particular, I wish to acknowledge Prof. M. Buruiană, Prof. S. Mitran and Mrs. and Dr. V. Silivestru.

I have no words to describe the respect and thankfulness to my parents, Steluta and Ilie Ivan, and sisters, Mihaela and Daniela Ivan. They have been ardent proponents of my endeavours, and for that I am forever grateful. Last, but certainly not least, I owe my profoundest thanks to my loving wife, Cristina, and dearest children, Isabella and Eduard, for their great sacrifice, patience, inspiration and understanding during this whole endeavour.

Finally, I would like to acknowledge the financial support from Ontario Graduate Scholarship (OGS) and other sources.

Computational resources for performing all of the calculations reported herein were provided by the SciNet High Performance Computing Consortium at the University of Toronto and Compute/Calcul Canada through funding from the Canada Foundation for Innovation (CFI) and the Province of Ontario, Canada. The author is very thankful for access to these computational resources.

LUCIAN IVAN  
University of Toronto Institute for Aerospace Studies  
October 12, 2010

# CONTENTS

- Abstract** **ii**
  
- List of Tables** **xi**
  
- List of Figures** **xii**
  
- List of Symbols** **xix**
  
- 1 Introduction** **1**
  - 1.1 Motivation: Enable LES for Practical Combustion Devices . . . . . 1
  - 1.2 Thesis Objectives . . . . . 6
  - 1.3 Background . . . . . 7
    - 1.3.1 High-Order Methods . . . . . 7
    - 1.3.2 Solution Adaptivity . . . . . 20
  - 1.4 Overview of Dissertation . . . . . 25
  
- 2 Conservation Equations of Interest** **27**
  - 2.1 Advection-Diffusion Equation . . . . . 27
  - 2.2 Euler Equations . . . . . 30
  - 2.3 Navier-Stokes Equations . . . . . 32
  
- 3 High-Order CENO Finite-Volume Scheme** **35**
  - 3.1 Finite-Volume Method for Conservation Equations . . . . . 36
  - 3.2 Body-Fitted Quadrilateral Grids . . . . . 38
  - 3.3 Semi-Discrete Form . . . . . 40
  - 3.4 Explicit Temporal Discretization Methods . . . . . 41

3.5	Numerical Residual Evaluation . . . . .	44
3.6	CENO Reconstruction . . . . .	48
3.6.1	$k$ -Exact Least-Squares Reconstruction . . . . .	51
3.6.2	Monotonicity Enforcement . . . . .	58
3.6.3	Reconstruction at Boundaries and Implementation of High-Order Boundary Conditions . . . . .	66
3.7	Inviscid (Hyperbolic) Flux Evaluation . . . . .	74
3.8	Viscous (Elliptic) Flux Evaluation . . . . .	75
<b>4</b>	<b>High-Order CENO with Parallel Adaptive Mesh Refinement</b>	<b>77</b>
4.1	Parallel Block-Based AMR Algorithm . . . . .	77
4.2	CENO Scheme on Body-Fitted Multi-Block Mesh . . . . .	80
4.3	High-Order Solution Transfer on AMR Grids . . . . .	84
4.4	Refinement Criteria Based on High-Order Solution . . . . .	86
4.5	Domain Decomposition and Parallel Implementation . . . . .	87
4.6	Computational Resources Used for Numerical Simulations . . . . .	89
<b>5</b>	<b>Numerical Results</b>	<b>90</b>
5.1	Assessment of Properties of CENO Discretization . . . . .	90
5.2	Function Reconstruction on Fixed Mesh . . . . .	93
5.2.1	One-Dimensional Smooth Trigonometric Function . . . . .	93
5.2.2	One-Dimensional Smooth Exponential Function with Large Gradients	96
5.2.3	One-Dimensional Discontinuous Step Function . . . . .	98
5.2.4	Two-Dimensional Smooth Trigonometric Function . . . . .	99
5.2.5	Two-Dimensional Abgrall's Function . . . . .	102
5.3	Function Reconstruction with AMR . . . . .	103
5.4	Solution of the Advection-Diffusion Equation on Fixed Mesh . . . . .	107
5.4.1	Circular Advection of Smooth Inflow Variation . . . . .	107
5.4.2	Circular Advection of Discontinuous Inflow Variation . . . . .	107
5.4.3	Numerical Investigation of Elliptic Operator Discretization . . . . .	107
5.4.4	Residual-Based Accuracy Assessment for Poisson Equation . . . . .	110
5.4.5	Heat Conduction Equation with Source Term . . . . .	112
5.4.6	Solution of the Laplace Equation on Curved Boundaries . . . . .	112

5.4.7	Two-Dimensional Channel Flow Problem . . . . .	114
5.5	AMR Solution of the Advection-Diffusion Equation . . . . .	116
5.6	Solution of the Euler Equations on Fixed Mesh . . . . .	118
5.6.1	One-Dimensional Blastwave Interaction Problem . . . . .	118
5.6.2	Sod Shock-Tube Problem . . . . .	119
5.6.3	Ringleb Flow . . . . .	121
5.6.4	Subsonic Flow Past Circular Cylinder . . . . .	124
5.6.5	Supersonic Flow Past Circular Cylinder . . . . .	125
5.7	Solution of the Euler Equations with AMR . . . . .	126
5.7.1	Supersonic Flow Past Circular Cylinder . . . . .	126
5.7.2	Smooth Translation of Density Variation . . . . .	128
5.8	Solution of the Navier-Stokes Equations on Fixed Mesh . . . . .	130
5.8.1	Laminar Flat Plate . . . . .	130
5.8.2	Steady Laminar Subsonic Flow Past Circular Cylinder . . . . .	131
5.8.3	Unsteady Laminar Subsonic Flow Past Circular Cylinder . . . . .	135
5.9	Solution of the Navier-Stokes Equations with AMR . . . . .	137
5.9.1	Unsteady Laminar Subsonic Flow Past Circular Cylinder . . . . .	137
5.10	Computational Cost of High-Order Scheme . . . . .	144
<b>6</b>	<b>Conclusions and Recommendations</b>	<b>148</b>
6.1	Summary and Conclusions . . . . .	148
6.2	Recommendations for Future Research . . . . .	151
	<b>References</b>	<b>155</b>
	<b>APPENDICES</b>	<b>173</b>
	<b>A Properties of Quadrilateral Elements and Function Integration</b>	<b>174</b>
	<b>B Calculation of Geometric Moments</b>	<b>181</b>

# LIST OF TABLES

3.1	<i>Examples of condition numbers of the coefficient matrix of the least-squares reconstruction problem . . . . .</i>	57
3.2	<i>General form of the left-hand-side matrix, <math>A</math>, corresponding to a constrained least-squares reconstruction in which two variables are coupled. . . . .</i>	72
5.1	<i>The slope values corresponding to each convergence error plot shown in Fig. 5.16. . . . .</i>	111
5.2	<i>Error norms, simulation CPU time and memory requirements associated with the unlimited and limited 2nd-order, and the 4th-order CENO schemes obtained for the periodic linear convection of the density variation shown in Fig. 5.44. . . . .</i>	147
A.1	<i>Analytical expression for computing the geometric moment of powers <math>(2, 1)</math>, <math>(\overline{x^2y^1})_I</math>, function of the coordinates of the quadrilateral vertices. . . . .</i>	177

# LIST OF FIGURES

1.1	<i>Time evolution of local temperature in a turbulent flame as captured by DNS, LES and RANS approaches.</i>	4
1.2	<i>Grid points used in the discrete representation of the numerical solution.</i>	8
1.3	<i>Illustration of high-order <math>k</math>-exact approximations</i>	12
1.4	<i>Illustration of Gibbs-like phenomenon near solution discontinuity.</i>	13
1.5	<i>Plots of estimated computational cost vs. error in one-space dimension for several shock-capturing schemes</i>	17
1.6	<i>Illustration of block-based mesh adaptation on a body-fitted grid.</i>	23
3.1	<i>Illustration of body-fitted quadrilateral structured grids.</i>	39
3.2	<i>Schematic representation of quadrilateral element types.</i>	40
3.3	<i>Illustration of <math>k</math>-exact reconstruction for smooth analytic functions represented on resolved and under-resolved grids.</i>	62
3.4	<i>The graph of <math>f(\alpha) = \frac{\alpha}{(1-\alpha)}</math>.</i>	64
3.5	<i>Representation of geometric elements near boundaries for a quadrilateral mesh.</i>	67
4.1	<i>Adaptive mesh refinement quad-tree data structure and associated adaptive blocks for a quadrilateral body-fitted mesh</i>	79
4.2	<i>Illustration of refined multi-block grid and various types of inter-block communication</i>	80
4.3	<i>Comparison between the low- and high-order representations of a multi-block computational domain</i>	82
4.4	<i>Representation of the valid combinations of boundary types for the SW block corner.</i>	83

4.5	<i>Depiction of a coarse cell division into four fine cells for low- and high-order block boundary elements.</i>	84
4.6	<i>Block distribution among the processors of a multi-processor architecture.</i>	88
5.1	<i>Piecewise cubic (<math>k = 3</math>) hybrid CENO reconstruction of the smooth trigonometric function <math>f(x) = \sin(2x) + 2 \cos(x)</math></i>	94
5.2	<i><math>L_1</math>, <math>L_2</math>, and <math>L_\infty</math> norms of the reconstruction errors for ENO and hybrid CENO reconstructions of the function <math>f(x) = \sin(2x) + 2 \cos(x)</math></i>	95
5.3	<i>Reconstruction of <math>f(x) = e^{-4x} \sin(5x)</math> on cell-centred one-dimensional uniform mesh for <math>-5 &lt; x &lt; 5</math> containing 64 and 100 computational cells</i>	97
5.4	<i><math>L_1</math> and <math>L_\infty</math> norms of the reconstruction errors for ENO and hybrid CENO reconstructions of the function shown in Fig. 5.3</i>	97
5.5	<i>Reconstruction of a discontinuous step function on a one-dimensional uniform mesh for <math>-1 &lt; x &lt; 4</math> containing 11 computational cells</i>	98
5.6	<i>Exact solution of function <math>u(x, y) = 1.1 + \cos(\pi x^2 + 4\pi y)</math> and the quartic solution reconstruction on a regular mesh with <math>40 \times 40</math> computational cells.</i>	99
5.7	<i>The regular and disturbed body-fitted meshes with <math>40 \times 40</math> cells used in the convergence study of the function depicted in Fig. 5.6.</i>	100
5.8	<i>Convergence history for <math>L_1</math>, <math>L_2</math>, and <math>L_\infty</math>-norms of the reconstruction error for the smooth trigonometric function <math>u(x, y) = 1.1 + \cos(\pi x^2 + 4\pi y)</math></i>	101
5.9	<i>Contours of the exact and the fourth-order CENO reconstructed solutions for the non-smooth two-dimensional function of Abgrall</i>	102
5.10	<i>Cross sections of the CENO reconstructed solutions of the Abgrall function for different orders</i>	103
5.11	<i><math>L_1</math> norm of the reconstruction error versus the number of computational cells for Abgrall's function</i>	104
5.12	<i>Contour plots of a third-order CENO reconstruction solution of the Abgrall's function on AMR meshes</i>	105
5.13	<i>Sequence of AMR meshes generated by the block-based refinement algorithm for the refinement of the Abgrall's function</i>	106
5.14	<i>Predicted 4th-order CENO solution and error convergence history for the circular advection of a smooth inflow variation</i>	108

5.15	<i>Numerical and exact solution plots for the circular advection of top-hat inflow variation . . . . .</i>	108
5.16	<i><math>L_1</math>, <math>L_2</math> and <math>L_\infty</math> norms of the residual error obtained for Poisson equation <math>\vec{\nabla} \cdot (\vec{\nabla}u) = a e^{\beta u}</math> . . . . .</i>	111
5.17	<i>Fourth-order CENO solution to the heat conduction equation . . . . .</i>	113
5.18	<i>Error norms for the problem defined in Fig. 5.17 obtained with different reconstruction orders . . . . .</i>	113
5.19	<i>Fourth-order solution to the Laplace equation and associated error norms for cubic and quartic polynomial interpolants . . . . .</i>	114
5.20	<i>Numerical CENO prediction of channel flow problem and convergence error history for different Péclet numbers . . . . .</i>	115
5.21	<i>Contour plots and sequence of refined meshes for CENO prediction to circular advection of a smooth and discontinuous inflow . . . . .</i>	117
5.22	<i>Numerical solutions of the one-dimensional blast wave interaction problem . . . . .</i>	118
5.23	<i>Comparison between the numerical prediction of the 4th-order CENO with different mesh resolutions and the exact solution for Sod's shock tube problem . . . . .</i>	119
5.24	<i>Predicted density profiles in the regions defined in Fig. 5.23 obtained with a 4th-order ENO, a 4th-order CENO and a 2nd-order TVD scheme on different mesh resolutions. . . . .</i>	120
5.25	<i>Predicted Mach number and error norms for subsonic Ringleb's flow . . . . .</i>	121
5.26	<i>Predicted Mach number and error norms for transonic Ringleb's flow . . . . .</i>	123
5.27	<i>Predicted Mach number distributions and error norms in entropy for <math>M=0.38</math> subsonic flow past a circular cylinder . . . . .</i>	124
5.28	<i>Predicted Mach number distribution for <math>M=1.5</math> flow past a circular cylinder obtained using the 4th-order CENO scheme on a <math>160 \times 160</math> grid . . . . .</i>	125
5.29	<i>Numerical results for <math>M=2.1</math> flow past a circular cylinder obtained using the 4th-order CENO scheme on the final refined AMR mesh . . . . .</i>	126
5.30	<i>Sequence of adaptively refined AMR mesh for supersonic flow past a circular cylinder . . . . .</i>	127
5.31	<i>Representation of initial and final solutions for the smooth translation of density variation with unsteady AMR . . . . .</i>	128
5.32	<i>Computed error norms for the smooth translation of density variation problem . . . . .</i>	129

5.33	<i>Variation of the non-dimensional velocity components at <math>Re_x = 8,000</math> and distribution of skin friction coefficient for a laminar flat plate problem . . . .</i>	131
5.34	<i>Prediction of Mach number and streamlines for laminar flow over cylinder at <math>M_\infty=0.1</math> and <math>Re=30</math> on a mesh with 51,200 cells. . . . .</i>	132
5.35	<i>Comparison of drag coefficients predicted by the 4th-order CENO and the 2nd-order scheme for the laminar flow over cylinder at <math>M_\infty=0.1</math> and <math>Re=30</math> . .</i>	133
5.36	<i>The parallel speedup and efficiency achieved for the steady laminar flow past cylinder by the 4th-order CENO algorithm as a function of the number of computing cores. . . . .</i>	135
5.37	<i>Entropy prediction and streamlines for vortex shedding flow past circular cylinder at <math>Re=110</math> obtained using the 4th-order (<math>k=4</math>) CENO scheme . . . . .</i>	136
5.38	<i>Time dependent variation of the lift and drag coefficients for the flow past circular cylinder at <math>Re=110</math> obtained using the 4th-order CENO scheme . .</i>	136
5.39	<i>Entropy field prediction and density contour lines for vortex shedding flow past circular cylinder at <math>Re=110</math> obtained on the initial AMR mesh . . . . .</i>	138
5.40	<i>Variation of the drag coefficient for unsteady vortex shedding flow past the circular cylinder as a function of physical time obtained using the 4th-order CENO scheme in conjunction with the block-based AMR algorithm . . . . .</i>	139
5.41	<i>Predicted entropy and density fields for unsteady vortex shedding obtained on the first five AMR meshes at various time instances . . . . .</i>	141
5.42	<i>Comparison between the drag coefficients predicted by the 4th-order CENO scheme and previously reported values . . . . .</i>	142
5.43	<i>Predicted flow fields for vortex shedding over the cylinder on a sequence of dynamically refined meshes . . . . .</i>	143
5.44	<i>Exact and predicted 4th-order CENO solution for the periodic linear advection of the density variation <math>\rho(x, y) = 1.0 + 0.5 \cos(\pi \mathcal{J}(x)) \sin(5 \pi \mathcal{J}(x))</math> . . . .</i>	145

# LIST OF SYMBOLS

## Alphanumeric Symbols

$\mathcal{A}$	the domain of a two-dimensional control volume
$A$	cell area
$a$	local speed of sound
2D	two-dimensional
$C_p$	gas specific heat at constant pressure
$C_v$	gas specific heat at constant volume
$D_{p_1 p_2}^k$	polynomial coefficient of powers $(p_1, p_2)$ in the $k$ -exact polynomial approximation
$E$	total energy
$e$	specific total energy
$\mathbf{F}$	$x$ inviscid flux in vector notation
$\vec{\mathbf{F}}$	flux dyad
$\vec{\mathbf{F}}_E$	elliptic flux vector
$\vec{\mathbf{F}}_H$	hyperbolic flux vector
$\mathbf{G}$	$y$ inviscid flux in vector notation
$\hat{i}, \hat{j}$	Cartesian unit vectors
$\vec{I}$	identity tensor
$I$	index denoting an element pertained to the cell in which the reconstruction is performed

$i, j$	ordered pair of indices to identify a computational cell in a two-dimensional multi-block mesh
$J$	index denoting an element pertained to any neighbour in the reconstruction supporting stencil
$k$	polynomial order
$m_x, m_y$	$x$ - and $y$ -components of linear momentum
$\mathcal{N}_D$	number of unknown coefficients in the $k$ -exact polynomial approximation
$\vec{n}$	unit outward vector normal
$N_f$	number of faces of a control volume
$p$	gas pressure
$p_1, p_2$	indices used in the two-dimensional $k$ -exact polynomial approximation
$\vec{q}$	molecular heat flux vector
$q_x, q_y$	$x$ - and $y$ -components of molecular heat flux
$\mathbf{R}$	spatial residual vector
Re	Reynolds number
$R$	gas constant
$\mathbf{S}$	source term vector
$\vec{\vec{S}}$	strain rate tensor
$T$	gas temperature
$t$	time
$\mathbf{U}$	conserved solution variables in vector notation
$u$	scalar solution variable
$u_{i,j}^k$	$k$ -exact polynomial solution approximation for a cell $(i, j)$
$\mathcal{V}$	control volume
$\vec{V}$	velocity vector
$v_x, v_y$	$x$ - and $y$ -components of velocity

$\mathbf{W}$	primitive solution variables in vector notation
$\vec{X}_{i,j}$	centroid of the cell $(i, j)$ of a two-dimensional multi-block mesh
$(\overline{x^{p_1}y^{p_2}})_{CV}$	geometric moment of powers $(p_1, p_2)$ of a generic control volume, CV, about its own centroid
$(\widetilde{x^{p_1}y^{p_2}})_{IJ}$	geometric moment of powers $(p_1, p_2)$ of the neighbour cell $J$ about the centroid of the reconstructed cell, $I$
$x, y$	Cartesian coordinates

## Greek Symbols

$\Delta\ell$	length of the cell face
$\Delta$	LES filter size
$\gamma$	gas specific heat ratio
$\kappa$	generic diffusion coefficient in the advection-diffusion equation or fluid thermal conductivity in the Navier-Stokes equations
$\mu$	gas molecular viscosity
$\nu$	gas kinematic viscosity
$\Omega$	closed surface of the control volume
$\omega$	Gaussian-quadrature weighting coefficient
$\phi$	non-linear source term (a scalar function)
$\rho$	gas density
$\tau_{xx}, \tau_{yy}, \tau_{xy}$	two-dimensional components of the molecular fluid stress tensor
$\vec{\tau}$	molecular fluid stress tensor

## Abbreviations

AMR	Adaptive Mesh Refinement
BC(s)	Boundary Condition(s)
CENO	Central Essentially Non-Oscillatory
CFD	Computational Fluid Dynamics

CFL	Courant-Friedrichs-Lewy Number
DNS	Direct Numerical Simulation
DOF	Degrees of Freedom
ENO	Essentially Non-Oscillatory
FEM	Finite Element Method
FV(M)	Finite Volume (Method)
HPC	High-Performance Computing
IC(s)	Initial Condition(s)
LES	Large Eddy Simulation
LHS	Left-Hand-Side
ODE	Ordinary Differential Equation
PDE(s)	Partial Differential Equation(s)
RANS	Reynolds Average Navier-Stokes
RHS	Right-Hand-Side
SGS	Subgrid-Scale
TFI	Transfinite Interpolation
TV	Total Variation
TVB	Total Variation Bounding
TVD	Total Variation Diminishing

# Chapter 1

## INTRODUCTION

*“Nothing ever built ... arose to touch the skies unless some man dreamed that it should, some man believed that it could, and some man willed that it must.”*

by Charles F. Kettering (1876-1958), American engineer & inventor

Since the dawn of *computational fluid dynamics* (CFD) and the landmark established by the earliest numerical solution for flow past a cylinder carried out by Thom in 1933 [1], many great minds have been contributing their efforts to transforming the discipline of CFD into the modern research and design tool that currently impacts and cuts across *all* technical fields where the flow of a fluid is important. Indeed, current state-of-the-art CFD techniques are nowadays employed successfully for prediction of flow fields with complex physical phenomena and sophisticated geometries. Undoubtedly, these magnificent achievements in CFD have been nourished by the tremendous progress in high-performance computing (HPC) hardware over the years, and most notably by the introduction of *vector* and *parallel* processing in the last three to four decades. The present availability of such impressive computational power has opened up new horizons in the complexity and magnitude of scientific problems studied with CFD methods and has set new frontiers for research in improved CFD algorithms.

### **1.1 Motivation: Enable Large Eddy Simulation for Practical Combustion Devices**

In recent years, the accurate prediction of the type of flows encountered in practical combustion devices has become of prime importance to the engineering and scientific community due to the environmental and health issues posed by pollutants released from fossil fuel burning combustion devices and to the soaring price of oil-based fuels determined by global decrease in the availability of so-called “easy oil” [2]. Technical combustion devices are characterized

by very complicated flow patterns, such as swirling flows, breakdowns of large-scale vortical structures and recirculation regions, the accurate prediction of which is still a challenging problem. Therefore, the development of advanced combustion capabilities that allow the design of the next generation of combustion systems, having ultra-low nitrogen oxides ( $\text{NO}_x$ ) production, substantial thermal efficiency improvements, as well as significant reduction in carbon dioxide ( $\text{CO}_2$ ) emissions, is of current interest to the combustion community.

One avenue for studying combustion is offered by numerical simulations and, with increasing HPC resources, this approach is being considered more and more often. Indeed, rapid improvements in commodity computing hardware and the recent development of distributed-memory multi-processor clusters for HPC have dramatically extended the application spectrum of computational methods and improved the accuracy with which physically complex flows characterized by a wide range of spatial and temporal scales can be currently predicted. Thus, traditionally computationally expensive numerical simulations such as those of turbulent non-reacting and reacting flows, and numerical computation of complex unsteady aerodynamic flows, have been performed for realistic, complex, three-dimensional (3D) geometries with discretizations of millions of grid points. For turbulent flows, not only has this enhanced HPC hardware allowed the use of increased resolution for flow simulations with classical Reynolds Averaged Navier-Stokes models [3], but it has also made more routine the study of more sophisticated modelling techniques, such as large eddy simulations of turbulence for both scientific interest and practical engineering applications [4].

The numerical prediction of combustion processes is challenging as a wide variety of complex physical phenomena are involved, including multi-component and multi-phase transport phenomena, complex finite-rate chemical reactions, radiative heat transfer and turbulence. The presence of turbulence in a flow, even in a non-reacting one, represents a big increase in the complexity of the problem and makes the study of the reacting flows one of the most challenging fields of engineering science. The spectrum of approaches for turbulence representation is dominated by three main categories, *Direct Numerical Simulation (DNS)*, *Large Eddy Simulation (LES)* and *Reynolds Average Navier-Stokes (RANS) Modelling*. In the case of turbulent reacting flows additional modelling is required for each approach to account for the interaction between chemistry and turbulence. There have been a number of computational studies involving DNS that attempt to resolve detailed chemical kinetics and transport phenomena [5, 6, 7], but they have been restricted to very small ( $\mathcal{O}(1)$  cm) ideal-

ized configurations and to short simulation times. The huge discrepancy between the scale of the realistic problems and those that can be computed using DNS presently and in the foreseen near future generally restrict the application of this method to academic research, where it can produce valuable information and insight that is not available by using other techniques.

For many flows of engineering interest, the classical RANS approaches have proved to be very successful, but there are some applications for which a difference of several orders of magnitude between the unsteadiness of the flow and the turbulent fluctuations does not exist. For flows for which a clear separation of scales cannot be established, a sufficiently complex representation of the turbulent quantities must be formulated, in order to better represent the wide range of turbulent scales. At the same time, the representation must have a reduced computational cost as compared with DNS so as to be applicable to practical engineering problems.

The LES formulation represents a very promising tool in which the large turbulent scale of motions, that carry most of the turbulent kinetic energy and depend on the geometry of the system, are completely resolved, while the effect of the small scale fluctuations is introduced by turbulence closure models. The underlying premise is that the small-scale turbulence is weaker, contributing less to the turbulent energy, and is therefore less critical. Also, it is more isotropic and has a more universal structure. It is thus more amenable to modelling. Figure 1.1 shows schematically the typical differences between the results obtained using the three aforementioned turbulence representations. While DNS will provide the full unsteady solution of the problem, RANS approximation can at most represent only the time averaged solution. Since LES is a technique intermediate between DNS and RANS, the unsteady character of the problem can be represented, but the fidelity of the solution in space and time will be affected.

In LES, the turbulent fields are separated into large-scale resolved and small-scale unresolved contributions. A low pass spatial filtering operation applied to the instantaneous turbulent fields removes turbulent motions of length scales smaller than the filter size,  $\Delta$ . The governing equations for the remaining large-scale field are amenable to discretization using a mesh with grid spacing of order  $\Delta$  or smaller. Since the closure terms accounting for turbulence/chemistry interactions are modelled, a very dynamic research area in LES is represented by the study of different subgrid-scale (SGS) models. The filtering operation can

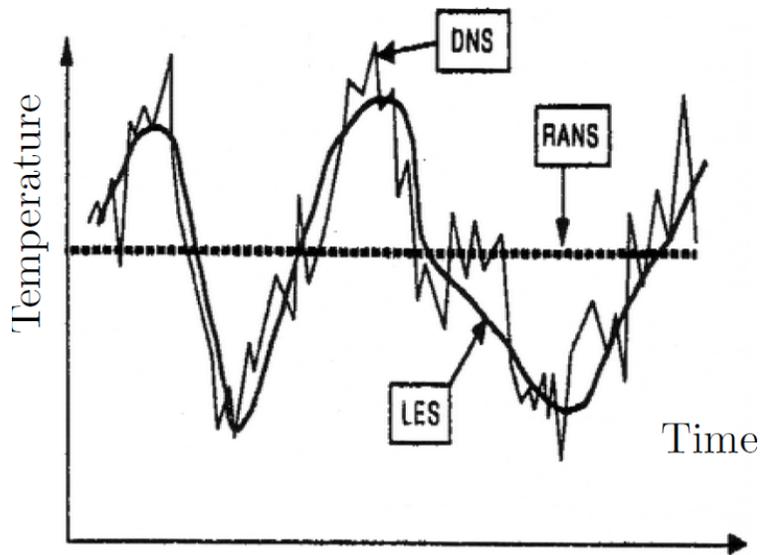


Figure 1.1: *Time evolution of local temperature in a turbulent flame as captured by DNS, LES and RANS approaches.*

be done implicitly or explicitly. Implicit filtering does not require explicit definition of the spatial filter. Instead, the procedure exploits the discrete nature of any numerical solution method and the filtering is due to the representation of the solution on a finite mesh. As such, the filter size,  $\Delta$ , is generally directly related to the grid spacing. It follows that the smallest resolved scales are actually under-resolved and any attempts to overcome this lack of accuracy by employing smaller mesh sizes have also the negative effect of changing the initial LES problem (i.e., new turbulent scales are added to the resolved field). Because of numerical dissipation, the energy content of the smallest resolved scales is under-predicted which also leads to under-predictions from the small-scale models. Therefore, a better alternative is to use an explicit filtering operation. Such an approach has the advantages of making sure that all the terms involved in the governing equations are numerically represented with the same order of accuracy (i.e., no aliasing errors), that the LES solution is the same, regardless of the mesh size, and that no commutation errors between the filtering operation and the spatial derivative of solution content are introduced at transition regions of different filter width. Explicit filtering and its combination with different discretization methods are currently actively researched [8].

Unfortunately, for the current computational capabilities the number of computational

cells required to accurately represent the smallest resolved scales with a low-order scheme (i.e., first or second order) may turn out to be prohibitive, especially in the wall layers where the size of the resolved eddies varies with the distance from the wall. This is especially true for upwind finite-volume schemes in which numerical dissipation is added automatically for stability and robustness purposes through the solution of a Riemann problem that is solved at every interface. While such numerical schemes allow for the robustness of the solver near solution discontinuities and for an easier implementation of the boundary conditions (BCs), they might also introduce numerical dissipation that is larger than the contribution from the SGS model [9]. Therefore, the investigation of improved solution algorithms, such as the use of higher-order spatial discretizations to make sure the numerical dissipation associated with the resolved large scales is smaller than the truncation error of the SGS models, represents an open research area. Many researchers already consider high-order numerical schemes a key feature for obtaining precise LES results [10].

Undoubtedly, the high solution-accuracy requirements of LES techniques have been a driving force for the development of high-order methods which have the potential to capture fine solution details with a lower mesh resolution and at a reduced total computational cost than the standard lower-order spatial discretizations (i.e, methods up to second order). However, the application of a high-order scheme to turbulent combustion simulation can be challenging due to large gradients (or discontinuities due to insufficient mesh resolution) that are expected to occur in the temperature and other flow field variables close to the flame front. This means that the high-order spatial discretization must be designed so as to provide the required accuracy throughout the flow field and yet remain monotone and robust in regions of under-resolved solution content.

Adaptive mesh refinement (AMR) is another approach for coping with the computational cost of large-scale numerical simulations, such as LES of turbulent combustion. For numerical simulations of complex flows characterized by a wide range of spatial and temporal scales both high-order discretizations and AMR are often demanded. For example, it is pointed out by Moin and Krishnan [11] that even ENO schemes as high as sixth-order can be too dissipative for the prediction of shock/turbulence interactions without refinement of the mesh in the vicinity of the shock wave. In other work, Jameson [12] has shown that high-order methods can be considerably more efficient than low-order AMR schemes for flows containing structures such as vortices, eddies and/or turbulence and concluded that high-

order schemes combined with AMR may provide very effective means of obtaining high solution accuracy. Moreover, even if a combination of high-order discretization and AMR can reduce significantly the cost of large-scale numerical simulations, a parallel implementation of these approaches is usually necessary in order to make the solution of such problems affordable on currently HPC hardware.

Consequently, it is felt that the design of a numerical algorithm that can successfully address the aforementioned issues arising in LES simulation of turbulent combustion should be based on a robust high-order spatial discretization procedure that can also be combined effectively with both mesh adaptation and an efficient parallel implementation.

## 1.2 Thesis Objectives

The main motivation for this research then stems from the high computational cost associated with the simulation of turbulent combusting flows using compressible LES techniques in conjunction with standard low-order spatial discretization procedures, as described above. Nevertheless, similar numerical challenges are encountered in many other scientific/engineering applications and the numerical methods developed in this thesis can be applied to this larger class of problems. The overarching objective of the research is to develop an efficient finite-volume procedure that, when used in conjunction with parallel implementations and AMR techniques, can make the performance of large-scale calculations, such as DNS and LES for engineering applications, more routine.

As such, this thesis aims to develop a robust high-order upwind finite-volume spatial discretization algorithm capable of providing significant computational savings to large-scale scientific computing applications such as the aforementioned ones. Part of this objective is to investigate strategies for obtaining high-order accuracy for curved body-fitted multi-block meshes. In order to facilitate analysis yet be practical, the high-order framework is developed for solving representative hyperbolic and/or elliptic conservation laws, such as the advection-diffusion, the Euler and the Navier-Stokes equations. Despite its development for a restrictive class of governing equations, the high-order framework is designed to be easily applied to numerical simulations of more complex flows, such as reacting and non-reacting LES in future follow-on studies. As previously mentioned, such physically complex flows are typically characterized by a wide range of spatial and temporal scales and therefore both

high-order discretizations and adaptive mesh refinement are often demanded. Consequently, the other objective of the research is to integrate the proposed high-order discretization with a two-dimensional parallel block-based AMR algorithm and to investigate refinement criteria other than physics-based techniques for directing the refinement process. The research is also expected to pave the way to some natural extensions of high-order algorithms, such as mesh refinement based on *a posteriori* error estimates, concurrent *h-p*-adaptation and *h-p*-multigrid methods.

## 1.3 Background

Not surprisingly, developing and exploring high-order methods to reduce the cost of numerical simulations is an active area of research, particularly for large-scale scientific computing applications such as DNS and LES of turbulent non-reactive and combusting flows, numerical computation of complex unsteady aerodynamic flows, aeroacoustic modelling, and computational electromagnetics. Standard lower-order spatial discretizations can exhibit excessive numerical error and are therefore very often not practical for the applications listed above. Improved numerical efficiency may be achieved by raising the order of accuracy of the spatial discretization, thereby reducing the number of computational cells required to achieve the desired solution accuracy. Moreover, for complex flows both high-order discretizations and adaptive mesh refinement may be required. Computational grids that automatically adapt to the solution of the governing equations are very effective in treating problems with disparate length scales, providing the required spatial resolution while minimizing memory and storage requirements. The main concepts and overviews of high-order methods and AMR techniques are reviewed next.

### 1.3.1 High-Order Methods

#### Basic Concepts and Elements

Current numerical algorithms for finding solutions to partial differential equations (PDEs) of the type encountered in the governing fluid flow equations are based on replacing the integrals or the partial derivatives (as the case may be) in these equations with discretized algebraic forms, which in turn are solved for the flow field values at discrete points in time and/or in

space. The solution of the algebraic equations represents an *approximation* that converges to the *continuous analytical solution* of the PDEs in the limit that the number of discrete points approaches infinity (i.e., as the mesh spacing goes to zero) if the numerical scheme satisfies certain conditions, such as those imposed by the Lax equivalence theorem for finite-difference methods [13]. The error of the approximate solution relative to the analytical solution is a function of the number of grid points and the *truncation error* of the numerical scheme, a subject amply discussed in a number of textbooks [14, 15]. For the sake of completeness, several finite-difference approximations are presented below to illustrate what the *scheme truncation error* is and how it affects the accuracy of the numerical solution.

Here, two approaches are considered for deriving algebraic difference quotients to replace the first order partial derivative,  $\left(\frac{\partial u}{\partial x}\right)_{i,j}$ , of a continuous flow variable,  $u$ , at the discrete point  $(i, j)$  shown in Fig. 1.2. The first approach uses the Taylor's series expansion with respect to a neighbouring point,  $u_{i+1,j}$ , given by

$$u_{i+1,j} = u_{i,j} + \left(\frac{\partial u}{\partial x}\right)_{i,j} \Delta x + \left(\frac{\partial^2 u}{\partial x^2}\right)_{i,j} \frac{(\Delta x)^2}{2} + \left(\frac{\partial^3 u}{\partial x^3}\right)_{i,j} \frac{(\Delta x)^3}{6} + \dots \quad (1.1)$$

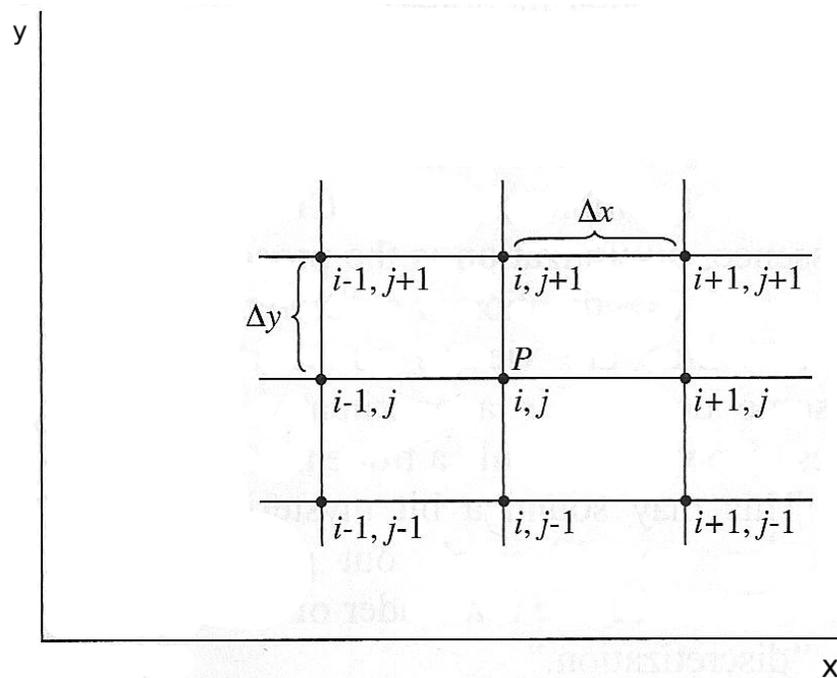


Figure 1.2: Grid points used in the discrete representation of the numerical solution.

to determine an expression for the first order derivative as

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \underbrace{\left(\frac{u_{i+1,j} - u_{i,j}}{\Delta x}\right)}_{\text{Finite-difference}} - \underbrace{\left(\frac{\partial^2 u}{\partial x^2}\right)_{i,j} \frac{\Delta x}{2} - \left(\frac{\partial^3 u}{\partial x^3}\right)_{i,j} \frac{(\Delta x)^2}{6} + \dots}_{\text{Truncation error}}, \quad (1.2)$$

in which both the *finite-representation* of the partial derivative and the *truncation error* are identified. The truncation error in Eq. (1.2) indicates what is being neglected in this approximation and because its lowest-order term involves  $\Delta x$  to the first-power it means that this particular finite-representation, called a *forward difference*, provides a *first-order-accurate* approximation to the exact derivative, formally written as

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \frac{u_{i+1,j} - u_{i,j}}{\Delta x} + \mathcal{O}(\Delta x). \quad (1.3)$$

Equivalently, it is said that the *order of accuracy* of the forward difference approximation is one.

Similarly, a finite-difference quotient of second-order accuracy can be derived as

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} + \mathcal{O}(\Delta x^2) \quad (1.4)$$

which is called a *central difference* approximation as fully described in the textbook by Lomax *et al.* [14]. The textbook of Lomax *et al.* also demonstrates how differencing schemes of any order can be constructed systematically based on Taylor tables.

The Taylor series expansions discussed above are in fact equivalent to standard polynomial series representations on uniform mesh and as such, the differencing schemes presented earlier actually represent the application of more general *polynomial approximation* techniques to uniform meshes. In this general approach, *truncated polynomial series* are used to approximate the continuous variation of the flow variable  $u$  over a given domain. The order of accuracy in this method is dictated by the polynomial order,  $k$ . For example, in  $k$ -exact reconstruction as proposed by Barth [16], the polynomial coefficients are computed so as the integrals of the polynomial function over a set of control volumes recover the average values of variable  $u$  in those elements, as explained in details in Chapt. 3. In the case of known pointwise solution values, these coefficients are determined by fitting the polynomial function to the variable values at a set of discrete neighbouring points, which can be taken equal or larger than the number of unknown coefficients.

Hence, assuming that  $u(x)$  can be expressed by the *quadratic* polynomial approximation

$$u^k(x) = u^2(x) = a + bx + cx^2, \quad (1.5)$$

the unknown parameters  $a$ ,  $b$  and  $c$  can be determined by constraining the polynomial expression at a number of points equal to the number of unknowns as follows:

$$\begin{aligned} u^2(0) = u_{i,j} &= a \\ u^2(\Delta x) = u_{i+1,j} &= a + b\Delta x + c\Delta x^2 \\ u^2(-\Delta x) = u_{i-1,j} &= a - b\Delta x + c\Delta x^2, \end{aligned} \quad (1.6)$$

where  $u_{i,j}$ ,  $u_{i+1,j}$  and  $u_{i-1,j}$  are assumed to be known. The solution of this linear system is  $a = u_{i,j}$ ,  $b = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x}$  and  $c = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{2\Delta x^2}$ . It can be easily seen that  $\left(\frac{\partial u}{\partial x}\right)_{i,j} = b$  yields the same expression as Eq. (1.4) for the first-order derivative, which shows that a quadratic polynomial provides a second-order-accurate representation of this derivative.

It is obvious from Eqs. (1.3) and (1.4) that, neglecting round-off errors or finite precision arithmetic of the computer, the accuracy of the finite-difference approximation increases with the decrease of  $\Delta x$ , the error becoming zero in the limit of infinite mesh resolution. Note that even if the order of truncation error does not quantify an exact value for the absolute error introduced by each approximation, it gives a clear indication of how the discretization error should behave as the mesh resolution is varied uniformly and smoothly. Hence, considering a given mesh resolution characterized by  $\Delta x_1$  on which the numerical approximation introduces an error,  $E_1$ , the error,  $E_2$ , generated by the first-order finite-difference given by Eq. (1.3) is *half* of  $E_1$  when the mesh resolution is doubled while maintaining the same relative point distribution, but *four times less* than  $E_1$  in the case when the second-order approximation, Eq. (1.4), is used. It is worth observing that under the same doubling mesh resolution condition, a *fourth-order approximation* would reduce the error *sixteen* times, thereby providing motivation for employing higher-order approximations in the numerical scheme.

To exemplify the concepts introduced so far and to illustrate the practical significance of a high-order approximation an example of  $k$ -exact reconstruction is considered in which the reconstruction of different polynomial orders is applied to the one-dimensional smooth trigonometric function  $f(x) = \sin(2x) + 2\cos(x)$  over a domain with 11 discrete elements in

which the exact average values had been calculated. The piecewise numerical approximations corresponding to constant ( $k = 0$ ), quadratic ( $k = 2$ ), cubic ( $k = 3$ ) and quartic ( $k = 4$ ) polynomials are shown at the top of Fig. 1.3. In the same figure the numerical derivatives obtained based on the quartic polynomial are compared against the first four analytical ones of this trigonometric function. It can be easily observed that as the polynomial order is increased the corresponding piecewise numerical approximation recovers the exact function with a smaller and smaller error. Moreover, solution jumps at interfaces are also reduced. Obviously, the quartic polynomial produces the best approximation of all orders considered. This is expected as its truncation error has order five (i.e.,  $\mathcal{O}(\Delta x^5)$ ). Note also that the accuracy of the derivative prediction decreases as the derivative order increases, the worst situation being encountered for the 4th-order derivative, shown in the bottom-left corner. This behaviour is also expected as each differential operation reduces by one the order of the truncation error in the polynomial approximation.

It is worth re-emphasizing here several well-known aspects regarding the properties of high-order approximations. One thing to bear in mind is that increasing the order does not necessarily increase the accuracy, especially in polynomial interpolation, and the equivalency between the two is subject to additional conditions. An important fact is that the aforementioned numerical properties are *only* valid provided that the function to be approximated is *smooth* in the sense that all derivatives up to the order of the truncation error are continuous (i.e., at least  $C^{k+1}$ -continuity). A violation of this condition renders incorrect the use of a Taylor series expansion with a degree higher than the order of discontinuity of the exact function and can lead to large errors in the numerical approximation. An illustration of this problem is provided by the  $k$ -exact reconstruction for a discontinuous function as shown in Fig. 1.4(a), in which *Gibbs-like phenomena* (i.e., spurious oscillations proportional to the size of the jump) can be identified in the vicinity of the discontinuity. Moreover, the spurious oscillations observed near discontinuity do not disappear as the mesh resolution is increased, a fact shown in Fig. 1.4(b).

In addition to introducing significant errors near discontinuities, there are other problems related to the occurrence of over- and undershoots in the solution approximation. First, the values predicted by the polynomial approximation may lie outside the allowable range (e.g., for compressible gaseous flow, reconstructed values of pressure and/or density may be negative, violating positivity requirements for these variables arising from the physics) and

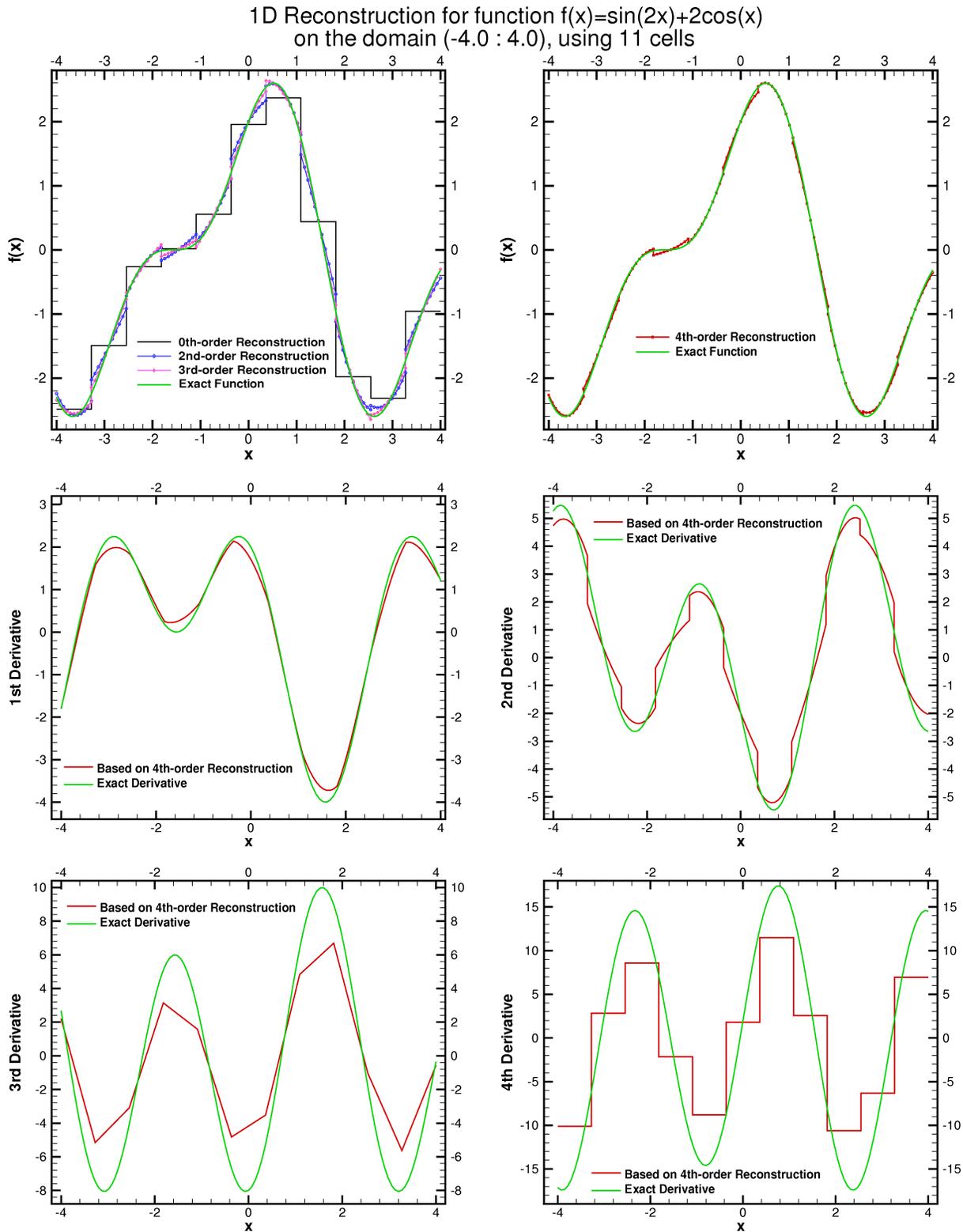


Figure 1.3: Comparison of  $k$ -exact reconstructions relative to the exact function shown for  $k=0, 2, 3$  and  $4$  in the top two plates. The rest of the figures illustrate the first four exact function derivatives and their approximation based on the quartic ( $k=4$ ) reconstruction.

second, these wiggles can grow in time and completely pollute the accuracy of the numerical discretization. Equally hazardous to the numerical scheme can be the wild oscillations that may occur for smooth function interpolation with significantly high orders (e.g., five, six and higher), a behaviour known in the literature as the *Runge's phenomenon* [17]. The occurrence of Runge oscillations is one of the main reasons why it is seldom the practice that very high interpolation orders are used, unless there is quite rigorous monitoring of estimated errors. A more robust and frequently used approach is to define the high-order interpolant as a *piecewise function* comprised of lower order interpolants, such as cubic and quartic, on each domain interval. Therefore, in addition to accuracy another important issue in the design of space discretizations is the *robustness* property of the numerical scheme, which refers to the behaviour of the scheme near large solution gradients and/or discontinuities.

An obvious question that may arise is why one would use a high-order polynomial approximation near such problematic regions, considering that it can lead to the aforementioned issues. Note that these regions correspond not only to areas in which the solution would naturally exhibit physical discontinuities but also to domains on which the solution is presently deemed under-resolved due to a lack of sufficient mesh resolution and appears as discontinuous on the current grid. Naturally, if one knew *a priori* where those regions with solution discontinuities were located, which is seldom the case in general, then one would not

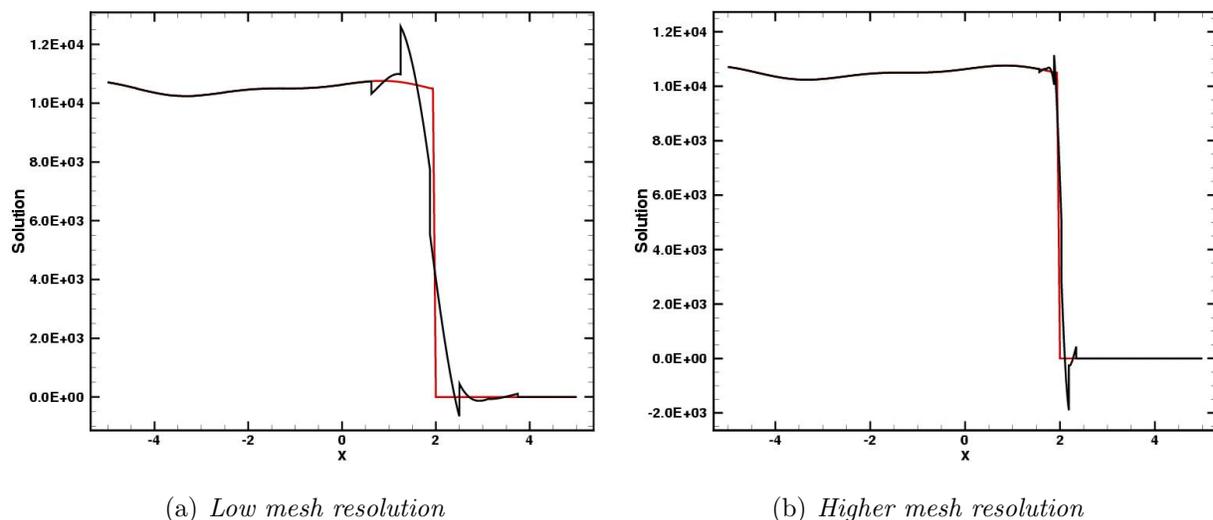


Figure 1.4: *Illustration of Gibbs-like phenomenon for a cubic ( $k=3$ ) reconstruction on two different resolution meshes.*

employ high-order polynomial approximation at those locations. Moreover, discontinuous regions typically coexist with smooth solution regions for which high-order approximations can present significant advantages as it has been shown. In conclusion, high-order polynomial approximations are desirable for regions where they can accurately resolve the solution with fewer mesh points than the lower-order ones, but they also must possess a mechanism to ensure the robustness of the numerical approximation near sharp solution variations.

Finally, a big challenge associated with using high-order solution procedures is that in general, regardless of the method of choice (e.g., finite-differences, finite-volume or finite-element algorithms), the computational cost of the numerical scheme increases with the order of accuracy and therefore it can mitigate the benefits generated by smaller truncation errors. In our practical example, the evaluation of the quartic polynomial is obviously more expensive than the cost associated with the quadratic one as additional coefficients are determined. On the positive side, due to a lower truncation error a higher-order method would require *fewer* computational cells than a lower-order one to obtain the same level of accuracy, that may translate into fewer numerical operations and ultimately into a reduced *total computational cost* for the entire problem solution. In other words, there is a trade-off between the *order of accuracy* and the *computational cost* associated with it, and whether a high-order method would successfully manage to reduce the cost of a numerical simulation is in many cases very dependent on the characteristics of the problem under consideration and the efficiency with which the high-order accuracy is obtained. Consequently, it may be that not all flow problems necessitate a high-order treatment and that only certain types of flows, such as the aforementioned ones, are potentially well suited to the application of high-order methods.

### Overview of the Development of High-Order Methods

In the last decades, there have been a number of studies of high-order schemes. For hyperbolic conservation laws and/or compressible flow simulations, the challenge has been to achieve accurate discretizations while coping in a reliable and robust fashion with discontinuities and shocks. The essentially non-oscillatory (ENO) schemes first proposed by Harten *et al.* [18] provide robust frameworks for high-order finite-volume discretizations of hyperbolic systems. Such schemes remain uniformly accurate yet are *essentially non-oscillatory* in the sense that they do not produce Gibbs-like  $\mathcal{O}(1)$  oscillations at discontinuities, although *small* spurious

oscillations may be generated having a magnitude on the order of the truncation error (i.e.,  $\mathcal{O}(\Delta x^{k+1})$ ). The latter disappears as the mesh is refined. The first interpolation technique proposed by Harten *et al.* to provide the ENO property is based on an adaptive-stencil strategy, in which the stencil leading to the “smoothest” reconstruction is selected and therefore stencils containing discontinuities are avoided. More recently, Harten and Chakravarthy [19] have proposed a technique for Cartesian grids to obtain an ENO reconstruction on a fixed central stencil by hybridizing the high-order reconstruction with a first-order formulation. Abgrall [20] and Sonar [21] have since extended the ENO concept for application to flow prediction on unstructured grids. In addition, so-called weighted ENO (WENO) schemes have been developed for both structured and unstructured meshes [22, 23, 24]. However, the difficulty with these finite-volume approaches has been the extension of the method to multi-dimensional problems and large systems of coupled PDEs. The computational challenges are primarily related to the selection of stencils on multi-dimensional meshes, particularly in the case of unstructured grids [20, 21, 25, 26], and the poor conditioning of the linear systems that define the solution reconstructions for these stencils [25, 26]. The requirement of using a different stencil for each flow variable is also perceived to be problematic, particularly for problems in combustion, where there can be a large number of species present in the reactive mixture of gases. More recently, Haselbacher [26] has explored the use of fixed stencil central reconstruction in the formulation of WENO schemes for unstructured meshes, but schemes of accuracy higher than second order were not formulated. Additionally, some perceived challenges associated with non-compact high-order finite-volume schemes are the larger stencils which may lead to problems with the boundary-condition implementation and efficient parallelization of the algorithm. For application to practical engineering problems and arbitrary boundary topologies, the ability of the approach to deal with all of these challenges is especially important. Although successful implementations have been developed, in general the computational costs and complexity of implementing ENO and WENO schemes have limited their widespread application. Nevertheless, combinations of high-order ENO and WENO schemes with AMR for both structured and unstructured meshes have been developed and applied to engineering problems such as the prediction of high-speed flows as described in the recent work by Wolf and Azevedo [27].

Other researchers have considered more flexible high-order finite-volume schemes, which may be more easily extended to multi-dimensional problems and to unstructured meshes.

For example, Barth and Fredrickson [16, 28] developed a high-order finite-volume approach for unstructured mesh based on  $k$ -exact least-squares reconstruction. Following on their work, Ollivier-Gooch [29, 30] has more recently proposed a data-dependent weighted least-squares reconstruction procedure (DD-ENO), that uses a fixed stencil and seeks to enforce monotonicity of the scheme by introducing data-dependent weights for each point in the stencil. To circumvent some deficiencies in the DD-ENO procedure, Ollivier-Gooch *et al.* reconsidered the use of non-linear limiters for controlling oscillations and defined a high-order limiting strategy in combination with a  $k$ -exact reconstruction procedure [31, 32]. Capdeville recently revisited the DD-ENO concept and formulated a compact Hermite least-square monotone (HLSM) reconstruction scheme in one-dimension [33]. Additionally, Colella *et al.* [34] have developed a high-order finite-volume method in mapped coordinates for discretization of linear elliptic and hyperbolic PDEs. Their approach has been already extended to nonlinear systems of hyperbolic conservation laws on locally-refined grids by McCorquodale and Colella [35]. To suppress undesirable solution oscillations, the latter authors applied a fairly elaborate combination of slope limiters, slope flattening and artificial viscosity to their numerical scheme. Moreover, their proposed multidimensional limiting strategy employs a modified version of the one-dimensional limiter formulated by Colella and Sekora [36] for preserving accuracy at smooth extrema.

Before continuing it should be noted that, recent evidence obtained on more theoretical grounds for the class of high-order upwind finite-volume methods with explicit time-integration scheme applied to one-dimensional wave propagation problems suggests that the original ENO schemes are more cost-effective for an imposed level of solution error than the WENO counterparts. The results of the performance analyzes carried out by Pirozzoli [37, 38] for different upwind shock-capturing schemes, which are also shown in Fig. 1.5 courtesy of S. Pirozzoli [38], clearly indicate that, based solely on cost-effectiveness (i.e., computational cost for a given level of accuracy), any future development of multi-dimensional high-order finite-volume schemes should be considered more in the vein of the original ENO approach due to Harten *et al.* [18] than along the lines of the WENO methods, which suffer somewhat from a lack of robust, reliable, and accurate means for evaluation of the stencil weights. Note that the computational cost,  $C$ , in Pirozzoli's analyzes has been determined based on a normalized measure of the CPU time per grid point per time step required for one residual evaluation of each scheme considered in the study.

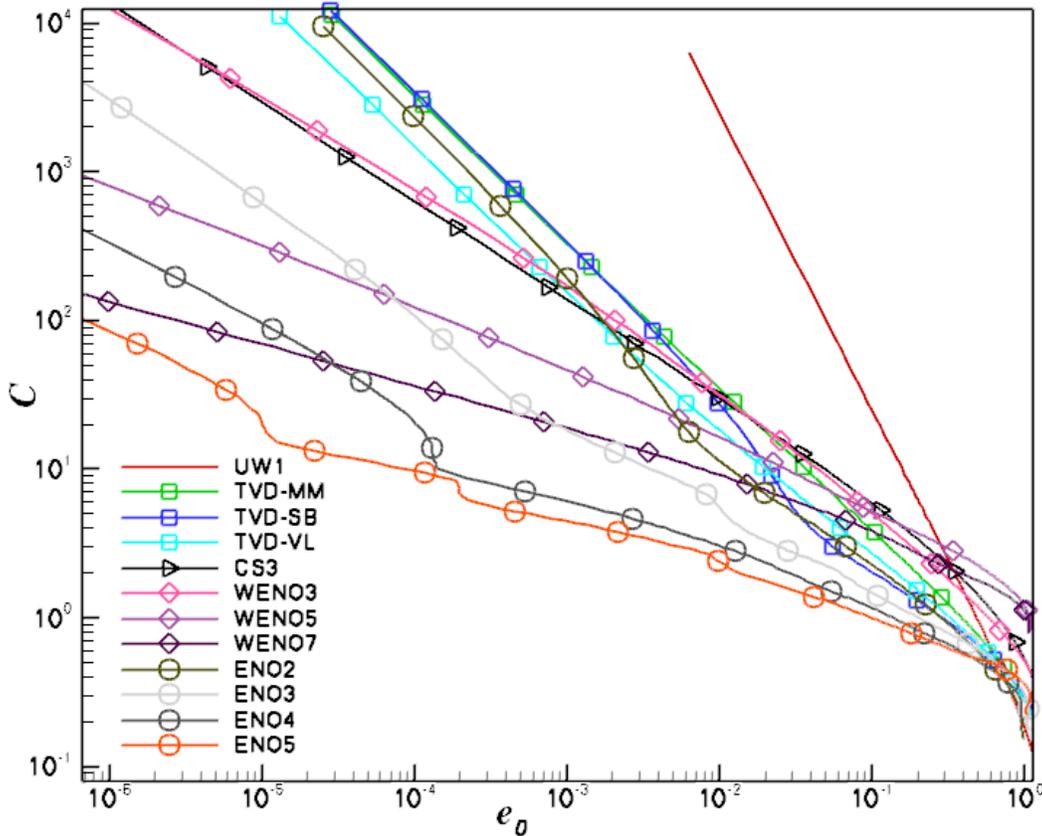


Figure 1.5: *Plots of estimated computational cost vs. error in one-space dimension for several shock-capturing schemes, courtesy of S. Pirozzoli [38]. The following schemes have been considered in the study: (1) second-order total variation diminishing with the minmod (TVD-MM), superbee (TVD-SB) and Van Leer (TVD-VL) flux limiters; (2) third-order compact scheme of Cockburn and Shu [39] with minmod limiter (CS3); (3) WENO schemes with accuracy order of three (WENO3), five (WENO5) and seven (WENO7); and (4) ENO schemes with accuracy order of two (ENO2), three (ENO3), four (ENO4) and five (ENO5).*

High-order finite-element schemes have also been considered for fluid flows involving discontinuities. In particular, the class of schemes now generally referred to as discontinuous Galerkin (DG) schemes have gained in popularity. Cockburn *et al.* [40, 41, 42] were the first to formulate a family of high-order total variation bounded (TVB) schemes for nonlinear systems of conservation laws referred to as Runge-Kutta discontinuous Galerkin (RKDG) methods. By combining elements from the finite-element and finite-volume methods, Cock-

burn *et al.* have obtained a family of numerical schemes that improve the data locality and yet allow for discontinuous solution spaces. Hence, a flexible high-order spatial discretization with a greatly reduced stencil size can be formulated. The reduced stencil is beneficial for both boundary-condition implementation and algorithm parallelization. To ensure non-oscillatory properties of the predicted solutions near discontinuities and/or large gradients, DG methods have been considered in combination with different limiting strategies, such as TVB- [42] and WENO-type [43] limiters. Additionally, Barter and Darmofal [44] formulated a DG scheme with a PDE-based artificial viscosity model to avoid spurious numerical oscillations in the proximity of shocks. Xu *et al.* [45] have also recently applied a hierarchical reconstruction (HR) strategy to DG methods in combination with a WENO-type linear reconstruction in each hierarchical level. Despite the interest generated by the DG formulation over the last decade, high-order DG schemes do suffer from a more restrictive time-step stability limit that is exacerbated as the solution order is increased [46] and from a relatively high computational cost per cell associated with the solution of the additional unknowns for the high-order terms within the cell. The former can be particularly problematic for explicit time-marching schemes. Partly in response to the latter, it is noted that Dumbser *et al.* [47] have recently formulated a unified framework for finite-volume and discontinuous Galerkin methods on unstructured mesh which introduces a new family of reconstructed DG schemes denoted as  $P_N P_M$  methods.

In other work, Wang *et al.* [48, 49] and Liu *et al.* [50] have proposed variants of the DG formulation, referred to as spectral volume (SV) and spectral difference (SD) methods, for obtaining high-order solutions to hyperbolic conservation equations. In fact, DG, SV and SD methods share many similarities, the main difference consisting in how the degrees of freedom of each scheme are chosen and updated. A new perspective regarding the DG, SV and SD methods has been introduced by Huynh [51], who was able to recover all these methods in one-space dimension as particular cases of a general formulation based on a flux reconstruction (FR) approach to high-order schemes for hyperbolic laws. For dealing with undesirable solution oscillations, the SV and SD methods have been usually considered in combination with total variation diminishing (TVD) and TVB limiting strategies. However, Yang and Wang [52, 53] have recently combined a SD scheme for arbitrary unstructured grids with a compact high-order hierarchical moment limiter. More recently, Wang and Gao [54, 55] have generalized Huynh's flux reconstruction approach to mixed grids and

formulated a unifying lifting collocation penalty (LCP) method for Euler equations and applied it to smooth flows. An advantage of their unified formulation is that the relative cost of LCP method with respect to a DG scheme was inferred easily by simply comparing the main operations required for the two algorithms. In addition, the newly proposed LCP formulation is shown to be more efficient in both memory and CPU time than DG methods.

Alternative high-order schemes have also been developed and are currently being pursued for application on either unstructured and/or multi-block structured meshes. For example, recent advances in high-order residual distribution (RD) methods [56, 57] are represented by the non-linear ‘mapped’ distribution schemes proposed by Abgrall and Roe [58]. More recently, Abgrall [59] have formulated an essentially non-oscillatory RD scheme for hyperbolic problems. Additionally, Guzik and Groth [60, 61] have compared higher-order RD and FV schemes quantitatively and proposed a new high-order mapped RD method for hyperbolic conservation laws. In other work, Dias and Zingg [62] have developed a high-order parallel Newton-Krylov finite-difference algorithm for three-dimensional structured multi-block meshes and inviscid aerodynamic flows. One of the key features of the latter method is the use of simultaneous approximation terms (SATs) [63] to deal with block interfaces and boundaries, which allows it to be more flexible with respect to the multi-block grid topology.

High-order schemes for PDEs governing diffusion processes and/or having a more elliptic nature (e.g., conservation equations governing viscous flows) have also been considered. In these cases, it is desirable that the discretization of the elliptic operator remain accurate while satisfying a maximum principle, even on stretched/distorted meshes [64, 65]. Even standard lower-order spatial discretizations may not always have these characteristics. Sun *et al.* [66] and May and Jameson [67, 68] have considered the applications of the SV and SD methods, respectively, to viscous flows. The latter have also applied their scheme in combination with AMR to the Navier-Stokes system of equations. More recently, Gao and Wang [69] and Haga *et al.* [70] have applied the LCP formulation to Navier-Stokes equations on two- and three-dimensional mixed grids, respectively. Barad and Colella have proposed a fourth-order-accurate AMR scheme for Poisson’s equation [71] and Ollivier-Gooch and van Altena [72] describe a general high-order framework for the solution of the advection-diffusion equation on unstructured mesh. In other work on diffusion problems on unstructured grids, van Leer *et al.* [73, 74] proposed a recovery-based DG method that eliminates the introduction of *ad hoc* penalty or coupling terms found in traditional DG methods. More recently, Oliver and

Darmofal [75] applied the DG with a PDE-based artificial viscosity model to aerodynamic flows governed by RANS equations. Finally, De Rango and Zingg [76] have considered the application of high-order finite-difference methods to the prediction of turbulent aerodynamic flows.

In spite of the advances in high-order methods outlined above, there is still no consensus for a robust, efficient, and high-order accurate scheme that fully deals with all of the aforementioned issues and is applicable to more arbitrary mesh configurations. Additionally, with respect to the class of high-order finite-volume methods, it seems that there is not yet a formulation which to allow an efficient extension of the ENO ideas to multi-dimensional, multi-variable, and complex boundary-condition problems.

### 1.3.2 Solution Adaptivity

#### Overview of Mechanics of Adaptation

Solution-adaptation procedures are widely used and accepted strategies for improving the accuracy of numerical simulations while limiting the increase in computational cost. Currently, several different adaptation strategies have emerged to adjust the degrees of freedom (DOF) of a solution procedure. These approaches which originate in the finite-element method (FEM) community have been quickly adopted by the finite-volume practitioners and further refined to fulfil the needs in this branch of computational science. They can be classified into four broad categories depending on whether the changes introduced during the adaptation process affect the computational grid and/or the order of accuracy of the numerical scheme (i.e., the degree of the element in FEM). They are as follows: (1)  $p$ -adaptation, (2)  $h$ -adaptation, (3)  $r$ -adaptation and (4)  $h$ - $p$ -adaptation methods. A first analysis of mathematical properties of all these procedures in the context of FEM has been carried out by Gui and Babuška in [77]. A brief review of these adaptation strategies, in which the advantages and disadvantages of each method are also pointed out, is given below.

In  $p$ -adaptation, sometimes referred to as  $p$ -refinement, increased accuracy is achieved by changing the interpolation order of the numerical scheme locally while the mesh remains fixed. In spite of the excellent error convergence achieved by this method for smooth flow variations, difficulties arise for flows in which non-smooth solutions and/or discontinuities are present. For such situations, a strongly refined mesh near singularities or discontinuities

is very important to get the desired accuracy. Such a mesh can be generated by performing  $h$ -adaptation where the grid is locally modified so as to increase the mesh resolution near solution features that need to be resolved while the order of interpolation is unchanged. This approach offers the potential for large savings through optimized mesh resolution. Further increase in computational efficiency can be obtained by coarsening the mesh concurrently with the refinement process in regions of over-refinement. The  $h$ -adaptation methods can be performed isotropically or anisotropically (i.e., directional refinement) and can be very effective for improving mesh efficiency in boundary layers, wakes, shocks and near complex geometries as well as in tracking sharp flow features that move through the computational domain. In general,  $h$ -refinement is considered to be the standard version of performing solution adaptation, mainly for reasons of cost and ease of programming as compared to  $p$ -adaptation. Adaptive re-meshing, known as  $r$ -adaptation, is a simpler variation of  $h$ -adaptation in which the mesh nodes are reallocated while maintaining the same number of DOF. The mesh redistribution in this case is usually solved by employing algorithms based on a spring analogy [78] or on a linear elasticity analogy [79, 80]. In general, this adaptation procedure is less expensive than re-meshing the entire solution domain and easier to program than  $h$ -type adaptation. In addition to many other applications [81, 82], this approach has proved very useful in adjusting the mesh for minor shape modifications arising in applications of shape optimization algorithms [83].

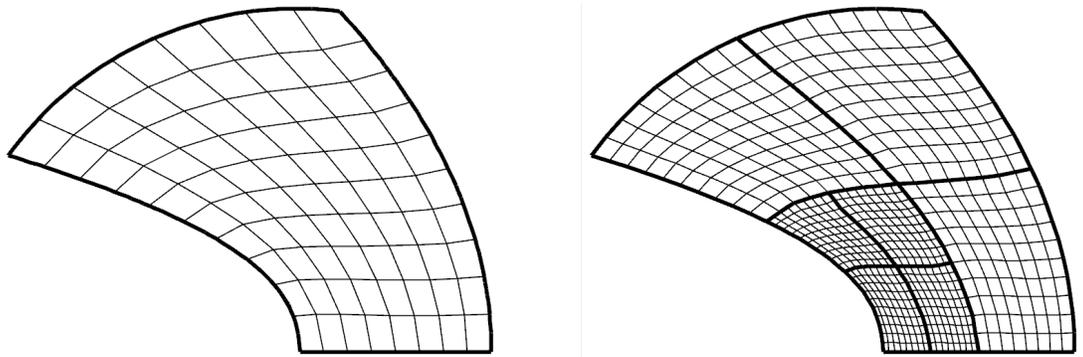
Finally, there are problems in which regions of rapidly varying physical changes coexist with large domains of smoothly varying flows (i.e., large length scale disparity), such as the thin premixed flame sheet deformation, and for which a combination of  $h$  and  $p$ -adaptation may be highly advantageous [84]. In general,  $h$ - $p$ -adaptation algorithms simultaneously refine the mesh and increase the order of accuracy of the numerical scheme, employing either one or both of the adaptation strategies in different regions depending on which approach is deemed to be more beneficial to solution accuracy. A remarkable property of this combined approach is the capability to produce exponential convergence rates [77, 85]. However, the choice of adaptation method in a particular area, that is  $h$  and/or  $p$ -adaptation, is not always clear and suitable criteria must be developed to aid that decision [84]. The formulation of adaptation criteria is the weakest link of automatic solution adaptivity methods and much literature has been devoted to this matter. More on this subject will be described in a later section.

In this thesis, the adaptation strategy considered for study in conjunction with the proposed high-order finite-volume algorithm is based on an  $h$ -refinement method. Nevertheless, the computational framework developed over the course of this project could be extended relatively easily to include both  $p$  and  $h$ - $p$ -adaptation due to its already embedded capability to generate numerical solutions of different accuracy orders. However, these directions have not been extensively pursued and rather than covering a broad area, the research has been focused instead on the challenges presented by the integration of a high-order algorithm on body-fitted mesh with a block-based AMR technique. Therefore, only elements of the block-based  $h$ -type adaptation technique considered herein are described next.

### Block-Based AMR

Following the work of Berger and Olinger [86] in adaptive meshing techniques for hyperbolic PDEs in multiple space dimensions, many related grid adaptation strategies have been developed since and applied to a variety of engineering problems [87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99]. Currently, the following broad categories of AMR methods can be identified: “patch-based”, “cell-based”, “block-based”, and “hybrid block-based”. An overview of all these AMR approaches is provided in the recent work by Gao [100].

In a block-based AMR technique, mesh adaptation is accomplished by the refining and coarsening of appropriate solution blocks. In general, each block has the same number of cells (i.e., they are *self-similar*) for computational efficiency and ease of implementation. In the refinement case, the mesh resolution is increased by dividing the block requiring refinement into four (in two-dimensions) new solution blocks which provides a resolution change of two. An illustration of this procedure is depicted in Fig. 1.6, in which the initial solution block and the final refined multi-block mesh after two refinement levels are shown in Fig. 1.6(a) and (b), respectively. In a similar manner, solution block coarsening is carried out by merging four neighbour blocks into one block with twice as lower mesh resolution. Consequently, a hierarchical tree-like data structure (e.g., a quadtree in two-dimensions) naturally lends itself for tracking the connectivity between blocks. In addition, the block-based AMR is well suited to parallel implementation via domain decomposition, in which case efficient and readily scalable parallel algorithms have been shown to result [98].



(a) A body-fitted coarse grid

(b) A refined body-fitted grid consisting of seven self-similar grid blocks

Figure 1.6: *Illustration of block-based mesh adaptation on a two-dimensional body-fitted grid. Block boundaries are depicted with thick lines while thin lines show the interior mesh.*

### Adaption Criteria

As previously mentioned, the crux of automatic mesh adaptation methods is the *a priori* translation of the user interest in resolving representative simulation characteristics such as flow features (e.g., shock waves, wakes, boundary layers) and/or output functionals (e.g., lift, drag), into a set of mathematical statements which to instruct the solution procedure to automatically direct the adaptation process so as to capture those required details with the desired level of accuracy. Obviously, the quality and the convergence properties of the numerical solution are very dependent on the chosen criteria and how mesh adaptation is performed. Moreover, whether the resultant adapted mesh is optimum in the sense that it produces the lowest error in a quantity of interest relative to the exact analytical solution for a specified number of DOF is definitely arguable (see [101] for an analysis of optimal mesh adaptation). In practice, a combination of multiple criteria are typically employed to get the desired effect: that is a reduced mesh size for the same desired level of solution error.

A common strategy is to adapt the grid based on the presence of strong local flow features, such as shock waves and/or high vorticity regions, which are identified using the so-called “physics-based” criteria. These indicators use flow variable gradients to isolate regions of rapid solution variations. Some disadvantages of this approach are that in general, it may require extensive tuning of the adaptation parameters in order to capture all desirable features and it also presents difficulties in identifying non-local (i.e., convective) features, such

as wakes [102]. Moreover, continuous local refinement of the dominant features of the flow do not necessarily result in a reduction of the discretization error, especially in the context of adaptive high-order solution procedures. A better alternative may be to perform mesh adaptation based on *a posteriori* error estimates, in which the adaptation is directed based on interpolation [103, 104] and/or residual [105] error estimators.

For applications in which it is important to predict integral quantities such as lift and drag forces on aircraft configurations, the aforementioned adaptation techniques may not always identify all regions that contribute *significantly* to the accuracy of these functionals and a more appropriate approach to direct mesh adaptation would be to use output error estimates based on adjoint analysis [106, 107, 108, 109, 110]. Basically, this method computes the discrete sensitivity of output functionals with respect to local solution residual using a dual-weighted residual method and consequently resolves more accurately those areas which contribute mostly to their accuracy (i.e., regions in which the output functionals are most sensitive to). A current disadvantage of this approach is that for unsteady flow adaptation the computation of the solution of adjoint equations (i.e., the sensitivities), which is required by this method, is a very expensive operation [111].

This work seeks to advance the capabilities of the block-based AMR algorithm proposed by Groth *et al.* [98, 112] for standard second-order upwind finite-volume schemes on body-fitted multi-block meshes. The approach has been already successfully applied to the prediction of laminar and turbulent diffusion flames [92, 96], micron-scale flows [113], and multi-phase rocket motor core flows [98]. This thesis research considers the extension of this framework to high-order solution of inviscid and viscous problems and the investigation of refinement criteria other than physics-based. The somewhat novel refinement indicators proposed in this work are based on a measurement of the degree of smoothness of high-order solution interpolants within their neighbourhood. Although this type of error quantification is not strictly speaking a direct measurement of the local interpolation error, it is shown to be quite effective, at least for a certain range of mesh resolutions. The proposed refinement criteria are shown to recognize regions with large discretization errors and/or unresolved solution content and to prevent over-refinement of regions in which the solution has been already resolved. Moreover, the assessment of accuracy and capability of the proposed combined high-order adaptive algorithm has been carried out to determine the potential savings of such an approach. However, it is felt that the proposed refinement indicators should be

better considered in conjunction with error estimators. Particularly, smoothness-based indicators can represent an inexpensive solution to obtain adequate grid resolutions from which output error estimates can become reliable and refine the solution with the desired accuracy.

## 1.4 Overview of Dissertation

In response to the main thesis objectives and previous research outlined above, the following avenues of research were pursued in this work:

- formulation of a high-order central essentially non-oscillatory (CENO) finite-volume scheme for solution of hyperbolic and elliptic systems of conservation laws;
- development of a fourth-order CENO scheme in conjunction with a parallel block-based adaptive mesh refinement (AMR) algorithm for body-fitted multi-block quadrilateral meshes and study of the predictive capabilities of this algorithm for solutions of the advection-diffusion, the Euler and the Navier-Stokes equations;
- formulation of a smoothness indicator which enables identifying regions of under-resolved and/or non-smooth solution content and assessment of the usefulness of an  $h$ -refinement criterion based on the smoothness indicator for mesh adaptation; and
- demonstration of a competitive high-order ENO-like formulation for multi-dimensional, multi-variable, and complex boundary-condition problems.

Additionally, the detailed knowledge about the discretization properties of the AMR CENO solution procedure obtained in this work based on analytical and numerical investigations carried out on a restrictive class of governing flow equations provides a good starting point for the application of the scheme to more complicated governing equation sets, such as those present in simulation of LES of turbulent combustion. The detailed characterization of the scheme included herein can prove especially valuable for such situations in which the algorithm development process is challenged by the lack of analytical or manufactured solutions [114] for numerical scheme verification and by the typical large computational cost required even by modest simulations used in the verification process.

This thesis document is organized as follows. Chapter 2 serves to review the conservation equations considered to be solved in this work and highlights their important mathematical

properties. In Chapt. 3, the numerical procedure proposed in this thesis for solving inviscid and viscous flows is presented in detail. The focus of this chapter is on introducing the high-order CENO approach and on assessing the properties of the discretization procedure. The extension of the CENO approach to multi-block meshes is considered in Chapt. 4. Furthermore, the same chapter treats the coupling of the CENO procedure with parallel block-based AMR techniques and introduces the refinement criteria considered in this work. Chapter 5 is devoted to numerical results obtained with the high-order CENO framework and includes both validation and simulation cases. Finally, conclusions and directions for future work are summarized in Chapt. 6.

# Chapter 2

# CONSERVATION EQUATIONS OF INTEREST

*“Equations are more important to me, because politics is for the present, but an equation is something for eternity.”*

by Albert Einstein (1879 - 1955), German-born theoretical physicist

The high-order solution procedure developed and thoroughly investigated in this work is applied to solutions of the advection-diffusion, the Euler and the Navier-Stokes equations, which are representative of many physical transport processes yet facilitate numerical analysis. This chapter provides a review of the conservation equations considered herein and emphasizes mathematical properties relevant for the development of numerical procedures.

## 2.1 Advection-Diffusion Equation

The advection-diffusion equation is a very convenient mathematical model for the development of numerical solution algorithms for it contains many of the features of more complicated governing equation sets used in practical fluid flow modelling applications, yet it retains a simplicity that readily permits direct analysis. Moreover, this generic scalar transport equation has been the subject of intense studies because of its appearance in modelling a broad range of physical phenomena which occur in fields as diverse as combustion, geology and biophysics thus its mathematical properties are well documented in the literature.

The particular form of the advection-diffusion equation considered herein is given by

$$\underbrace{\frac{\partial u}{\partial t}}_{\text{Transient term}} + \underbrace{\vec{\nabla} \cdot (\vec{V}(\vec{r}, u) u)}_{\text{Advective term}} = \underbrace{\vec{\nabla} \cdot (\kappa(\vec{r}, u) \vec{\nabla} u)}_{\text{Diffusive term}} + \underbrace{\phi(\vec{r}, u)}_{\text{Source term}}, \quad (2.1)$$

where  $u$  is the solution (a scalar quantity),  $\vec{V}$  is the advection velocity vector,  $\kappa$  is the diffusion coefficient, and  $\phi$  is a non-linear source term. In Eq. (2.1), the time variation of the local solution, accounted for by the *transient term*, occurs due to the existence of a velocity field,  $\vec{V}$ , which convects the scalar quantity  $u$  and thus introduces an *advective term* in the equation, due to the diffusive transport of  $u$  generated by solution gradients and represented in the equation by the *diffusive term*, and due to any sources or sinks that either create or destroy  $u$  which are accounted for by the *source term*. In the most general case,  $\vec{V}$  and  $\kappa$  are functions of the position vector,  $r$ , as well as the solution,  $u$ . Based on the relative magnitudes of the advective and diffusive fluxes, the solutions of this equation can range from those having a more hyperbolic nature and governed by wave propagation phenomena to those having a more elliptic nature and governed by diffusive processes.

This thesis considers numerical solutions of the two-dimensional (2D) planar form of the advection-diffusion equation, Eq. (2.1), which can be written in Cartesian coordinates as

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left( v_x(\vec{r}, u) u \right) + \frac{\partial}{\partial y} \left( v_y(\vec{r}, u) u \right) = \frac{\partial}{\partial x} \left( \kappa(\vec{r}, u) \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( \kappa(\vec{r}, u) \frac{\partial u}{\partial y} \right) + \phi(\vec{r}, u), \quad (2.2)$$

where  $v_x$  and  $v_y$  are the components of the advection velocity vector,  $\vec{V}$ , in the Cartesian  $x$ - and  $y$ -coordinate directions. In this two-dimensional formulation of the advection-diffusion equation the position vector,  $\vec{r}$ , is a function of the two spatial coordinates,  $x$  and  $y$ . Several relevant forms of this equation are described next.

An example of a hyperbolic PDE is provided by the scalar linear advection equation

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0, \quad (2.3)$$

which is recovered from Eq. (2.2) by considering only the one-dimensional advection term with a constant wave propagation speed,  $a$ , and no diffusion or source terms. This equation is hyperbolic for any *real* value of the PDE *eigenvalue*, which is the coefficient  $a$ . This PDE will simply translate an *initial condition* (IC) with velocity  $a$  in the positive or negative  $x$ -direction, depending on the sign of  $a$ . The analytical solution can be obtained with the *method of characteristics* [115]. Consequently, this particular PDE provides a very useful mean of testing the accuracy of numerical discretizations, especially when used in combination with *periodic* boundary conditions [14]. Note that discontinuous profiles are admitted solutions of this PDE which raises questions about how the spatial discretization should

be performed so as to avoid the occurrence of *Gibbs phenomenon* near solution discontinuities/large gradients.

At the opposite end of the spectrum, the classical Poisson's equation

$$\nabla^2 u = \phi(x, y, u) \quad (2.4)$$

is an example of elliptic PDE to which Eq. (2.2) can be reduced by including only the diffusive and source terms and set  $\kappa$  to be constant. The Laplace equation,  $\nabla^2 u = \Delta u = 0$ , is the simplified version of Poisson's equation, Eq. (2.4), for zero source term. Analytical solutions of these equations, and in general to elliptic PDEs, are characterized by their smooth profiles. Moreover, they satisfy a maximum principle which is another fundamental property. Statements of this principle in different forms are provided in the literature [116, 117], and for sake of completeness, the *strong* formulation given in [116] is reproduced without proof below.

**Theorem 2.1.1.** *The maximum-minimum theorem. Consider a harmonic function  $u(x, y)$  continuous in some closed bounded region  $\bar{G} = G + \Gamma$ . Then the values of  $u$  in  $G$  cannot exceed its maximum on  $\Gamma$  nor can they be less than its minimum on  $\Gamma$ .*

Theorem 2.1.1 basically says that if a function achieves its maximum in the interior of the domain, the function is uniformly constant. This property is very difficult to enforce in numerical solutions, especially on skewed meshes, and therefore, *weaker* formulations are better suited to development of numerical algorithms. Thus, the weak principle says that the maximum of the function is to be found on the boundary, but *may re-occur* in the interior as well. Other even weaker formulations exist which merely bound a function in terms of its maximum on the boundary. Numerical violations of this principle may introduce non-physical phenomena, such as heat flowing from lower to higher temperature [65].

In conclusion, it is very desirable that numerical schemes used for the solution of the advection-diffusion equation do not introduce excessive artificial dissipation, large dispersion error and spurious oscillations arising from the discretization of the hyperbolic term, and provide an accurate discretization of the elliptic term while satisfying a maximum principle, even on stretched and/or distorted meshes.

## 2.2 Euler Equations

The governing equations of inviscid compressible gas flow represent a system of conservation laws for mass (i.e., continuity), momentum (i.e., Newton's second law) and energy (i.e., first law of thermodynamics), called the Euler equations. A fluid is defined to be *inviscid* if it has no or negligibly small viscosity, which means that the fluid can support no shearing stress and no thermal conductivity, and consequently it flows without energy dissipation in the absence of any heat source terms, such as those accounting for radiation heat transfer [118]. For many flow situations, the inviscid fluid approximation is a relatively adequate assumption, allowing the Euler equations to be successfully applied to a large range of practical flow problems. From a mathematical perspective, approximating real fluids (i.e., fluids with *viscosity*) as inviscid or “*ideal*” introduces considerable simplifications to the governing equations of motion. Thus, in comparison to the *second-order* system of Navier-Stokes equations governing the dynamics of continuous real fluids as described in Sect. 2.3 to follow, the Euler equations for inviscid flows are a *first-order* system of non-linear coupled PDEs.

This thesis research considers the numerical solution of the one- and two-dimensional forms of the Euler equations of inviscid compressible gas dynamics with neglected external body forces, which can be expressed in the conserved form as

$$\frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla} \cdot \vec{\mathbf{F}} = 0, \quad (2.5)$$

where  $\mathbf{U}$  is the conserved variable solution vector,  $\vec{\mathbf{F}}$  is the solution flux dyad, and  $t$  is time.

For one space dimension, the Euler equations are given by

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0, \quad (2.6)$$

where  $\mathbf{U}$  has the form

$$\mathbf{U} = \left[ \rho, \rho v_x, \rho e \right]^T, \quad (2.7)$$

$x$  is the spatial coordinate,  $\rho$  is the gas density,  $v_x$  is the velocity of the gas,  $e = p/(\rho(\gamma - 1)) + v_x^2/2$  is the specific total energy,  $p = \rho RT$  is the pressure,  $T$  is the gas temperature,  $R$  is the gas constant,  $\gamma$  is the specific heat ratio, and

$$\mathbf{F} = [\rho v_x, \rho v_x^2 + p, v_x(\rho e + p)]^T, \quad (2.8)$$

is the solution flux vector. For a polytropic gas (thermally and calorically perfect gas), the ratio of specific heats,  $\gamma$ , is a constant and the specific heats are given by  $C_v = R/(\gamma - 1)$  and  $C_p = \gamma R/(\gamma - 1)$ .

Note that the system given by Eq. (2.6) has three dependent conserved variables, that are the gas density,  $\rho$ , the momentum in  $x$ -direction,  $m_x = \rho v_x$ , and the total energy,  $E = \rho e$ , which are function of the two independent variables, the spatial coordinate,  $x$ , and the time,  $t$ . The *non-conservative* form of Eq. (2.6) can be obtained by formulating this equation in terms of the primitive variable vector,  $\mathbf{W} = [\rho, v_x, p]^T$ , instead of the conserved one. Although the two formulations, conserved and non-conserved, are mathematically equivalent, the former formulation is better suited for *shock-capturing* methods whereas the latter may produce unsatisfactory results [115]. However, there are other important advantages in using primitive variables as part of the numerical procedure, including easier specification/implementation of physical boundary conditions and better control on the model *positivity*, which for Euler equations corresponds to positive pressure,  $p > 0$ , and density,  $\rho > 0$ , as will be discussed in the description of high-order numerical scheme to follow.

For 2D planar flows, the conserved form of the Euler equations are as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0, \quad (2.9)$$

where  $\mathbf{U}$  is given by

$$\mathbf{U} = \left[ \rho, \rho v_x, \rho v_y, \rho e \right]^T, \quad (2.10)$$

$x$  and  $y$  are the two spatial coordinates,  $v_x$  and  $v_y$  are the velocity components in the  $x$ - and  $y$ -coordinate directions,  $e = p/(\rho(\gamma - 1)) + (v_x^2 + v_y^2)/2$ , and

$$\begin{aligned} \mathbf{F} &= [\rho v_x, \rho v_x^2 + p, \rho v_x v_y, v_x(\rho e + p)]^T \\ \mathbf{G} &= [\rho v_y, \rho v_x v_y, \rho v_y^2 + p, v_y(\rho e + p)]^T, \end{aligned} \quad (2.11)$$

are the  $x$ - and  $y$ -direction solution flux vectors. The corresponding vector of primitive solution variables in this case is

$$\mathbf{W} = \left[ \rho, v_x, v_y, p \right]^T. \quad (2.12)$$

The mathematical properties of the Euler equations are analyzed and presented in detail in [15]. It can be shown that the *unsteady* Euler equations are hyperbolic in time, whereas

the mathematical behaviour of the steady equations (i.e., for  $\frac{\partial \mathbf{U}}{\partial t} = 0$ ) depend on a non-dimensional form of flow speed called *Mach number*,  $M$ , which is defined as

$$M = \frac{\|\vec{V}\|}{a} = \frac{\sqrt{v_x^2 + v_y^2}}{a},$$

where  $\|\vec{V}\|$  is the local velocity magnitude and  $a$  is the local sound speed given by

$$a = \frac{\partial p}{\partial \rho} = \sqrt{\frac{\gamma p}{\rho}} = \sqrt{\gamma RT}$$

for a polytropic gas. Thus, for subsonic Mach number,  $M < 1$ , the steady Euler equations are elliptic in space whereas they become hyperbolic for supersonic Mach,  $M > 1$  [119]. The location where the local velocity is equal to the local speed of sound, that is a Mach number of unity, is called a *sonic* or *critical point*. In the case of hyperbolic Euler equations, the information is propagated along the *characteristic lines* and there exists a limited domain of influence and dependence characteristic to each point in the computational domain, making this problem suitable for the application of a *space-marching* procedure [14]. In contrast, in the case of elliptic equations, information is propagated everywhere in all directions and consequently every point in the domain can influence and is influenced by any other point. Obviously, for such situations the space-marching methods are not suitable and the solution procedure must be carried out simultaneously for *all* interior and boundary points in the computational domain. A general approach to circumvent the problem of finding steady state solutions with both subsonic and supersonic regions is to solve the unsteady Euler equations, which are hyperbolic in time, with a time-marching method until all time derivatives become acceptably small, indicating that the steady-state has been reached.

## 2.3 Navier-Stokes Equations

As previously mentioned, the Navier-Stokes equations provide a more complete description of real fluids than Euler equations, as they account for diffusive transport phenomena such as viscous stresses and heat conduction. This mixed type system of conservation laws can be written in the form

$$\frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla} \cdot \vec{\mathbf{F}} = \frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla} \cdot \vec{\mathbf{F}}_H(\mathbf{U}) + \vec{\nabla} \cdot \vec{\mathbf{F}}_E(\mathbf{U}, \vec{\nabla} \mathbf{U}) = \mathbf{S}, \quad (2.13)$$

where, differently from Eq. (2.6) shown in the previous section, an arbitrary source term vector,  $\mathbf{S}$ , is introduced and the solution flux dyad,  $\vec{\mathbf{F}}$ , represents this time the sum of a hyperbolic term,  $\vec{\mathbf{F}}_H$ , which depends on the solution vector and accounts for convective transport phenomena and an elliptic term,  $\vec{\mathbf{F}}_E$ , associated with diffusion processes and therefore, dependent on both the solution and the solution gradient.

For two-dimensional planar flows, the Navier-Stokes equations are as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{F}_v}{\partial x} + \frac{\partial \mathbf{G}_v}{\partial y} = 0, \quad (2.14)$$

where  $\mathbf{U}$  is given by Eq. (2.10) and the  $x$ - and  $y$ -direction components of the hyperbolic solution flux vector,  $\vec{\mathbf{F}}_H(\mathbf{F}, \mathbf{G})$ , are the same as those defined for Euler equations in Eq. (2.11) and of the elliptic flux vector,  $\vec{\mathbf{F}}_E(\mathbf{F}_v, \mathbf{G}_v)$ , are

$$\begin{aligned} \mathbf{F}_v &= -[0, \tau_{xx}, \tau_{yx}, v_x \tau_{xx} + v_y \tau_{xy} - q_x]^T \\ \mathbf{G}_v &= -[0, \tau_{xy}, \tau_{yy}, v_x \tau_{yx} + v_y \tau_{yy} - q_y]^T, \end{aligned} \quad (2.15)$$

respectively. The components  $\tau_{xx}$ ,  $\tau_{yy}$  and  $\tau_{xy}$  of the molecular fluid stress tensor,  $\vec{\tau}$ , are function of the velocity gradients and the molecular fluid viscosity,  $\mu$ , and given by

$$\vec{\tau} = 2\mu(\vec{S} - \frac{1}{3}\vec{I}\vec{\nabla} \cdot \vec{V}), \quad \vec{S} = \vec{\nabla}\vec{V}, \quad (2.16)$$

where  $\vec{S}$  is the strain rate tensor. The molecular heat flux,  $\vec{q}(q_x, q_y)$ , is modelled using Fourier's law of conduction as  $\vec{q} = -\kappa\vec{\nabla}T$ , where  $\kappa$  is the thermal conductivity of the fluid.

An important similarity parameter that characterizes the regimes of viscous flows is the *Reynolds number* which is defined as the ratio of inertial to friction forces expressed as:

$$\text{Re} = \frac{\rho V d}{\mu} = \frac{V d}{\nu},$$

where  $V$  is the magnitude of the relevant velocity,  $d$  represents the characteristic length of interest and  $\nu$  is the gas kinematic viscosity. Thus, incompressible flows with equal Reynolds number are mechanically similar and this property can facilitate, among other things, the head-to-head comparison of experimental observations and numerical simulations. Note that in order to have mechanically-similar compressible flows both Mach and Reynolds numbers must be required to be the same. Moreover, Reynolds numbers are also used to differentiate between laminar and turbulent flow regimes [120].

Mathematically, the complete Navier-Stokes equations of Eq. (2.13) are a *second-order* system of non-linear coupled PDEs of hybrid nature, being parabolic-hyperbolic in time and space but becoming of mixed elliptic-hyperbolic type in space for the steady formulation. In spite of their widely application throughout engineering and natural sciences, a number of fundamental mathematical questions about solutions of the Navier-Stokes equations such as their existence, uniqueness and smoothness are yet unanswered and constitute grand challenges for the mathematics community. These issues are beyond the current work scope, but they are amply discussed by Temam [121] and very recently by Doering [122] in a survey on the current state of the mathematical theory and on the remaining open questions.

# Chapter 3

# HIGH-ORDER CENO FINITE-VOLUME SCHEME

*“Willst du ins Unendliche schreiten,  
Geh nur im Endlichen nach allen Seiten.”  
If you want to reach the infinite,  
Explore every aspect of the finite.*

by Johann W. von Goethe (1749-1832), German writer & polymath

The Finite Volume Method (FVM) has proved to be one of the most versatile discretization techniques used in CFD for solving the partial differential equations encountered in fluid flows [123]. A detailed presentation of this method is provided in the textbooks by LeVeque [123], Hirsch [124], Toro [115] and Lomax *et al.* [14]. In finite-volume methods the integral form of the conservation law is enforced discretely in each of the many contiguous control volumes (namely cells, elements, finite volumes) making up the computational domain. The most compelling feature of the FVM is that the resulting solution satisfies the conservation of conserved quantities such as mass, momentum, energy, and species. This important feature of the FVM generally makes it the method of choice for computing many compressible flows in which discontinuous solutions arise, as any discontinuity must satisfy the Rankine-Hugoniot or similar jump conditions [115]. Since finite volume methods are conservative they automatically satisfy the jump conditions and hence give physically correct weak solutions and ensure correct wave propagation speeds. Another advantage of the FVM is that, due to its formulation, it can be easily applied to both structured and unstructured meshes as well as hybrid grids. Despite the current widespread implementation of many variants of first and second-order FV algorithms in many research and commercial CFD packages (e.g., Clawpack [123], FUN3D [125, 126], Fluent [127]), the high-order counterpart of these

methods still represents an active area of research and is not widely available yet.

In the present work, a high-order central essentially non-oscillatory (CENO) cell-centred Godunov-type finite-volume scheme is proposed for solving the hyperbolic and elliptic systems of conservation laws described in Chapt. 2 on body-fitted multi-block mesh in conjunction with a block-based adaptive mesh refinement technique. This proposed variant of the original ENO scheme is not based on either selecting or weighting reconstructions from multiple stencils. Instead, a hybrid solution reconstruction procedure is used that combines the unlimited high-order  $k$ -exact least-squares reconstruction technique of Barth [16] based on a fixed central stencil with a monotonicity preserving limited piecewise linear least-squares reconstruction algorithm [16]. This hybrid approach avoids the complexity associated with other ENO and WENO schemes that require reconstruction on multiple stencils and provides an accurate and robust numerical algorithm.

This chapter presents a detailed description of the proposed high-order CENO finite-volume scheme as applied to single-block, body-fitted, quadrilateral mesh. Sections 3.1 and 3.2 summarize the key elements involved in the cell-centred FVM and its application to quadrilateral computational cells. Section 3.3 presents the high-order spatial discretization of the solution residual. The time-marching schemes used in this thesis research are described in Sect. 3.4. Details on the numerical evaluation of the hyperbolic and elliptic terms are provided in Sect. 3.5, 3.7 and 3.8. The hybrid CENO solution reconstruction is presented in Sect. 3.6 which fully describes the  $k$ -exact least-squares reconstruction technique, the solution monotonicity enforcement using the smoothness indicator and the boundary-condition prescription for straight and curved geometric boundaries. The extension of the CENO algorithm to multi-block structured meshes and its implementation in conjunction with a parallel block-based AMR strategy is considered in Chapt. 4.

## 3.1 Finite-Volume Method for Conservation Equations

The systems of conservation laws described in Chapt. 2 (Equations (2.1), (2.6), (2.9) and (2.14)) can all be re-expressed in vector form as

$$\frac{\partial \mathbf{U}}{\partial t} + \vec{\nabla} \cdot \vec{\mathbf{F}} = \mathbf{S}, \quad (3.1)$$

by making the appropriate substitutions for the conserved variable solution vector,  $\mathbf{U}$ , the flux dyad,  $\vec{\mathbf{F}}$ , and the non-linear source vector,  $\mathbf{S}$ . The advection-diffusion equation, Eq. (2.1), represents a degenerate form of Eq. (3.1) in the sense that the solution, the flux dyad and the source term vectors reduce to scalar quantities in this case.

The FVM used herein starts from the integral form of the Eq. (3.1) for an arbitrary fixed (i.e., time-invariant) control volume in which the volume integrals that contain divergence terms are converted to surface integrals using the divergence theorem to provide the form

$$\frac{d}{dt} \int_{\mathcal{V}} \mathbf{U} dv + \oint_{\Omega} \vec{\mathbf{n}} \cdot \vec{\mathbf{F}} d\Omega = \int_{\mathcal{V}} \mathbf{S} dv, \quad (3.2)$$

where  $\mathcal{V}$  is the control volume,  $\Omega$  is the closed surface of the control volume, and  $\vec{\mathbf{n}}$  is the unit outward vector normal to the closed surface. In two-space dimensions the control volume reduces to an area and the closed surface to a closed curve. Equation (3.2) is exactly satisfied for any control volume as well as for the whole computational domain regardless of the number of control volumes. It follows that even a coarse grid solution exhibits exact integral balances, because the flux entering a given volume is identical to that leaving the adjacent element. In other words, one cell's *loss* is another cell's *gain*.

Equation (3.2) can be further manipulated in terms of average values of solution variables, a procedure illustrated herein for a two-dimensional (2D) fixed control volume. Thus, the averaged value of the solution within a computational cell,  $\bar{\mathbf{U}}$ , can be defined by an integration over the control volume as follows:

$$\bar{\mathbf{U}} \equiv \frac{1}{A} \iint_{\mathcal{A}} \mathbf{U} da, \quad (3.3)$$

where  $A$  is the area of the two-dimensional domain  $\mathcal{A}$ . Substituting this definition into Eq. (3.2) and separating the spatially integrated terms, the final form of the integral conservation equation for a two-dimensional coordinate frame can be written as

$$\frac{d\bar{\mathbf{U}}}{dt} = -\frac{1}{A} \oint_{\Omega} \vec{\mathbf{F}} \cdot \vec{\mathbf{n}} d\ell + \frac{1}{A} \iint_{\mathcal{A}} \mathbf{S} da, \quad (3.4)$$

where  $d\ell$  is an element of the closed contour containing the control volume or cell of interest. Equation (3.4) describes the time evolution of the averaged conserved variables  $\bar{\mathbf{U}}$ , that are the unknowns or DOF of the FVM, due to the net contribution of the solution fluxes through the boundary of the computational cell and that of the surface sources tending to increase or

decrease the solution variables. Another important fact is that the use of cell-averaged data in Eq. (3.4) results in data that is discontinuous with that of neighbouring cells, allowing finite-volume schemes to naturally account for discontinuities (e.g., shocks) in the flow-field.

Despite the fact that Eq. (3.4) can be applied to any tessellation element, practical considerations regarding the calculation of the spatial integrals which are present on the *right-hand-side* of this equation suggest the use of simple geometric elements that are sufficiently flexible to allow descriptions of complex geometric boundaries. Common elements in 2D are triangles, quadrilaterals and hexagons. As the proposed high-order FVM is primarily developed for structured grids, that is a grid characterized by regular connectivity that can be expressed as a two or three dimensional array, the elements considered herein are *convex quadrilaterals* (i.e., a four-sided polygon with every internal angle less than 180 degrees) which represent the appropriate choice for a structured body-fitted mesh. Nevertheless, the proposed high-order algorithm can be extended relatively easily to other element types. The next section outlines some of the important features of the body-fitted structured mesh.

## 3.2 Body-Fitted Quadrilateral Grids

The two-dimensional structured multi-block body-fitted meshes used in this research represent a collection of single-block structured grids, each of which has the boundary geometry defined by four curved boundaries which are provided as piecewise polynomial splines of an order consistent with that of the finite-volume numerical scheme. Details regarding the interconnection between the building blocks of the multi-block grid can be found in Chapt. 4 while this section focuses on the structure of a single-block constituent of this grid.

An example of a regular single-block body-fitted quadrilateral structured mesh is shown in Fig. 3.1(a). An irregular mesh which takes advantage of the flexibility of quadrilaterals and somewhat resembles an unstructured mesh although still logically structured is illustrated in Fig. 3.1(b). The latter mesh has been obtained with an iterative algorithm that randomly perturbed the interior nodes of the regular mesh. Although the perturbed mesh does not present computational advantages and it most likely introduces discretization errors larger than those obtained on the regular mesh, it has been considered in this work when examining the robustness and accuracy of the high-order method on highly irregular meshes.

Representative quadrilateral elements of the type considered in this work are depicted

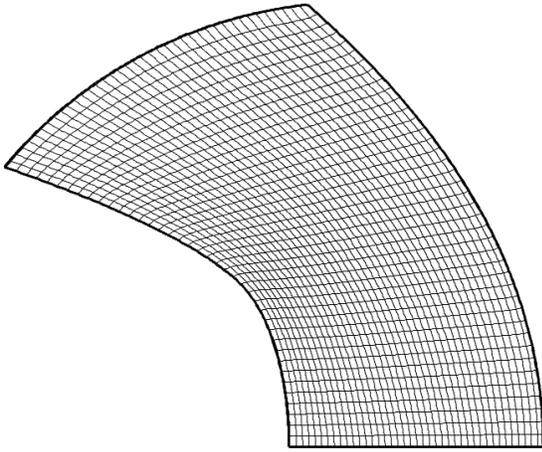
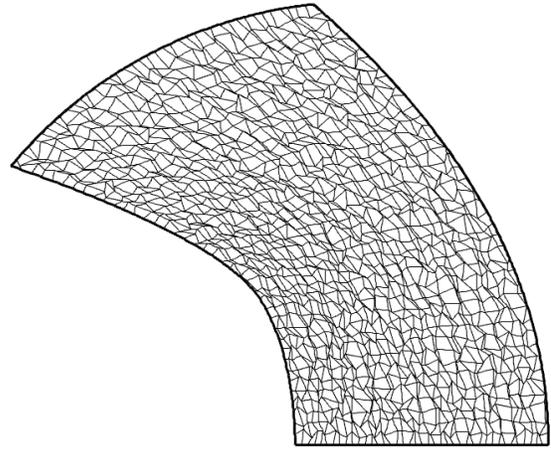
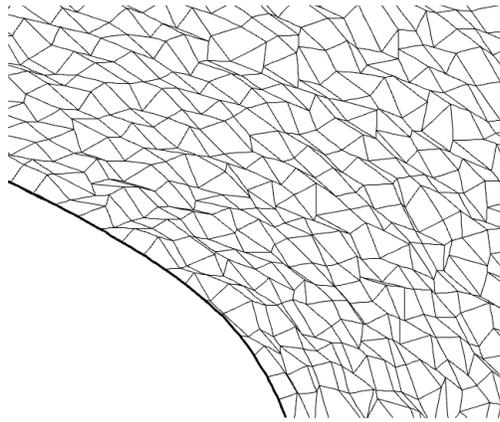
(a) *A regular grid with  $40 \times 40$  cells*(b) *An irregular grid obtained by disturbing the mesh shown in (a)*(c) *Detail of the irregular mesh*

Figure 3.1: *Illustration of two-dimensional body-fitted quadrilateral structured grids. Block boundaries are depicted with thick lines while thin lines show the interior mesh.*

in Fig. 3.2, in which each of the four faces is identified with the initials of the cardinal directions (i.e., North, South, East and West) based on the relative position with respect to the element centroid,  $\vec{X}_{i,j}$ . The four nodes of the computational element are specified by a combination of two cardinal directions. The two element types depicted in Fig. 3.2(a) and (b) are distinguished from one another by the connector,  $S$ , between two adjacent nodes, which can be either a straight or a curved line. Straight-edge quadrilaterals are always used for interior cells whereas quadrilaterals with curved edges may be required for representing

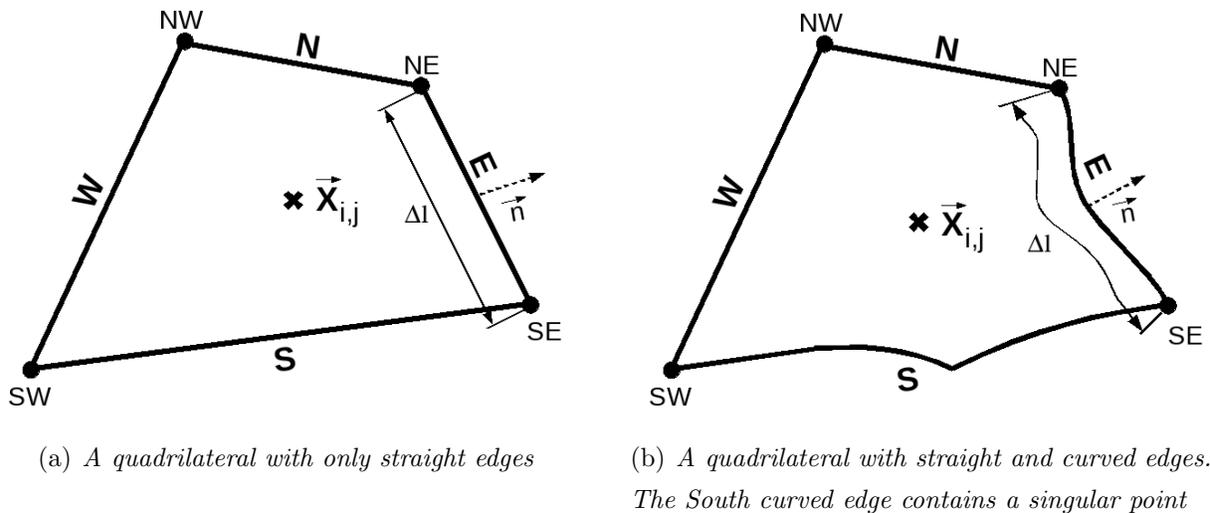


Figure 3.2: Schematic representation of convex quadrilateral elements considered in this work. Each face and node is identified by its corresponding cardinal direction (i.e., N, S, E, W) relative to the element centroid,  $\vec{X}_{i,j}$ . The figure also shows the unit outward vector normal,  $\vec{n}$ , at a given location and the length of a face,  $\Delta\ell$ .

the boundaries of the body-fitted mesh with high-order accuracy. Note that curved-edge quadrilaterals need not be used for schemes up to second-order even if the domain is bounded by curved boundaries. Note also that in any two-dimensional computational cell maximum two curved edges can occur and they need not be *smooth* curves. If any edge contains singular points as shown in Fig. 3.2(b), the control volume defined by the four mesh nodes, SW, SE, NE, NW, becomes a polygon with a total number of  $N_f$  faces which represents the sum of the straight-line and smoothly-curved segments. Computation of geometric properties of quadrilaterals relevant for the current work and the numerical integration of functions over such domains are described in detail in Appendix A.

### 3.3 Semi-Discrete Form

Applying the integral form of the conservation equation Eq. (3.4) to a 2D polygonal control volume,  $(i, j)$ , and using a  $N_G$ -point Gaussian quadrature numerical integration procedure to evaluate the solution flux along each of the  $N_f$  polygonal faces, the form of Eq. (3.4)

becomes

$$\frac{d\bar{\mathbf{U}}_{i,j}}{dt} = -\frac{1}{A_{i,j}} \sum_{l=1}^{N_f} \sum_{m=1}^{N_G} \left( \omega \vec{\mathbf{F}} \cdot \vec{n} \Delta\ell \right)_{i,j,l,m} + \bar{\mathbf{S}}_{i,j} = \mathbf{R}_{i,j}(\bar{\mathbf{U}}), \quad (3.5)$$

where the variable  $\bar{\mathbf{U}}_{i,j}$  is the conserved solution state for cell  $(i, j)$ ,  $A_{i,j}$  is the area of the cell,  $\omega$  is the quadrature weighting coefficient, and  $\Delta\ell$  and  $\vec{n}$  are the length of the cell face and unit vector normal to the cell face or edge, respectively. Each quadrilateral element has  $N_f=4$  unless it falls in the rare case when it contains curved edges with singular point. The average source term,  $\bar{\mathbf{S}}_{i,j}$ , is defined as

$$\bar{\mathbf{S}}_{i,j} = \frac{1}{A_{i,j}} \iint_{\mathcal{A}_{i,j}} \mathbf{S} da, \quad (3.6)$$

and the vector  $\mathbf{R}_{i,j}$  is the so-called residual vector operator.

Equation 3.5 represents a coupled system of non-linear, first-order *Ordinary Differential Equations* (ODE) which equates the time variation of the averaged conserved solution in cell  $(i, j)$  (the *left-hand-side* (LHS)) to the *spatial* solution residual (the *right-hand-side* (RHS)). This semi-discrete form of the integral equation, Eq. (3.4), forms the basis for devising both low- and high-order solution procedures, depending on the accuracy with which the LHS and RHS of Eq. (3.5) are resolved. A Godunov-type finite-volume algorithm for solving an initial or boundary value problem (IVP or BVP) given by Eq. (3.5) starts from a set of specified initial and boundary values of  $\bar{\mathbf{U}}$  and involves three major steps for obtaining the final solution states: 1) **reconstruction** in which an approximation to  $\mathbf{U}(\vec{X})$  is constructed; 2) **residual evaluation** which approximates the RHS of Eq. (3.5) based on the reconstruction function; and 3) **time integration**, in which the solution is advanced to the next time level based on the residual value. The remainder of this chapter presents in detail each of these steps in reverse order with an emphasis on the methodology used for obtaining a high-order accurate numerical procedure.

## 3.4 Explicit Temporal Discretization Methods

The time integration of Eq. (3.5) can be carried out with either implicit or explicit time-marching schemes. As noted in the introduction, the focus of this thesis research has been on the investigation of high-order spatial discretization procedures and little emphasis has been

given to temporal aspects of the FVM. Consequently, this work has been restricted to standard explicit time-integration schemes based on considerations of simplicity and availability. Future work will consider the integration of high-order CENO scheme with an implicit time-marching method. Note that combinations of high-order FVM and implicit time integration methods have been already proposed in the literature for inviscid compressible flows [31].

In this work, different explicit time-marching schemes are adopted to carry out the time integration of Eq. (3.5) depending on whether the problem to be solved represents a steady or unsteady calculation. For steady-state calculations the system of equations is marched until the transient part of the solution is removed and

$$\mathbf{R}_{i,j}(\bar{\mathbf{U}}) = 0. \quad (3.7)$$

It is easily noted that the accuracy of the solution in these situations is entirely dependent on the accuracy of the spatial residual operator and any stable time-marching scheme can be used to obtain solutions of any accuracy order. In particular, the explicit optimally-smoothing multi-stage scheme developed by van Leer *et al.* [128, 129] has been used to obtain the solution for all the steady-state test cases considered in this work. It is noted however that these schemes were designed to provide optimal damping of the high frequency content of the solution when used with a lower-order (i.e., up to second-order) upwind scheme. It is also noted that the number of iterations required to remove the transient part in a steady-state calculation is greatly determined by how effective a time-marching scheme is in damping the high-frequency content of the solution. This aspect has important implications on the computational effort required by a high-order spatial algorithm to reach the steady-state solution as the high-order discretization is designed specifically to resolve the high-frequency solution content. To put it in perspective, it will take a lot more computational effort for a high-order algorithm to reach a steady-state solution starting from a rough initial guess as compared to that of a low-order method on the same mesh resolution, on one hand due to a higher computational cost per node of the first relative to the latter and on the other hand due to reduced dissipation and their ability to preserve the transients longer in the iterative procedure. A better strategy, used frequently over the course of this research for steady-state calculations, is to obtain first a better initial guess with a low-order spatial discretization and then to switch to a high-order algorithm for resolving the fine solution details.

On the other hand, for time-accurate calculations it is deemed desirable to integrate

the set of ODEs given by Eq. (3.5) with a time-marching method that is at least of the same order of accuracy as that of the spatial discretization scheme in order to obtain a consistent high-order discretization procedure. Consequently, for time-accurate problems, either a two- or four-stage standard Runge-Kutta scheme is used, depending on the accuracy of the spatial reconstruction. The details and analysis of these well-known time-marching schemes are omitted from this thesis as there are several comprehensive textbooks on the subject [14, 15, 124].

The maximum allowable time-step for the stability of the time-marching schemes used herein is provided by

$$\Delta t = \text{CFL} \min [\tau_h, \tau_e, \tau_s] \quad (3.8)$$

where  $\tau_h$ ,  $\tau_e$  and  $\tau_s$  correspond to the stability criterion for the hyperbolic, the elliptic and the source term, respectively. The Courant-Friedrichs-Lewy number, CFL, is used to ensure stability of the time-marching scheme and typically, values of less than unity are used herein. The relationships for estimating the stability criterion for each term depend on the studied system of equations. Hence, for the advection-diffusion equation the stability criteria take the particular form

$$\tau_h = \frac{\Delta x}{|V|_{\max}}, \quad |V| > 0; \quad \tau_e = \frac{1}{2} \frac{\Delta x^2}{\kappa_{\max}}, \quad (3.9)$$

where  $\Delta x$  is the estimated length of a mesh computational cell and  $|V|_{\max}$  and  $\kappa_{\max}$  are the largest velocity and diffusion coefficient occurring in that cell. The source stability criterion,  $\tau_s$ , is defined differently for each particular form of the source term. Similarly, the stability criteria used for the Navier-Stokes equations are

$$\tau_h = \frac{\Delta x}{(|V| + a)_{\max}}; \quad \tau_e = \frac{1}{2} \frac{\Delta x^2}{\nu_{\max}}, \quad (3.10)$$

where the largest velocity magnitude has been replaced by the largest acoustic wavespeed and the diffusion coefficient by the gas kinematic viscosity. The Euler equations require only the criterion for the hyperbolic term in Eq. (3.10). Moreover, no restrictions are imposed by  $\tau_s$  when solving Euler and Navier-Stokes equations as no sources have been considered with these equation sets.

As a final note, it should be mentioned that for the unsteady problems in which a comparison between a high-order and a low-order method has been carried out in this thesis, the order of the time-marching scheme for the low-order spatial discretization was chosen based

on the objective of the analysis. Hence, for problems in which the accuracy comparison was the main concern both high-order and low-order simulations were carried out with the same time accurate method and CFL-number. On the other hand, for problems in which the computational cost for a given accuracy was analyzed, the second-order Runge-Kutta method having the same CFL-number as that of the fourth-order one was used for the low-order spatial discretization which provides a consistent scheme in the space-time domain. Although a fourth-order Runge-Kutta scheme may reduce the error for a given spatial resolution, it would also increase the computational cost of the low-order scheme and thus may result in an unfair comparison. One could also argue that the opposite may be true and the low-order scheme in space may be penalized by the lack of accuracy in time and a better option would be to use a fourth-order time-marching method even for the low-order spatial discretization. As the reader can easily spot there is a fine line here and the best choice depends most likely on the problem at hand. However, as long as the leading error is due to the spatial discretization the two approaches will produce very similar error results and the choice should be based solely on computational performance reasons.

### 3.5 Numerical Residual Evaluation

High-order accurate solutions of Eq. (3.5) are sought here in two space dimensions by applying a high-order spatial discretization to the solution residual vector,  $\mathbf{R}_{i,j}$ , in conjunction with high-order polynomial solution reconstruction, upwind discretization of the hyperbolic flux and centrally weighting discretization of the elliptic flux. Thus, the high-order CENO numerical procedure requires the computation with high-order accuracy of both the contour and surface integrals making up the spatial residual while ensuring the stability of the numerical scheme. The computation of these two integrals is explained next, starting with the surface integration of the source term.

There are two situations which arise in the computation of the average source term,  $\bar{\mathbf{S}}_{i,j}$ , defined by Eq. (3.6). The first possibility is that the particular form of the source term can be integrated analytically and the resulting expression is a function only of the space geometry and/or the average solution data. In such situations, the average source term is computed accurately by simply providing the average solution and geometry data with the appropriate order of accuracy. However, analytical integration is limited to simple source

term expressions in which the solution dependency is at most linear. Therefore, a high-order accurate calculation of the average source term for non-linear expressions relies in general on numerical integration of an appropriate order. This work considers only smooth non-linear source terms and as such, the quadrature integration techniques described in Appendix A for convex quadrilaterals are applied. Moreover, the source term expression must be accurately estimated at each integration point in order to obtain a high-order accurate  $\bar{\mathbf{S}}_{i,j}$ , which implies in turn that a high-order representation of the solution is required for integrating non-linear expressions of the form  $\mathbf{S} = \mathbf{S}(\mathbf{U}(\vec{X}))$ . In the FVM described herein the high-order polynomial solution reconstruction is used for this purpose. Finally, it should be mentioned that either large uncertainty and/or high computational cost related to the calculation of the source term may render useless or impractical a high-order quadrature integration, in which case a second-order approximation (i.e., the source term evaluated at the cell centroid) would suffice for practical computations. Such situations could arise in combustion simulations in which evaluation of chemistry source terms is a very computationally expensive operation.

The high-order accurate numerical computation of the net flux through the boundary of a computational cell starts with the selection in Eq. (3.5) of a number of Gauss quadrature points,  $N_G$ , dictated by the desired order of solution accuracy at which the numerical flux,  $\vec{\mathbf{F}} \cdot \vec{\mathbf{n}}$ , is evaluated. Clearly, the approximate quadrature rule must be of sufficient order so as to generate the desired solution accuracy while not being overly accurate to create a waste of computer resources. Note that just increasing the order of the quadrature rule without computing the flux function more accurately at each integration point can only result in increased accuracy (i.e., a lower *absolute value* of the error) and not in high-order accuracy (i.e., an increase in the *order* of solution accuracy) [28]. Hence, for high-order accuracy the numerical flux representing the sum of the non-linear hyperbolic and elliptic fluxes,  $\vec{\mathbf{F}} \cdot \vec{\mathbf{n}} = \vec{\mathbf{F}}_H(\mathbf{U}) \cdot \vec{\mathbf{n}} + \vec{\mathbf{F}}_E(\mathbf{U}, \vec{\nabla}\mathbf{U}) \cdot \vec{\mathbf{n}}$ , needs to be estimated at each quadrature point,  $m$ , of a cell face,  $l$ , with a flux function that approaches the *true* flux at the rate imposed by the expected order of accuracy in the asymptotic limit of infinitely small mesh size. This requirement translates to a similar set of conditions on the accuracy of the solution state,  $\mathbf{U}$ , and solution gradient,  $\vec{\nabla}\mathbf{U}$ , used to calculate both the hyperbolic,  $\vec{\mathbf{F}}_H$ , and elliptic,  $\vec{\mathbf{F}}_E$ , fluxes. Other than this, the flux function is required to produce a dissipative and stable scheme.

In this work, the hyperbolic flux at each inter-cellular face is determined using a high-

order upwind formulation, originally pioneered by Godunov for linear and the inviscid gas-dynamics equations in 1959 [130]. In this landmark paper, Godunov proposes a monotone scheme by introducing information about the propagation of physical properties of the governing equations into the numerical discretization to prevent the growth of unwanted solution oscillations in the vicinity of sharp solution variations (e.g., shocks). For hyperbolic equations, this procedure is equivalent to using information about the propagation of the solution content along the characteristic lines in the space-time domain. Consequently, the procedure for computing the hyperbolic flux at a cell interface is based on the wave structure emerging at an interface with discontinuous solution data, which is equivalent to solving a local Riemann problem with the left and right solution states,  $\mathbf{U}_l$  and  $\mathbf{U}_r$ , as initial data [115]. Thus, the hyperbolic numerical flux,  $\vec{\mathbf{F}}_H \cdot \vec{n}$ , at the quadrature points is given by

$$\vec{\mathbf{F}}_H \cdot \vec{n} = \mathcal{F}(\mathbf{U}_l, \mathbf{U}_r, \vec{n}), \quad (3.11)$$

where the numerical flux  $\mathcal{F}$  is evaluated by solving the Riemann problem in a direction defined by the normal to the face with  $\mathbf{U}_l$  and  $\mathbf{U}_r$ . In the present algorithm, both exact and approximate Riemann solvers can be used to solve the Riemann problem and evaluate the numerical flux. Details of the flux functions considered in this thesis for each of the governing equation sets described in Chapt. 2 are given in Sect. 3.7.

The left and right solution states,  $\mathbf{U}_l$  and  $\mathbf{U}_r$ , are determined by performing piecewise  $k$ -order polynomial solution reconstruction within each computational cell, which obviously makes the spatial accuracy of the finite-volume scheme very dependent on the order of the solution reconstruction. Since the truncation error for  $k$ -order exact reconstruction is  $\mathcal{O}(\Delta x^{k+1})$ , a  $(k+1)$ -order accurate spatial discretization can be achieved based on this reconstruction for smooth hyperbolic problems. Consequently, the number of Gauss quadrature points at which the numerical flux is evaluated is chosen on the basis of the order of the solution reconstruction. To determine the optimum number of quadrature points for a desired accuracy recall that an  $N$ -point Gaussian quadrature rule integrates polynomials of degree  $2N - 1$  exactly, providing a  $2N$ -order accurate formula [131]. Following this rule, the practise adopted here is to use one quadrature point ( $N_G = 1$ ) for second-order schemes (piecewise linear,  $k = 1$ , reconstruction) and two quadrature points ( $N_G = 2$ ) for third- and fourth-order schemes (piecewise quadratic,  $k = 2$ , and cubic,  $k = 3$ , reconstruction) to ensure that the order of accuracy of the schemes is preserved. Ample details of the polynomial

solution reconstruction considered in this work are provided in Sect. 3.6.

In a similar manner to hyperbolic fluxes, numerical elliptic fluxes,  $\vec{\mathbf{F}}_E \cdot \vec{n}$ , must be evaluated at each quadrature point of each computational cell face. High-order elliptic fluxes can be calculated by using information from the  $(k + 1)$ -order accurate solution approximation obtained in the reconstruction step. Note that a  $(k + 1)$ -order accurate solution reconstruction will usually produce a  $k$ -order accurate gradient (i.e., one order less due to differential operation), which should in turn lead to a  $k$ -order accurate pointwise flux evaluation and finally, a  $k$ -order accurate flux integral. It is thus inferred that a  $(k+1)$ -order accurate spatial discretization scheme for *both* hyperbolic and elliptic terms on arbitrary meshes, implies the use of a  $k$ -order exact gradient (i.e., a gradient with  $(k+1)$ -order truncation error) for the evaluation of elliptic fluxes, which here is derived from a  $(k+1)$ -order exact reconstruction. Thus, having determined the left and right  $(k+1)$ -order exact piecewise solution reconstructions,  $U_l^{k+1}(\vec{X})$  and  $U_r^{k+1}(\vec{X})$ , a unique  $(k+1)$ -order accurate solution gradient at the inter-cellular face is obtained as the arithmetic mean of the left and right reconstruction gradients. Note that, the arithmetic mean procedure may provide a  $(k+1)$ -order accurate gradient from two  $k$ -order exact reconstructions if error cancellation occurs, a situation which arises only for odd reconstruction orders and for generally regular meshes. Note also that, as previously stated, a  $(k+1)$ -order exact solution reconstruction can provide a  $(k+2)$ -order accurate spatial discretization of the hyperbolic term if the optimum number of Gauss integration points is used, but only a  $(k+1)$ -order accurate discretization of the elliptic term under the same conditions. For this reason, even if a piecewise quartic ( $k=4$ ) reconstruction has the potential to generate a 5th-order discretization of the hyperbolic flux in combination with at least three quadrature points ( $N_G = 3$ ), it will only provide a 4th-order accurate discretization of the elliptic flux. As the proposed residual discretization seeks to obtain a global  $(k + 1)$ -order accurate scheme on arbitrary meshes, this condition is achieved herein by carrying out a  $(k+1)$ -order exact solution reconstruction to ensure a  $k$ -order exact solution gradient and by choosing the number of quadrature points which are optimum for  $k$ -order exact solution reconstruction. Consequently, a consistent 4th-order accurate scheme for *both* hyperbolic and elliptic operators is formed here with a piecewise quartic reconstruction ( $k=4$ ) and two quadrature points ( $N_G = 2$ ). More details about the high-order discretization of the elliptic operator are provided in Sect. 3.8.

## 3.6 CENO Reconstruction

The main purpose of the reconstruction process in a FVM is to compute in each computational cell a piecewise polynomial representation of the solution that can provide accurate values for calculating the numerical flux at each Gauss quadrature point. The reconstruction operation is a strictly-mathematical procedure in which no physical characteristics of the problem are directly included and depends solely on the average solution states provided as input data. Consequently, it can be equally applied to any of the independent solution variables regardless of their physical meaning and of whether they represent conserved, primitive or characteristic quantities. Note that in our computational framework for multivariate PDEs we prefer to perform the reconstruction with the primitive solution states whereas the solution update is carried out with the conserved variables. To map the average conserved into average primitive states and vice-versa our algorithm applies the same relationships as those used for pointwise mapping. An exact conversion between the two types of average quantities cannot be obtained for non-linear relationships with this approach. Very recently, McCorquodale and Colella [35] have proposed a transformation which provides a fourth-order accurate conversion between the aforementioned solution variables. Although no corrections have been considered herein, it seems that the error introduced by the conversion of the average quantities based on pointwise mapping does not seem to affect the order of accuracy of the high-order scheme, a fact demonstrated in Sect. 5.6.3 by numerical experiments with both sets of variables.

As previously explained, the accuracy of the reconstruction procedure has direct implications on the overall numerical accuracy of the scheme and any attempt to formulate a high-order FV numerical procedure should start with the reconstruction step. In general, the functions considered to describe the solution variations are chosen as piecewise polynomials of an order dictated by the desired accuracy for the numerical scheme. In Godunov-type finite-volume schemes, the task at hand in the reconstruction step is to compute the polynomial coefficients in each computational cell starting from the integral cell averages. In general, this is the most difficult task in the solution process and for many governing equation sets it represents the most computationally-expensive operation, depending on the efficiency of the numerical flux function and the complexity of the source terms. This is particularly true for the classical Euler and Navier-Stokes equations. Consequently, the efficient design of

the reconstruction procedure can have significant impact on the computational performance of the numerical scheme. Moreover, its impact grows in importance with the increase in the order of accuracy as high-order reconstructions can quickly become more complex and expensive than their low-order counterparts.

A high-order central ENO (CENO) method is used herein for performing the piecewise  $k$ -order polynomial reconstruction within each of the computational cells. Before proceeding further with a description of CENO method, we shall re-emphasize several disadvantages of other ENO and WENO schemes that motivated the search for a new ENO variant which is able to overcome some of these drawbacks. As mentioned in Sect. 1.3.1, the original ENO reconstruction proposed by Harten *et al.* [18] constitutes a robust framework for obtaining uniform accurate high-order discretizations of any order. However, the idea of selecting the “smoothest” reconstruction in some norm encounters tremendous practical issues when extended to multi-dimensional problems with large numbers of unknowns. In particular, the challenges are associated with the stencil selection algorithm and the large computational cost resulting from carrying out multiple reconstructions on different stencils with a different stencil for each solution variable. The stencil switching behaviour encountered during a simulation can also prevent some algorithm optimizations that are possible with a fixed stencil. Additionally, ENO schemes require reconstruction with characteristic variables to obtain accurate results when discontinuous wave interactions occur in the simulation [18]. While WENO schemes have solved some of the problems associated with ENO methods, such as convergence to steady-state solutions, they still suffer from the necessity of performing reconstructions on multiple stencils, with different stencils for each variable, and may employ extra algorithms for dealing with singular stencils [26] and occurrence of negative weights [132], especially on unstructured meshes. In addition, the weights may require some tuning such that the central stencil, recognized to be the most accurate one, is recovered in the smooth parts of the solution [133, 134]. A step forward in the direction of reducing the number of reconstructions to a maximum of two has been taken by the DD-ENO procedure [29, 30]. Unfortunately, in addition to requiring two reconstructions everywhere in the solution domain this method can also fail in some situations to assign sufficiently small weights to the non-smooth data which can result in non-monotone solutions. It should be also mentioned that in some situations the original ENO-type reconstructions may lead to negative values of density and pressure due to their design principle (i.e., selection of the

“smoothest” stencil) which does not strictly enforce solution monotonicity. This aspect has been pointed out in the literature [20, 135]. Despite all these drawbacks, both ENO and WENO schemes provide a solid starting point for the search for more effective high-order discretizations.

The proposed CENO variant of the original ENO scheme is not based on either selecting or weighting reconstructions from multiple stencils. Instead, a hybrid solution reconstruction procedure is used that combines the high-order  $k$ -exact least-squares reconstruction technique of Barth [16] based on a *fixed central stencil* with a monotonicity preserving limited piecewise linear least-squares reconstruction algorithm [16]. Due to cancellation of truncation errors, use of the central stencil will generally provide the most accurate reconstruction. In the case of unstructured mesh, the *central stencil* should be interpreted as a stencil that includes all nearest neighbour cells up to a specified order. Unfortunately, the resulting  $k$ -exact reconstruction from a central stencil near under-resolved solution content, sharp solution gradients and/or discontinuities exhibits undesirable solution oscillations (wiggles) as illustrated in Fig. 1.4, which can deteriorate solution accuracy or generate non-physical solution values. Therefore, in the CENO approach a limited reconstruction procedure is applied to computational cells with under-resolved solution content and the unlimited  $k$ -exact reconstruction scheme is used for cells in which the solution is fully resolved. Switching in the hybrid procedure is determined by a solution smoothness indicator that indicates whether or not the solution is resolved on the computational mesh. This hybrid approach avoids the complexity associated with other ENO and WENO schemes that require reconstruction on multiple stencils which in some cases can produce poorly conditioned coefficient matrices. Additionally, the same fixed stencil is used for each variable and the solution of the least-squares problem for the reconstruction can then be made quite efficient. For this reason, the hybrid CENO algorithm would seem very well suited for solution reconstruction on unstructured mesh. Another benefit of the CENO approach is that mesh adaptation can be directed based on the ability of the scheme to differentiate between resolved and under-resolved or non-smooth solution content.

The CENO reconstruction leads to a finite-volume scheme for hyperbolic conservation equations that is high-order accurate for smooth solutions even near extrema and avoids the appearance of  $O(1)$  numerical oscillations in under-resolved regions and for solutions containing strong discontinuities and/or shocks. Note however that in the proposed formulation the

formal ENO property of uniform accuracy is lost for non-smooth solutions. Accuracy to any order is possible by simply expanding the support for the cell-centred reconstruction. Note that in earlier work, Harten and Chakravarthy [19] have proposed a technique to obtain an ENO reconstruction on a fixed central stencil by hybridizing the high-order reconstruction with a first-order formulation. The switching in their proposed hybrid central ENO scheme was based on undivided differences and the total variation diminishing (TVD) property [136] and not directly on the smoothness of the reconstructions. More recently, Haselbacher [26] has since explored the use of fixed stencil central reconstruction in the formulation of WENO schemes for unstructured mesh, but Haselbacher's approach is somewhat different to the current approach and schemes of accuracy higher than second order (piecewise linear reconstruction) were not formulated.

In what follows, a detailed summary of the proposed high-order CENO reconstruction is provided, including discussions of the  $k$ -exact least-squares reconstruction, monotonicity enforcement, solution smoothness indicator and treatment of high-order boundary conditions.

### 3.6.1 $k$ -Exact Least-Squares Reconstruction

In piecewise  $k$ -exact polynomial reconstruction, it is assumed that a solution variable,  $u$ , at any location,  $\vec{X}$ , in computational cell  $(i, j)$  has the general form

$$u_{i,j}^k(\vec{X}) = \sum_{\substack{p_1=0 \\ (p_1+p_2 \leq k)}}^k \sum_{p_2=0}^k (x - \bar{x}_{i,j})^{p_1} (y - \bar{y}_{i,j})^{p_2} D_{p_1 p_2}^k, \quad (3.12)$$

where  $(x, y)$  are the Cartesian coordinates of the position vector,  $\vec{X}$ , at the point of interest,  $(\bar{x}_{i,j}, \bar{y}_{i,j})$  are the coordinates of the cell centroid  $\vec{X}_{i,j}$ ,  $k$  is the order of the piecewise polynomial interpolant, the summation indices  $p_1$  and  $p_2$  must satisfy the condition that  $p_1 + p_2 \leq k$ , and  $D_{p_1 p_2}^k$  are the coefficients of the  $k$ -exact polynomial approximation to be determined. The latter are in general functions of the mean or average value solution,  $\bar{u}_{i,j}$ , within the cell and its neighbours. In general, the number of coefficients,  $\mathcal{N}_D$ , for a particular order  $k$  and one solution variable is determined with the following relation

$$\mathcal{N}_D = \frac{1}{d!} \prod_{n=1}^d (k + n), \quad (3.13)$$

where  $d$  represents the number of space dimensions. For 2D problems  $d=2$  and  $\mathcal{N}_D$  becomes

$$\mathcal{N}_D = \frac{(k+1)(k+2)}{2} . \quad (3.14)$$

To determine the coefficients,  $D_{p_1 p_2}^k$ , the following conditions are required to be satisfied by the reconstruction procedure:

- i) the solution reconstruction must reproduce exactly polynomials of degree  $N \leq k$ ;
- ii) the solution reconstruction must preserve the mean or average value within the computational cell;
- iii) the reconstruction procedure must have compact support.

The first condition is equivalent to

$$u_{i,j}^k(\vec{X} - \vec{X}_{i,j}) - u_{\text{exact}}(\vec{X}) = \mathcal{O}(\Delta x^{k+1}) , \quad (3.15)$$

which is assumed to hold anywhere in the vicinity of cell  $(i, j)$ . The  $k$ -exactness property is also motivated by Barth [137] with the fact that for sufficiently smooth solutions it ensures that the difference between the predicted solution states at inter-cell boundaries diminishes with increasing  $k$  at a rate proportional to  $h^{k+1}$  where  $h$  is a maximum diameter of the two cells. As we shall see in Sect. 3.7, this property has direct implications on the amount of dissipation introduced by the Riemann solver. The second condition requires the integral of the piecewise polynomial approximation to recover the cell average data, which gives

$$\bar{u}_{i,j} = \frac{1}{A_{i,j}} \iint_{\mathcal{A}_{i,j}} u_{i,j}^k(\vec{X}) dx dy . \quad (3.16)$$

Finally, the third condition involves the number and locality of the neighbouring solution states used in the cell reconstruction.

The evaluation of the coefficients,  $D_{p_1 p_2}^k$ , requires the Eq. (3.16) to be supplemented by at least  $\mathcal{N}_D - 1$  additional equations which are determined as follows. Based on the relation expressed by Eq. (3.15), the mean preservation condition is extended to the neighbourhood of the reconstructed cell and similar relations to Eq. (3.16) are written for a set of neighbours which together with the reconstructed cell form the so-called *reconstruction supporting stencil*. For brevity, the cell in which the reconstruction is performed will be denoted by index  $I$  and any of the neighbours in the supporting stencil by index  $J$ .

The minimum size of the compact stencil is determined by the number of unknown coefficients,  $\mathcal{N}_D$ , but in practise, additional neighbours are included to make the reconstruction more robust in the presence of stretched meshes and solution gradients not aligned with the mesh. For body-fitted quadrilateral mesh, the current  $k$ -exact reconstruction scheme uses a symmetric fixed central stencil which includes 8 neighbour cells for  $k=1$  and 24 neighbours for  $k=2, 3$  and 4. This particular choice ensures the homogeneity of the reconstruction operator but can also incur significant overdetermination for some reconstruction orders. Thus, the most disadvantageous situation occurs for the quadratic reconstruction which includes about four times more cells than the minimum required for  $\mathcal{N}_D = 6$ . However, this ratio becomes 2.5 and only 1.67 for cubic ( $k=3$ ) and quartic ( $k=4$ ) reconstruction, respectively.

To compute the polynomial coefficients with an overdetermined supporting stencil, we seek to minimize the error,  $E_J$ , in predicting the mean value in each  $J$  neighbour, written as

$$E_J = \frac{1}{A_J} \iint_{\mathcal{A}_J} u_{i,j}^k(\vec{X} - \vec{X}_{i,j}) dx dy - \bar{u}_J . \quad (3.17)$$

The evaluation of the coefficients  $D_{p_1 p_2}^k$  requires the least-squares solution of an overdetermined system of linear equations  $\mathbf{A} \mathbf{x} - \mathbf{B} = \mathbf{E}$ , where the coefficient matrix,  $\mathbf{A}$ , of the linear system depends only on the mesh geometry and can be partially calculated in a preprocessing step. The average solution data at each time step is contained in the matrix,  $\mathbf{B}$ , and the mean value error in each control volume is in matrix,  $\mathbf{E}$ , which has the norm minimized in the least-squares sense. Depending on how many solution variables are present in the discretization procedure, the matrices  $\mathbf{x}$ ,  $\mathbf{B}$  and  $\mathbf{E}$  can have either a single column if there is only one solution variable or more columns for multiple solution variables with each column belonging to one solution unknown.

The preservation of the average value  $\bar{u}_I$  within the reconstructed cell is explicitly enforced by expressing the coefficient,  $D_{00}^k$ , as a function of the other unknowns as

$$D_{00}^k = \bar{u}_I - \sum_{\substack{p_1=0 \\ (p_1+p_2 \neq 0)}}^k \sum_{p_2=0}^k D_{p_1 p_2}^k (\overline{x^{p_1} y^{p_2}})_I , \quad (3.18)$$

where the term  $(\overline{x^{p_1} y^{p_2}})_I$  is defined below. Thus, the linear system that needs to be solved

for all unknowns except  $D_{00}^k$  has the form

$$\underbrace{\begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_J \\ \vdots \\ L_{N_n} \end{bmatrix}}_A \underbrace{\begin{pmatrix} D_{01} \\ D_{02} \\ \vdots \\ D_{p_1 p_2} \\ \vdots \end{pmatrix}}_x - \underbrace{\begin{pmatrix} w_1(\bar{u}_1 - \bar{u}_I) \\ w_2(\bar{u}_2 - \bar{u}_I) \\ \vdots \\ w_J(\bar{u}_J - \bar{u}_I) \\ \vdots \\ w_{N_n}(\bar{u}_{N_n} - \bar{u}_I) \end{pmatrix}}_B = \underbrace{\begin{pmatrix} E_1 \\ E_2 \\ \vdots \\ E_J \\ \vdots \\ E_{N_n} \end{pmatrix}}_E \quad (3.19)$$

where  $N_n$  is the number of neighbours in the supporting stencil and the generic row  $L_J$  in the coefficient matrix,  $\mathbf{A}$ , of the linear system, Eq. (3.19), is given by

$$L_J = \left( w_J \left( \widehat{x^0 y^1} \right)_{IJ} \quad w_J \left( \widehat{x^0 y^2} \right)_{IJ} \quad \dots \quad w_J \left( \widehat{x^{p_1} y^{p_2}} \right)_{IJ} \quad \dots \right), \quad (3.20)$$

where the geometric weight  $w_J$  is specific to each control volume  $J$  and serves the purpose of improving the locality of the reconstruction. The matrix coefficient  $\left( \widehat{x^{p_1} y^{p_2}} \right)_{IJ}$  depends only on the mesh geometry of  $I$  and  $J$  cells and has the expression

$$\left( \widehat{x^{p_1} y^{p_2}} \right)_{IJ} = \left( \widetilde{x^{p_1} y^{p_2}} \right)_{IJ} - \left( \overline{x^{p_1} y^{p_2}} \right)_I, \quad (3.21)$$

where  $\left( \overline{x^{p_1} y^{p_2}} \right)_I$  represents the geometric moment of powers  $(p_1, p_2)$  of control volume  $I$  (i.e., the reconstructed cell) about its own centroid and is given by

$$\left( \overline{x^{p_1} y^{p_2}} \right)_I = \frac{1}{A_I} \iint_{\mathcal{A}_I} (x - \bar{x}_{i,j})^{p_1} (y - \bar{y}_{i,j})^{p_2} dx dy, \quad (3.22)$$

and  $\left( \widetilde{x^{p_1} y^{p_2}} \right)_{IJ}$  represents the geometric moment of powers  $(p_1, p_2)$  of the neighbour  $J$  about the centroid of the reconstructed cell  $I$ . Thus, the geometric moment  $\left( \widetilde{x^{p_1} y^{p_2}} \right)_{IJ}$  is obtained by integrating over the domain  $\mathcal{A}_J$  of a given neighbour  $J$  the geometrically-dependent monomial expression in Eq. (3.12) as follows

$$\left( \widetilde{x^{p_1} y^{p_2}} \right)_{IJ} = \frac{1}{A_J} \iint_{\mathcal{A}_J} (x - \bar{x}_{i,j})^{p_1} (y - \bar{y}_{i,j})^{p_2} dx dy. \quad (3.23)$$

Details of the procedure used in the computation of the geometric moments given by Eq. (3.22) and Eq. (3.23), including discussion about memory requirements in case of stored data, can be found in Appendix A and B.

As previously mentioned, geometric weights of the form presented below are assigned to each control volume in order to have a more localized reconstruction

$$w_J = \frac{1}{|\vec{X}_J - \vec{X}_{i,j}|^\theta}, \quad (3.24)$$

where  $\vec{X}_J$  is the centroid of neighbour cell  $J$ , and values considered herein for the exponent  $\theta$  are either one or two. The importance of distance weighting in the least-squares reconstruction for stretched meshes with surface curvature has been investigated by Mavriplis in [138]. In general, the inverse squared distance weighting has been found to be more accurate. However, the inverse distance was still used for certain problems.

Both Householder QR factorization algorithm [139] and orthogonal decomposition by SVD method [139] can be used to solve the weighted least squares problem represented by Eq. (3.19) and determine the polynomial coefficients for all solution variables concurrently, the latter being favoured for the computation of the left pseudo-inverse matrix  $\mathbf{A}^{-1}$  [140] as described below. Note that use of a fixed central stencil here avoids the complexities associated with ENO and WENO schemes that require reconstruction on multiple stencils as well as many of the difficulties associated with poorly conditioned coefficient matrices for the least squares problems. In addition, the use of a fixed stencil allows the pseudo-inverse matrix  $\mathbf{A}^{-1}$  to be stored and reused for the calculation of *all* solution variables at successive time steps as long as the computational grid is not modified. In particular, it was found that such a procedure can provide a computational speed-up by a factor of about 5 for high-order two-dimensional Euler computations compared with the situation in which the coefficient matrix,  $\mathbf{A}$ , is formed and a least-squares subroutine is called for each spatial reconstruction. Although the procedure based on reusing the pseudo-inverse matrix requires extra storage, the additional memory requirements are not that substantial and are generally readily available on most modern distributed memory architectures. Thus, the dimensions of the pseudo-inverse matrices encountered in this work are  $2 \times 8$  for linear ( $k=1$ ),  $5 \times 24$  for quadratic ( $k=2$ ),  $9 \times 24$  for cubic ( $k=3$ ) and  $14 \times 24$  for quartic ( $k=4$ ), which corresponds to storing 16, 120, 216 and 336 floating point numbers, respectively. Moreover, these memory requirements do not depend on the number of solution variables since the same pseudo-inverse matrix is employed for the computation of each solution unknown. This procedure therefore represents a practical approach for reducing the cost of the proposed high-order reconstruction and is used here.

Finally, to provide concrete examples concerning the condition number,  $\kappa(\mathbf{A})$ , of the coefficient matrix,  $\mathbf{A}$ , for the least-squares problem corresponding to different reconstruction orders,  $k$ , several representative quadrilateral meshes have been analyzed herein. Note that the condition number has been estimated here based on  $L_\infty$  norm of the matrix, such that  $\kappa_\infty(\mathbf{A}) = \|\mathbf{A}\|_\infty \|\mathbf{A}^{-1}\|_\infty$ , where the pseudo-inverse matrix,  $\mathbf{A}^{-1}$ , has been determined as described above. Thus, Table 3.1 depicts the maximum values of  $\kappa_\infty(\mathbf{A})$  associated with the linear ( $k=1$ ), quadratic ( $k=2$ ), cubic ( $k=3$ ) and quartic ( $k=4$ ) solution reconstructions on Cartesian grids with aspect ratio,  $a_r$ , of 1, 240 and 14,012, respectively, and on the regular smooth and irregular grids illustrated in Fig. 3.1(a) and (b), respectively. In this study, the effect of different geometric weighting formulations,  $w_J$ , on the condition number for the least-squares problem has also been considered, and the results are also shown in the table.

An analysis of the data in Table 3.1 reveals that the condition numbers of the coefficient matrix increase with the increase in the reconstruction order. This trend is somewhat expected as the elements of the coefficient matrix,  $\mathbf{A}$ , corresponding to larger reconstruction orders are determined based on a larger range of powers of the distances between points in the stencil. Additionally, the data shows that larger and larger maximum values occur as the mesh deviates more and more from the Cartesian square grid, the largest values encountered in this study corresponding to the Cartesian grid with highest aspect ratio. However, the data also reveals that significant improvements in the condition number can be obtained, even up to two orders of magnitude, by using some sort of geometric weighting. For the current scheme and the meshes studied herein it seems that the inverse-distance geometric weighting (i.e.,  $w_J = |\Delta \vec{X}_{IJ}|^{-1}$ ) provides the lowest condition number values for all reconstruction orders analyzed. Moreover, although the condition numbers encountered on Cartesian meshes with a high aspect ratio of  $a_r = 14,012$  are very large, it is argued that a high-order scheme may not require such high aspect ratios to accurately resolve solution features of thin boundary and/or shear layers in practice.

Therefore, it seems that for the reconstruction orders and the meshes considered in this work, the corresponding condition numbers can be dealt with, but caution should be exercised when considering both higher than quartic reconstruction orders or very high grid aspect ratios. For the latter cases, it may be more appropriate to consider polynomial expansions which can provide lower condition numbers, such as the expansion in barycentric coordinates described by Abgrall [20] for triangle elements, or application of regularization

techniques for solving ill-conditioned systems [141].

Table 3.1: *The condition numbers of the coefficient matrix,  $\mathbf{A}$ , obtained based on  $L_\infty$  norm of the matrix for different geometric weighting formulations and for several representative meshes, which are as follows: the Cartesian grids with aspect ratio,  $a_r$ , of 1, 240 and 14,012; the regular smooth grid shown in Fig. 3.1(a); and the irregular grid shown in Fig. 3.1(b).*

Mesh	Geometric Weighting $w_J =  \Delta \vec{X}_{IJ} ^{-\theta}$	$\kappa_\infty(\mathbf{A})$			
		$k=1$	$k=2$	$k=3$	$k=4$
Cartesian Grid $a_r = 1$	$\theta=0$	2.0	25.4	452.2	6,748.4
	$\theta=1$	1.7	22.0	273.2	3,223.8
	$\theta=2$	1.3	32.6	335.3	4,007.2
Cartesian Grid $a_r = 240$	$\theta=0$	227.6	292,335.1	$5.71 \times 10^8$	$9.69 \times 10^{11}$
	$\theta=1$	1.8	1,116.6	$1.72 \times 10^6$	$2.01 \times 10^9$
	$\theta=2$	239.0	245,311.9	$3.15 \times 10^8$	$4.03 \times 10^{11}$
Cartesian Grid $a_r = 14,012$	$\theta=0$	12,780.4	$8.40 \times 10^8$	$8.78 \times 10^{13}$	$1.10 \times 10^{14}$
	$\theta=1$	1.8	60,118.4	$4.66 \times 10^9$	$9.17 \times 10^{13}$
	$\theta=2$	15,047.1	$8.32 \times 10^8$	$5.40 \times 10^{13}$	$2.78 \times 10^{14}$
Regular Smooth Grid	$\theta=0$	4.1	172.9	9,232.4	524,247.3
	$\theta=1$	2.2	77.5	3,815.2	171,418.2
	$\theta=2$	3.2	141.9	4,928.2	192,885.6
Irregular Grid	$\theta=0$	5.2	191.7	10,552.5	699,826.2
	$\theta=1$	2.4	84.1	4,661.8	199,873.8
	$\theta=2$	4.6	203.2	6,688.5	240,616.7

### 3.6.2 Monotonicity Enforcement

*“A problem well stated is a problem half solved.”*

by Charles F. Kettering (1876-1958), American engineer & inventor

In his 1959 paper [130], Godunov introduced the important concept of monotonicity and proved in a theorem that currently bears his name that among schemes of second-order accuracy for the linear advection equation with constant coefficients there are none which transform any monotonic function into other monotonic one between two successive time steps. The theorem implies that any higher-order *linear* schemes (i.e.,  $k > 0$ ), although more accurate for smooth solutions, tend to introduce spurious oscillations (wiggles) where discontinuities or shocks arise. To overcome these drawbacks, various high-resolution, *non-linear* techniques have been proposed in the literature.

Using the concept of *total variation* first introduced by Harten [136], these methods can be formally classified into *total variation diminishing (TVD)*, *total variation bounding (TVB)* and *essentially non-oscillatory (ENO)* schemes. It should be pointed out that all these monotonicity concepts introduce gradually **less restrictive conditions** for satisfying monotonicity which allow them to increase the spatial solution accuracy while still to be sufficient to guarantee the convergence of the numerical solution to weak solutions of the solved PDE. Technically, a numerical scheme is said to be TVD if the following condition holds true at any time step  $n$

$$\text{TV}(u^{n+1}) \leq \text{TV}(u^n), \quad (3.25)$$

where the total variation, TV, of a one-dimensional numerical solution,  $u(x)$ , is defined as

$$\text{TV}(u) \equiv \sum_i |u_{i+1} - u_i|. \quad (3.26)$$

Harten [136] showed that all TVD schemes are *monotonicity preserving*, which means the following properties are maintained as function of time :

- i) No new local extrema in  $x$  (i.e., spatial coordinate) may be created
- ii) The value of a local minimum is non-decreasing, the value of a local maximum is non-increasing.

A general framework for generating second-order, upwind, TVD discretizations was formulated by Van Leer [142, 143] with the introduction of Monotone Upstream-centred Schemes for Conservation Laws (MUSCL) scheme. In the MUSCL approach, the TVD conditions are ensured by restricting the amplitude of the gradients appearing in the piecewise linear solution reconstructions via *non-linear slope limiting*. Similar results can also be obtained by applying flux limiting [144]. Since their introduction in the early 1970s, various limiter functions have been defined in the literature. A detailed description of several limiters along with their properties is given in [15, 144, 145]. Note that TVD schemes are globally second-order, but reduce to first-order accuracy at local extrema. The introduction of the limiters within second-order upwind schemes has the effect of removing solution oscillations near discontinuous and/or almost discontinuous solution content, producing monotone profiles. However, it also has the effect of reducing locally the accuracy of the solution around smooth extrema, a phenomenon known as “peak clipping”. This phenomenon occurs due to the fact that limiters, in general, cannot readily distinguish between sharp discontinuities and local smooth extrema and results in a degradation of the scheme spatial accuracy in smooth regions. Even for smooth solutions, the clipping of extrema effectively reduces the order of accuracy of the numerical scheme to  $\mathcal{O}(\Delta x)$  in the  $L_\infty$ -norm, potentially rendering too dissipative a TVD procedure for simulation of complex flows characterized by a significant presence of solution minima and maxima. Nevertheless, several limiting procedures for preserving accuracy at smooth extrema, at least for specific schemes, have also been proposed. Included in this category of extremum preserving limiters are the recently proposed limiters by Colella and Sekora [36] and by McCorquodale and Colella [35], and the parameter-free generalized moment limiter of Yang and Wang [52, 53] based on a hierarchical approach.

The class of TVB schemes attempt to avoid to some extent the problem of clipping the extrema at the expense of compromising the solution monotonicity near discontinuities/large gradients. These schemes are defined by the following property [146]:

$$\text{TV}(u^n) \leq B, \tag{3.27}$$

where the constant  $B$  is fixed, greater than zero and depends only on the total variation of the initial solution,  $\text{TV}(u^0)$ . Consequently, TVB schemes tolerate large spurious oscillations provided only that the oscillations remain bounded.

However, formulation of truly *uniform* second- and even higher-order schemes that do

not exhibit  $\mathcal{O}(1)$  spurious oscillations near discontinuities was possible by enlarging the TVD concept to a broader class of schemes, designated as ENO. This new concept introduced by Harten *et al.* in their landmark paper [18], allowed *spurious oscillations* to exist in smooth regions on the level of the scheme truncation error, thereby allowing the numerical discretization to provide high-order accuracy even around smooth extrema, while avoiding a Gibbs-like phenomenon to occur near discontinuities, that is no solution oscillations proportional to the size of the solution jump exist. Moreover, the spurious oscillations would diminish as the mesh is refined at a rate imposed by the order of the scheme. Following Harten *et al.* [18], the ENO property in terms of total variation is defined as

$$\mathrm{TV}(u^{n+1}) = \mathrm{TV}(E_h \bullet u^n) \leq \mathrm{TV}(u^n) + \mathcal{O}(\Delta x^{k+1}) , \quad (3.28)$$

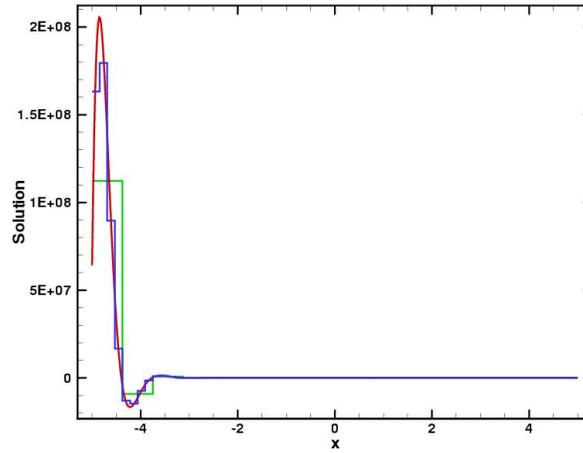
where  $E_h$  is the discrete operator representing the effect of all stages of the numerical scheme in the process of advancing the solution from time level  $n$  to level  $n + 1$ . In case of Godunov-type finite-volume schemes, the ENO condition is translated directly to ENO criteria for the reconstruction procedure. A reconstruction algorithm that provides the ENO property was also presented in the original work of Harten *et al.* [18] and was based on selecting the “smoothest” reconstruction stencil among several candidates. The quality of being the *smoothest stencil* (i.e., leading to the solution reconstruction with the least variation in some norm) was determined based on the value of the largest divided difference, which, for one-dimensional problems, gives a clear indication of how oscillatory a reconstruction may be. Note that direct extensions of this selection criterion to multi-dimensions are not easily possible and other methods to characterize the reconstruction smoothness are often necessary. However, despite its inherent difficulties for extension to multi-dimensions with more irregular and unstructured meshes, the ENO algorithm proposed by Harten *et al.* is still possibly the most cost-effective accurate approach for high-order finite-volume schemes, as discussed in Sect. 1.3.1. Note also that the ENO property as defined by Harten *et al.* does not require the existence of a stencil selection algorithm, the latter representing merely one of the means in which the former can be provided to a numerical scheme.

In the current work, an alternative approach is considered for providing a numerical scheme with the ENO property. The proposed CENO scheme described herein combines the unlimited high-order  $k$ -exact least-squares reconstruction described in Sect. 3.6.1 with a monotonicity preserving limited piecewise linear least-squares reconstruction algorithm.

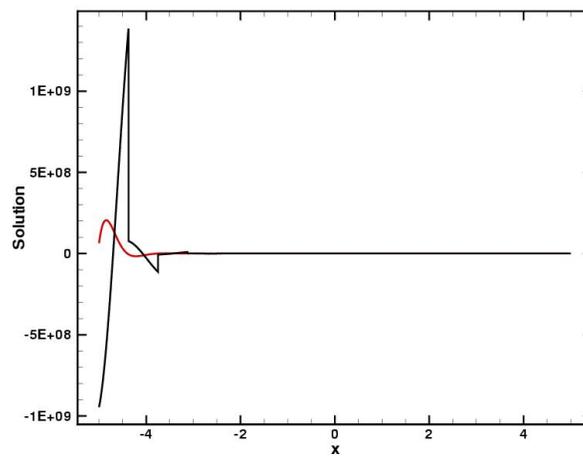
The limited reconstruction procedure is applied to computational cells with under-resolved solution content and the unlimited  $k$ -exact reconstruction scheme is used for cells in which the solution is fully resolved. Switching in the hybrid procedure is determined by a solution smoothness indicator that indicates whether or not the solution is resolved on the computational mesh. Note that by design the piecewise  $k$ -exact polynomials would introduce spurious numerical oscillations on the level of truncation error in smooth regions as shown in Fig. 1.3. Unfortunately, these otherwise small oscillations become large in under-resolved regions generating a Gibbs-like phenomenon. Even smooth analytic functions can generate this phenomenon if the mesh is under-resolved, a behaviour illustrated in Fig. 3.3 with the cubic reconstruction of  $f(x) = e^{-4x} \sin(5x)$  represented on two different mesh resolutions. As can be easily observed in Fig. 3.3(b), the reconstruction performed on the under-resolved mesh exhibits the same behaviour as that for a discontinuous data, Fig. 1.4, which demonstrates that the smoothness property of discrete data is in general very dependent on the particular mesh resolution. Consequently, a reliable smoothness indicator must be able to identify correctly as *smooth* those reconstructions that oscillate on the level of truncation error and as *non-smooth* those polynomials that exhibit large oscillations relative to the solution content within the neighbourhood. As the reader can spot, defining what is an *acceptable and unacceptable* degree of oscillation for a large range of solution length scales represented on a given mesh resolution can be problematic, especially for reconstructions in multiple dimensions, and the task at hand is by far not an easy one at all. The next section will describe the particular form of the smoothness indicator considered in this work and how it manages to overcome the resolution problem in a relatively simple way.

### Smoothness Indicator

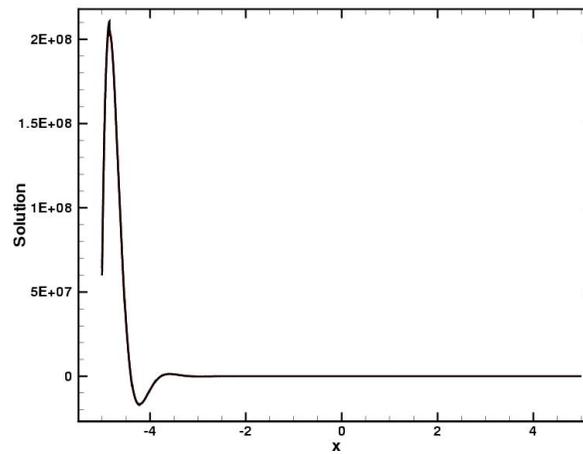
The proposed CENO scheme preserves solution monotonicity in regions of large gradients or discontinuities by reverting the high-order  $k$ -exact reconstruction to a limited piecewise linear ( $k=1$ ) reconstruction. This approach of preserving solution monotonicity by “dropping” the reconstruction order leads to a non-uniformly accurate reconstruction (i.e., reconstructions of lower and higher order coexist in the computational domain) for under-resolved and/or non-smooth solution content. However, the reconstruction procedure remains uniformly accurate, even in the presence of smooth extrema, as long as the solution is smooth everywhere. In the current work, the slope limiters of Barth-Jespersen [16] and Venkatakrishnan [147] are used



(a) Exact solution (red) and piecewise constant data for low- and high-resolution shown in green and blue



(b) Reconstruction on low-resolution mesh



(c) Reconstruction on high-resolution mesh

Figure 3.3: Cubic ( $k=3$ ) reconstruction of analytical function  $f(x) = e^{-4x} \sin(5x)$  on two mesh resolutions. Note the large oscillations for the under-resolved mesh shown in (b).

in the limited reconstruction, but other limiters would do almost equally as well. In order to detect regions where the order of the reconstruction should be reduced and the limiters applied, a smoothness indicator is computed for every variable individually within each cell as part of a post-analysis step after the unlimited  $k$ -exact reconstruction has been performed. The smoothness indicator is then used in the manner described below to ensure that the limited linear reconstruction procedure is applied to computational cells with under-resolved and/or non-smooth solution content and the unlimited  $k$ -exact reconstruction scheme is where the solution is fully resolved.

The form of the smoothness indicator used here was inspired by the definition of multiple-correlation coefficients that are often used in evaluating the accuracy of curve fits [139]. The basic idea is to assess how accurately the truncated polynomial expansion represents the solution data within the reconstruction stencil. This is achieved by comparing the reconstructed solution to those in neighbouring cells. In the case of non-smooth solution data, the intrinsic assumption of solution smoothness made in determining the accuracy of  $k$ -exact reconstruction, given by Eq. (3.15), is definitely incorrect and the post-analysis of the obtained reconstructions should illustrate this fact. It should also be clear that for smooth solution content, the predictions of the reconstructions within neighbouring cells should become increasingly more similar as the mesh is refined, in a manner similar to the way in which the difference between the predicted solution states at inter-cell boundaries diminishes. Therefore, this approach naturally accounts for transitions from under-resolved to resolved solution content with increasing mesh resolution.

The smoothness indicator,  $\mathcal{S}$ , is calculated in terms of a smoothness parameter,  $\alpha$ , as well as information about the number of unknowns (degrees of freedom),  $DOF$ , and size of the stencil,  $SOS$ , used in the reconstruction. It is taken to have the form

$$\mathcal{S} = \frac{\alpha}{\max((1 - \alpha), \epsilon)} \frac{(SOS - DOF)}{(DOF - 1)}, \quad (3.29)$$

where  $\alpha$  is determined as follows

$$\alpha = 1 - \frac{\sum_{\gamma} \sum_{\delta} (u_{\gamma,\delta}^k(\vec{r}_{\gamma,\delta}) - u_{i,j}^k(\vec{r}_{\gamma,\delta}))^2}{\sum_{\gamma} \sum_{\delta} (u_{\gamma,\delta}^k(\vec{r}_{\gamma,\delta}) - \bar{u}_{i,j})^2}, \quad (3.30)$$

and where the ranges of the indices,  $\gamma$  and  $\delta$ , are taken to include all control volumes in the reconstruction stencil for cell  $(i, j)$ ,  $\vec{r}_{\gamma,\delta}$  is the centroid of the cell  $(\gamma, \delta)$ , and the tolerance,  $\epsilon$ ,

has been introduced in order to avoid division by zero. A suitable value for  $\epsilon$  has been found to be  $10^{-8}$ . It should be evident that the parameter,  $\alpha$ , compares the values of the reconstructed solution at the centroids of neighbouring cells used in the solution reconstruction for cell  $(i, j)$ . Note also that the average value in the reconstructed cell is used to normalize the relative predictions of the reconstructions. The range for  $\alpha$  is  $-\infty < \alpha \leq 1$  and it will approach unity as the solution becomes smooth and near perfect matching of the reconstructions in adjacent cells is achieved. The quotient in the expression for  $\alpha$  can become a ratio of two very small numbers or  $\frac{0}{0}$  in situations with little or no solution variation is encountered. This scenario would translate into the problem that reconstructions of free-stream conditions are not deemed as smooth and a second reconstruction would be required in these regions. Obviously, such situations can be easily avoided by computing  $\alpha$  only in those regions of the domain where a minimum level of solution variation relative to a reference solution value is encountered. The variation of  $\frac{\alpha}{(1-\alpha)}$  is depicted in Fig. 3.4 and the figure shows that  $\mathcal{S}$  rapidly becomes large as  $\alpha$  approaches unity.

Once evaluated, the smoothness indicator,  $\mathcal{S}$ , is then compared to a pass/no-pass cutoff value,  $\mathcal{S}_c$ . As appropriate value for the cutoff was determined from a wide range of numerical experiments and values for  $\mathcal{S}_c$  in the range 1,000-5,000 seem to work well. It should be pointed

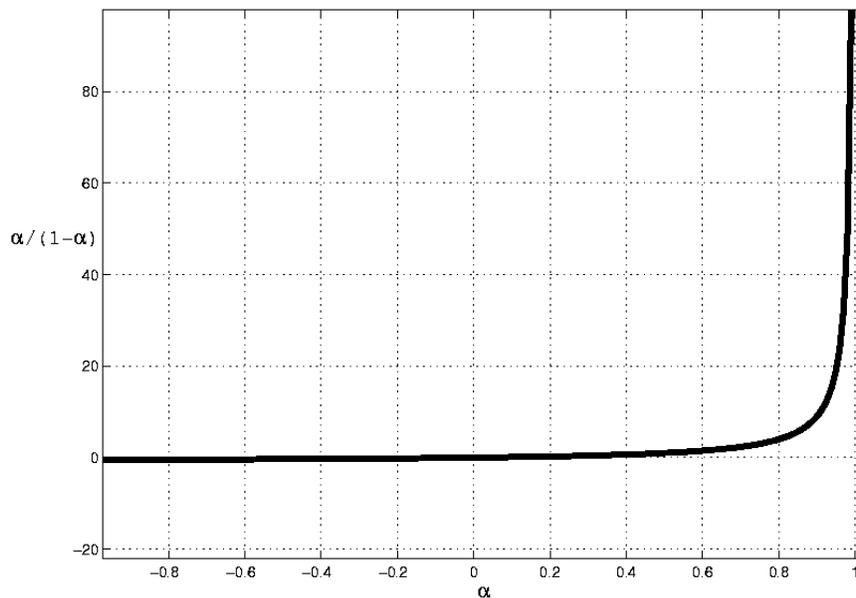


Figure 3.4: The graph of  $f(\alpha) = \frac{\alpha}{(1-\alpha)}$ .

out that smoothness indicators for smooth solutions are typically orders of magnitude larger than the range of cutoff values (e.g., 100,000). Moreover, typical values for cells located in the “middle” of discontinuities are close to 10. Thus for  $\mathcal{S} < \mathcal{S}_c$ , the solution is deemed to be under-resolved and/or non-smooth and the high-order  $k$ -exact reconstruction is replaced by limited linear reconstruction in that cell. For  $\mathcal{S} > \mathcal{S}_c$ , the unlimited high-order reconstruction is deemed to be acceptable and retained. Finally, it should be pointed out that the effect of a reconstruction which contains discontinuous data is reflected in the smoothness indicator calculation of few neighbours and consequently, any solution discontinuity is typically confined to within at most 7 to 10 cells.

Numerical experiments with the smoothness indicator calculated using only information from a reduced set of neighbour reconstructions have recently shown that a more local approach can be as successful as using the whole supporting stencil in identifying smooth and non-smooth reconstructions correctly, while significantly reducing the computational cost and improving the locality of this operation. Thus, instead of using the reconstructions of all cells that are part of the supporting stencil, two algorithms have been devised using only the first-degree (i.e., the nearest) neighbours. The first algorithm used all first-degree neighbours, including the corners, and the reconstructions have been compared at the cell centroids. The second algorithm used only the neighbours that shared a face with the reconstructed cell and the reconstructions have been compared at the Gauss quadrature points of each shared face. Obviously, these two approaches require fewer evaluations for each smoothness check and can potentially confine the non-smooth data to a more local region. However, more extensive testing of these ideas is required prior to making conclusive statements on alternative evaluation procedures for the smoothness indicator.

### **Monotonicity enforcement for systems of conservation laws**

For systems of conservation laws, the reconstruction procedure can be applied to different sets of interrelated solution variables, such as conserved, primitive or characteristic variables, and it is important to realize that enforcing monotonicity on the variables that are used for reconstruction does not necessarily ensure the solution positivity when the conditions for positivity are expressed in terms of a different set of equivalent solution variables. For such situations, the recommended approach is to detect non-smooth solution regions based on the reconstructed variables and subsequently to perform a limited piecewise linear reconstruction

with those variables on which positivity conditions are formulated. As an example, the solution procedure of Euler and Navier-Stokes equations that is based on the high-order reconstruction of the conserved variables should enforce positivity of density and pressure by performing a limited reconstruction of the primitive variables in those regions detected as non-smooth based on the conserved ones.

### 3.6.3 Reconstruction at Boundaries and Implementation of High-Order Boundary Conditions

Correct high-order treatment of boundary conditions is a crucial element for developing accurate numerical schemes. It is especially important for high-order methods, where errors due to geometrical approximation may dominate the discretization error, mitigating the full capabilities of a high-order scheme. One approach to imposing high-order boundary conditions is to make use of extra rows of ghost cells which are added beyond the geometric boundary of the computational domain. Solution states are then imposed in the ghost cells in such a way that the reconstructed solution and/or flux at the boundary flux interior cell approximates those associated with the particular boundary condition. An alternative approach is to enforce the boundary conditions by constraining the least-squares reconstruction in control volumes adjacent to the boundary as described by Olivier-Gooch and Van Altena [72]. In the current work, both procedures (i.e., ghost cells and constrained reconstruction) for boundary-condition prescription have been implemented and are described herein. Note that in this work it has not been considered the possibility to reduce by one the order of the scheme at boundaries, a procedure shown by Gustafsson [148] to produce the same convergence rate as that of the interior scheme, due to additional algorithm complexities.

In order to obtain high-order accuracy at boundaries, the geometric data (i.e., cell area, centroid, geometric moments, normals, edge lengths, locations of the Gauss quadrature integration points) are computed to the same order of accuracy as that of the interior scheme. A typical configuration encountered near curved geometric boundaries is depicted in Fig. 3.5, in which curved-edge quadrilaterals are used to represent the geometric boundary accurately. The reader is reminded that computation of the required geometric properties of quadrilateral cells is described in detail in Appendix A.

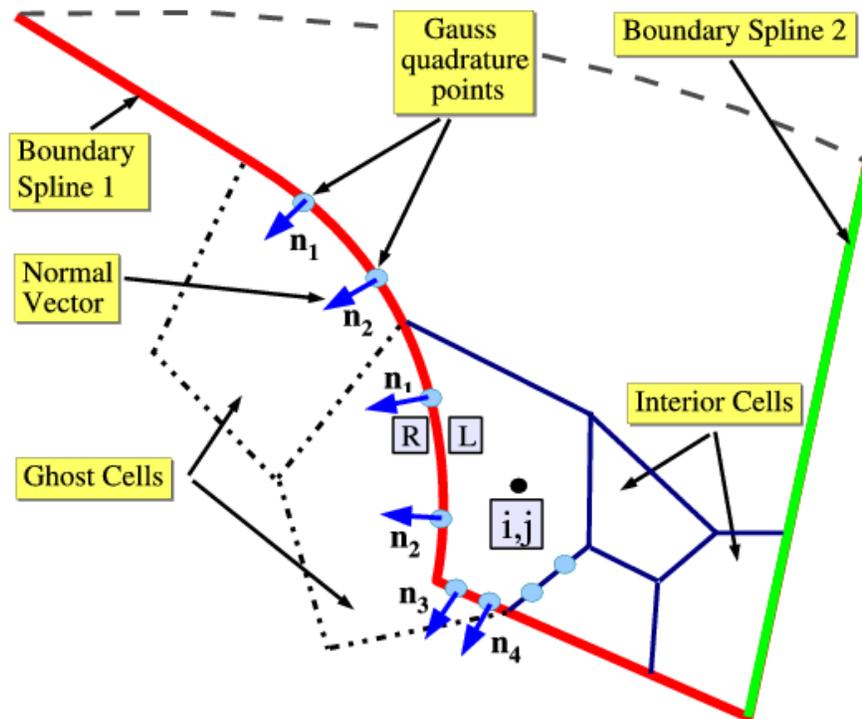


Figure 3.5: *Schematic representation of geometric elements at the boundaries of a quadrilateral mesh. The blue vectors represent the local normals at the Gauss quadrature points.*

### High-Order BCs via Ghost Cells

As outlined above, there are two approaches considered for implementing boundary conditions which have direct implications on the reconstruction algorithm used near boundaries. If additional layers of ghost cells are used for specifying boundary conditions, the reconstruction algorithm for the cells near boundaries is the same as that outlined for an interior cell. Note that in our implementation of BCs with additional ghost cells, the reconstruction stencil remains unmodified near boundaries and the order of the polynomial interpolation is unchanged. While different approaches may be used in conjunction with the CENO algorithm described herein and potentially offer some advantages, they have not been considered in the current work. Thus, the only requirements for implementing high-order BCs with ghost cells are to ensure that sufficient layers of ghost cells are used in order to carry out the reconstruction and the computation of the smoothness indicator in all necessary cells and to specify accurate solution data in each ghost cell. Before discussing the first requirement, it should be pointed out that the latter condition cannot be always easily satisfied, especially

when curved boundaries are present. A more appropriate approach to dealing with these situations is the use of constrained reconstruction which is described later in this section. Turning our attention to the first condition, we note that the minimum number of ghost cells is defined by the size of the reconstruction stencil and the number of neighbours used to assess the smoothness indicator. For CENO reconstructions of orders up to four the minimum number of ghost cells varies between three and five. Note that for a standard second-order scheme with a least-squares reconstruction a minimum of two ghost cells is required, the first one to be used for computing the flux through the domain boundary and the second one to be included in the reconstruction of the former. To understand why more than two cells are required in the case of CENO, it is instructive to look at the requirements of a quartic reconstruction. First, the reconstruction of the ghost cell used in the boundary flux calculation will require *two* additional ghost cells for obtaining its symmetric central reconstruction stencil with 24 neighbours, which brings to *three* the minimum number of required ghost cell layers. This number would be sufficient if the smoothness indicator were not calculated for the reconstruction of the first ghost cell near boundary. However, depending on how many neighbours one wants to use for assessing the smoothness indicator of that cell, more additional layers are required for performing the reconstructions of the neighbours. Thus, one needs a total of five layers of ghost cells when all cells in the reconstruction stencil are used for checking the smoothness and four layers instead when the smoothness indicator is calculated only with the first-degree neighbours.

### High-Order BCs via Constrained Reconstruction

Implementing BCs with additional layers of ghost cells can work well for relatively straight boundaries, but may give rise to large errors when the geometry is more highly curved. The difficulty consists in specifying accurate solution information in the ghost cells which then translates to accurate high-order reconstructions that correctly resemble the solution variation required for the particular boundary condition. A concrete illustration of the difficulties encountered in such a process is provided by the implementation of inviscid solid wall boundary condition,  $\vec{V} \cdot \vec{n} = 0$ . This boundary condition can be implemented in the weak sense by mirroring the velocity vector in the interior cells adjacent to the boundary into the correspondent ghost cells [149]. For straight geometric boundaries this approach works well, but for curved boundaries one is immediately confronted with the problem of

providing a suitable normal for the mirroring process which results in accurate high-order reconstructions for the velocity components. The problem is further exacerbated by the fact that the aforementioned condition  $\vec{V} \cdot \vec{n} = 0$  should be ideally satisfied everywhere along the solid boundary and must be definitely fulfilled at the flux calculation points, which can be more than one. Obviously, these requirements on the velocity vector, which translate into conditions on the least-squares reconstructions of the two velocity components, are difficult, if not impossible, to be satisfied with the information provided by the ghost cells.

Therefore, a better alternative is to enforce the boundary conditions using only information from the interior domain and from the required boundary values at the Gauss integration points along the boundary. Thus, additional constraints are added to the reconstruction procedure to ensure that the values taken by the polynomial approximations of the interior cells next to the boundary at the Gauss integration points are exactly those required by the particular boundary condition. By constraining the least-squares reconstruction in control volumes adjacent to the boundary, complex boundary conditions can be enforced. Nevertheless, for certain boundary conditions this approach might be too difficult, and therefore, in the current implementation both procedures (i.e., ghost cells and constrained reconstruction) are allowed. In general, boundary conditions which include linear relations between the reconstructed variables are relatively easy to be implemented with the constrained reconstruction approach whereas non-linear relations can be difficult to specify.

The general boundary-condition framework developed as part of this thesis research allows for two basic types of constraints to be specified. They are: 1) Cauchy (i.e., combination of Dirichlet and Neumann) conditions which can be applied individually to any reconstructed variable as the need may be; and 2) linear relations among variables which are applied as coupling constraints to a set of reconstructed variables. Thus, any boundary condition that can be expressed by these two basic types can be accommodated by the framework. Each of these boundary types is discussed in detail below. In order to solve the enlarged system of equations arising from the constrained least squares reconstruction (boundary constraints plus reconstruction conditions), Gauss elimination with pivoting is used to eliminate the rows associated with the boundary-condition constraints, and the remaining least-squares problem is again solved using Householder QR factorization. Note that, in general, storage of a pseudo-inverse matrix  $\mathbf{A}^{-1}$  for re-use is not possible in this case, as entries in the matrix  $\mathbf{A}$  can change even for a fixed mesh. Even though some performance penalties are incurred

in the proposed approach for constrained BCs they will be restricted only to those cells adjacent to boundary.

Similarly to the derivation outlined in [72], the equation used to describe the Cauchy boundary condition at each flux integration point is derived for a reconstructed variable,  $u$ , as follows. The function  $f(\vec{X}, a, b)$  which provides the value of the Cauchy boundary condition at each location  $\vec{X}$  of interest can be expressed as

$$f(\vec{X}, a, b) = a(\vec{X}) f_D(\vec{X}) + b(\vec{X}) f_N(\vec{X}) \quad (3.31)$$

where the coefficients  $a(\vec{X})$  and  $b(\vec{X})$  define the contribution of the Dirichlet,  $f_D = u(\vec{X})$ , and Neumann,  $f_N = \frac{\partial u(\vec{X})}{\partial n}$ , components respectively. All elements in the linear combination are considered to be known at each Gauss integration point. Thus, for a given integration point,  $\vec{X}_g$ , the Dirichlet condition can be expressed in terms of the cell reconstruction,  $u^k(\vec{X})$ , as

$$f_D(\vec{X}_g) = u^k(\vec{X}_g) = \sum_{\substack{p_1=0 \\ (p_1+p_2 \leq k)}}^k \sum_{\substack{p_2=0 \\ (p_1+p_2 \leq k)}}^k (x_g - \bar{x}_{i,j})^{p_1} (y_g - \bar{y}_{i,j})^{p_2} D_{p_1 p_2}^k. \quad (3.32)$$

Similarly, the Neumann condition becomes

$$\begin{aligned} f_N(\vec{X}_g) &= \frac{\partial u^k(\vec{X}_g)}{\partial n} = \nabla u^k(\vec{X}_g) \cdot \vec{n}_g \\ &= n_x^g \frac{\partial u^k}{\partial x} \Big|_{\vec{X}_g} + n_y^g \frac{\partial u^k}{\partial y} \Big|_{\vec{X}_g} \\ &= \sum_{\substack{p_1=0 \\ (p_1+p_2 \leq k)}}^k \sum_{\substack{p_2=0 \\ (p_1+p_2 \leq k)}}^k (x_g - \bar{x}_{i,j})^{(p_1-1)} (y_g - \bar{y}_{i,j})^{(p_2-1)} \left[ p_1 (y_g - \bar{y}_{i,j}) n_x^g + p_2 (x_g - \bar{x}_{i,j}) n_y^g \right] D_{p_1 p_2}^k. \end{aligned} \quad (3.33)$$

Replacing the equations for  $f_D$ , Eq. (3.32), and  $f_N$ , Eq. (3.33), in Eq. (3.31) the general constraint equation which must be satisfied at a flux calculation point  $\vec{X}_g$  becomes

$$\begin{aligned} f(\vec{X}_g) &= f(a, b, f_D, f_N) \Big|_{\vec{X}_g} \\ &= \sum_{\substack{p_1=0 \\ (p_1+p_2 \leq k)}}^k \sum_{\substack{p_2=0 \\ (p_1+p_2 \leq k)}}^k \left\{ \Delta X_g^{(p_1-1)} \Delta Y_g^{(p_2-1)} \left[ a \Delta X_g \Delta Y_g + b p_1 \Delta Y_g n_x^g + b p_2 \Delta X_g n_y^g \right] \right\} D_{p_1 p_2}^k \end{aligned} \quad (3.34)$$

where  $\Delta X_g = x_g - \bar{x}_{i,j}$ ,  $\Delta Y_g = y_g - \bar{y}_{i,j}$ , and  $a$  and  $b$  linear coefficients are specific to the location of interest. This equation is valid for any reconstruction order  $k$  and the general term

between braces can be used directly to generate the entries corresponding to any combination of  $p_1$  and  $p_2$  in the LHS matrix of the linear system for a particular integration point.

An example of coupled constraints is the implementation of inviscid solid wall condition,  $\vec{V} \cdot \vec{n} = 0$ , which is imposed by solving a coupled constrained reconstruction with the velocity components and requiring that the velocity vector is tangent to the geometry at every Gauss integration point. Thus, the equation that needs to be satisfied by the reconstruction coefficients of the  $x$ - and the  $y$ -velocity components,  $(D_{p_1 p_2}^k)_u$  and  $(D_{p_1 p_2}^k)_v$  respectively, is

$$\sum_{\substack{p_1=0 \\ (p_1+p_2 \leq k)}}^k \sum_{\substack{p_2=0 \\ (p_1+p_2 \leq k)}}^k \Delta X_g^{p_1} \Delta Y_g^{p_2} n_x^g (D_{p_1 p_2}^k)_u + \sum_{\substack{p_1=0 \\ (p_1+p_2 \leq k)}}^k \sum_{\substack{p_2=0 \\ (p_1+p_2 \leq k)}}^k \Delta X_g^{p_1} \Delta Y_g^{p_2} n_y^g (D_{p_1 p_2}^k)_v = 0 . \quad (3.35)$$

The matrix  $\mathbf{A}$  for a constrained least-squares reconstruction in which two variables are coupled contains entries for the relational constraints, for the individual constraints of each variable and for the exactly and approximately satisfied mean conservation equations provided by the reconstructed cell and the neighbours that are part of the stencil, respectively. Thus, the matrix  $\mathbf{A}$  in one cell has the general form shown in Table 3.2, where all equations above the horizontal line are solved exactly and those below this line have the error minimized in the least-squares sense. Note the importance of pivoting in the Gauss elimination algorithm applied to the constrained equations in order to avoid division by zero.

In the boundary-condition framework, the practical implementation of high-order boundary conditions starts with the requirement that each of the four block boundaries defined by boundary splines must have one of the two approaches for boundary-condition implementation specified: that is either boundary-condition specification based on ghost cells or constrained reconstruction. Note that there are no restrictions regarding the choice of boundary-condition implementation for each geometric boundary, so any combination is acceptable. The treatment of the interior boundaries between active solution blocks, in the case of multi-block meshes, is *always* done using reconstruction with ghost cells. However, this level of flexibility for implementing boundary conditions requires exercising care in the way the reconstruction stencils are defined such that one always obtains a *valid yet compact* stencil. The validity of a stencil is assessed based on the condition that there are sufficient equations to compute the reconstruction coefficients. For multi-block meshes, as discussed in Chapt. 4, it is also required that the stencils of overlapping ghost and interior cells in adjacent blocks are *identical* for the sake of consistency.

Table 3.2: *General form of the left-hand-side matrix, A, corresponding to a constrained least-squares reconstruction in which two variables are coupled.*

$$\mathbf{A} = \left[ \begin{array}{c} \dots \\ \text{Relational Constraints} \\ \text{Between Variable 1 and 2} \\ \dots \\ \dots \\ \text{Individual Constraints} \\ \text{for Variable 1} \\ \dots \\ \dots \\ 0 \\ \dots \\ \dots \\ \text{Mean Conservation for Variable 1} \\ \dots \ 0 \ \dots \\ \hline \dots \\ \text{Approximately Satisfied} \\ \text{Mean Conservation Equations} \\ \text{for Variable 1} \\ \dots \\ \dots \\ 0 \\ \dots \\ \dots \\ \text{Approximately Satisfied} \\ \text{Mean Conservation Equations} \\ \text{for Variable 2} \ \dots \end{array} \right]$$

To develop a systematic way of defining consistent and valid reconstruction stencils for any combination of BC implementations, two criteria are considered for determining the particular type of a specific block boundary. The first criterion categorizes a block boundary

based on whether the ghost cells behind it can be part or not of reconstruction stencils. Thus, a block boundary that allows its ghost cells to support solution reconstructions is termed *transparent*. In contrast, a boundary that obstructs the use of ghost cells in reconstruction stencils is labelled *opaque*. The second criterion divides block boundaries into *constrained* and *unconstrained*, based on whether the BCs associated to a particular boundary are implemented or not via constrained reconstruction. The presence of a boundary which requires the BCs to be enforced via constrained reconstruction automatically excludes any usage of the ghost cells behind it and thus, the boundary becomes *opaque and constrained*. In general, *opaque* boundaries are associated with BCs enforced with constraints but for multi-block meshes, as discussed in the next chapter, situations arise in which a boundary is deemed to be *opaque*, although it is *unconstrained*, so as to maintain consistency of reconstruction stencils in adjacent blocks. This type of block boundary is denoted as *opaque and unconstrained*. Note that the effect of an *opaque* boundary on the central stencil of an interior cell is to significantly reduce the number of supporting neighbouring cells used in the reconstruction.

To illustrate the stencil construction issues in the presence of opaque boundaries, it is instructive to study the situation of a quartic reconstruction for a corner cell which has both boundary edges requiring constrained reconstruction. One can easily observe that the central stencil obeying the restrictions imposed by the opaque boundaries includes only 9 instead of 25 cells, which means one does not have enough equations to compute all 15 polynomial unknowns for the reconstruction. In case the reconstructed variable has physical boundary conditions that need to be satisfied at the Gauss integration points, which may or may not be the case for multivariate PDEs, the system of equations is completed with four additional constraints of the type of Eq. (3.34), so instead of 9 one would have 13 equations. Even in this situation, there are still *insufficient* equations to find the solution, so the stencil would be deemed as *invalid*. The approach used herein to avoid any occurrence of invalid stencils is to extend the supporting reconstruction stencil for any interior cell affected by the presence of opaque boundaries towards the interior domain. For the aforementioned example, this procedure would result in a stencil with 16 cells which provides a sufficiently large set of linear equations (i.e., one exactly and 15 approximately satisfied) to compute all 15 quartic polynomial unknowns.

To have well-posed problems for multivariate PDEs with boundary conditions based on constrained reconstruction, the number of variables for which constraints are imposed must

be in agreement with conditions imposed by the characteristics for the problem [124]. Note that in this approach no Riemann problem is solved at the boundary and therefore, the upwinding procedure must be directly embedded in the constrained reconstruction. An interesting situation arises for implementing the far-field boundary condition based on constraints. While the solution of a Riemann problem would ensure the correct propagation of the information for this BC, with constrained reconstruction one has to choose the number and the variables to which the constraints are applied based on the direction of the flow before the reconstructions are carried out. An estimation of the flow direction at the boundary has been considered herein based on the method suggested by Gottlieb and Groth [150] and adequate constraints for either inflow or outflow conditions are then imposed accordingly. Finally, it is worth mentioning that in our implementation, a constrained reconstruction identified as non-smooth is reverted to a limited linear least-squares reconstruction which satisfies the boundary conditions with ghost cells and the boundary flux is calculated as though no constraints would exist for that particular cell edge.

### 3.7 Inviscid (Hyperbolic) Flux Evaluation

An upwinding formulation is used herein to evaluate the hyperbolic flux at each quadrature point. Thus, given the left and right solution values,  $u_l$  and  $u_r$ , at the quadrature points, the hyperbolic flux in the advection-diffusion equation, Eq. (2.1), is given by

$$\vec{F}_H \cdot \vec{n} = \begin{cases} u_l (\vec{V} \cdot \vec{n}) & \text{if } \vec{V} \cdot \vec{n} \geq 0, \\ u_r (\vec{V} \cdot \vec{n}) & \text{if } \vec{V} \cdot \vec{n} < 0. \end{cases} \quad (3.36)$$

The left and right solution states,  $u_l$  and  $u_r$ , are determined by performing piecewise  $k$ -order CENO solution reconstruction within each computational cell, as outlined in Sect. 3.6.

For Euler and Navier-Stokes equations, the upwinding procedure is obtained by solving a Riemann problem at each Gauss quadrature point with the left and right solution vector states,  $\mathbf{U}_l$  and  $\mathbf{U}_r$ . In the present algorithm, both exact and approximate Riemann solvers can be used to solve the Riemann problem and evaluate the numerical flux. The Roe linearized Riemann solver [151], HLLE and modified HLLE flux function due to Linde [152, 153, 154], and the exact Riemann solver of Gottlieb and Groth [150] have all been implemented and may be used. In general, the approximate Riemann problem can be writ-

ten as a centrally weighting average of the left and right solution fluxes and some sort of numerical dissipation proportional to the solution jump at the interface, which results in

$$\mathcal{F}(\mathbf{U}_l, \mathbf{U}_r, \vec{n}) = \frac{1}{2} \left( \vec{\mathbf{F}}_H(\mathbf{U}_l, \vec{n}) + \vec{\mathbf{F}}_H(\mathbf{U}_r, \vec{n}) \right) - \frac{1}{2} |A(\mathbf{U}_l, \mathbf{U}_r, \vec{n})| (\mathbf{U}_l - \mathbf{U}_r), \quad (3.37)$$

where  $A(\mathbf{U}_l, \mathbf{U}_r, \vec{n})$  is a linearized dissipation matrix which depends on the local eigenstructure of the flux Jacobian matrix and is characteristic to each particular Riemann solver. For multi-dimensional problems, the normal flux is computed by applying a rotation of the left and right solution states along the local normal direction [115]. Note that the amount of numerical dissipation in Eq. (3.37) is very dependent on the difference between the left and right solution states. As this difference decreases in the resolved solution regions with the increase in the order of reconstruction, as explained in Sect. 3.6.1, one can easily understand the one of the basic mechanisms through which the high-order procedure improves the numerical solution at each flux calculation point. Not only is the numerical flux more accurate, due to better solution approximations, but the amount of numerical dissipation introduced by the upwind spatial discretization is also significantly reduced. The downside of the latter is that the numerical scheme can have a greater tendency to become unstable.

### 3.8 Viscous (Elliptic) Flux Evaluation

In a similar manner to hyperbolic fluxes, numerical elliptic fluxes in the advection-diffusion equation given by Eq. (2.1),  $\vec{F}_E \cdot \vec{n} = -\kappa \vec{\nabla} u \cdot \vec{n}$ , must be evaluated at each quadrature point of each control volume face. Having determined the left and right ( $k+1$ )-order accurate scalar solution reconstructions,  $u_l^k(\vec{X})$  and  $u_r^k(\vec{X})$ , a  $k$ -order accurate solution gradient at the inter-cellular face is obtained as the arithmetic mean of the left and right reconstruction gradients and thus, the elliptic flux at the calculation point,  $\vec{X}$ , is evaluated as

$$\vec{F}_E \cdot \vec{n} = -\kappa \left[ \frac{1}{2} \left( \vec{\nabla} u_l^k(\vec{X}) + \vec{\nabla} u_r^k(\vec{X}) \right) \right] \cdot \vec{n}. \quad (3.38)$$

Similarly, numerical diffusion fluxes,  $\vec{\mathbf{F}}_E \cdot \vec{n}$ , in the Navier-Stokes equations given by Eq. (2.14) through Eq. (2.16), are evaluated in the following manner

$$\vec{\mathbf{F}}_E \cdot \vec{n} = \vec{\mathbf{F}}_E \left( \mathbf{U}, \vec{\nabla} \mathbf{U} \right) \cdot \vec{n} = \vec{\mathbf{F}}_E \left( \frac{1}{2} \left( \mathbf{U}_l^k(\vec{X}) + \mathbf{U}_r^k(\vec{X}) \right), \frac{1}{2} \left( \vec{\nabla} \mathbf{U}_l^k(\vec{X}) + \vec{\nabla} \mathbf{U}_r^k(\vec{X}) \right) \right) \cdot \vec{n}, \quad (3.39)$$

where the two-dimensional reconstruction gradient,  $\vec{\nabla}U^k$ , at a given location  $\vec{X}$  is given by

$$\vec{\nabla}U^k(\vec{X}) = \left. \frac{\partial U^k}{\partial x} \right|_{\vec{X}} \hat{i} + \left. \frac{\partial U^k}{\partial y} \right|_{\vec{X}} \hat{j}. \quad (3.40)$$

In the above equation, Eq. (3.40),  $\hat{i}$  and  $\hat{j}$  are the Cartesian unit vectors and the derivative of the reconstructed polynomial in the  $x$ - and  $y$ -direction,  $\left. \frac{\partial U^k}{\partial x} \right|_{\vec{X}}$  and  $\left. \frac{\partial U^k}{\partial y} \right|_{\vec{X}}$ , respectively, are computed based on the polynomial coefficients as

$$\left. \frac{\partial U^k}{\partial x} \right|_{\vec{X}} = \sum_{\substack{p_1=0 \\ (p_1+p_2 \neq 0)}}^k \sum_{p_2=0}^k p_1 (x - \bar{x}_{i,j})^{p_1-1} (y - \bar{y}_{i,j})^{p_2} D_{p_1 p_2}^k, \quad (3.41a)$$

$$\left. \frac{\partial U^k}{\partial y} \right|_{\vec{X}} = \sum_{\substack{p_1=0 \\ (p_1+p_2 \neq 0)}}^k \sum_{p_2=0}^k p_2 (x - \bar{x}_{i,j})^{p_1} (y - \bar{y}_{i,j})^{p_2-1} D_{p_1 p_2}^k. \quad (3.41b)$$

To summarize, this chapter introduced the high-order CENO FV algorithm and its implementation on single-block structured meshes. Moreover, a detailed anatomy of CENO schemes has been performed and several relevant properties have been revealed. Numerical investigation of the proposed CENO discretization is performed in Chapt. 5. The next chapter will take the CENO scheme to another level of complexity by introducing its application in combination with a flexible, parallel, block-based, AMR procedure.

# Chapter 4

# HIGH-ORDER CENO WITH PARALLEL ADAPTIVE MESH REFINEMENT

The CENO finite-volume scheme described in Chapt. 3 is now considered for application on multi-block structured mesh subject to automatic solution-directed mesh adaptation. Strategies for how to provide high solution accuracy in steady and unsteady flows at grid resolution changes are investigated. Moreover, refinement criteria other than physics-based are formulated and examined. Additionally, the domain-decomposition method for parallel implementation on multi-processor architectures and the inter-processor communication associated with the high-order algorithm are discussed.

## 4.1 Parallel Block-Based AMR Algorithm

As previously mentioned in Sect. 1.3.2, the implementation of the proposed adaptive high-order CENO finite-volume algorithm is considered using the block-based AMR framework proposed and developed by Sachdev *et al.* [98], Gao and Groth [92, 112], and Gao *et al.* [155] for body-fitted mesh. Thus, a flexible block-based hierarchical data structure is used in conjunction with the CENO finite-volume scheme to facilitate automatic solution-directed mesh adaptation on two-dimensional body-fitted multi-block quadrilateral mesh. The method allows for the use of anisotropic mesh and is well suited to parallel implementation via domain decomposition. In contrast to the lower-order variant, the high-order mesh adaptation procedure described in this work requires the development of inter-block communication dependent on the order of solution accuracy and formulation of high-order

treatment of physical boundaries, boundary conditions and solution transfer between AMR grids. Aspects of the block-based AMR algorithm for multi-block quadrilateral mesh and second-order finite-volume scheme are described in the work by Sachdev *et al.* [98]. This approach has been already successfully applied to the prediction of various complex flows [92, 96, 113], and here it is extended to high-order accuracy.

In the block-based AMR algorithm considered herein, mesh adaptation is accomplished by the dividing and coarsening of appropriate solution blocks. In regions requiring increased cell resolution, a “parent” block is refined by dividing itself into four “children” or “offspring” (see Fig. 4.1(b)). Each of the four quadrants or sectors of a parent block becomes a new block having the same number of cells as the parent and thereby doubling the cell resolution in the region of interest. This process can be reversed in regions that are deemed over-resolved and four children are coarsened into a single parent block. The mesh refinement is constrained such that the grid resolution changes by only a factor of two between adjacent blocks and the minimum resolution is not less than that of the initial mesh.

A hierarchical tree-like data structure with multiple “roots”, multiple “trees”, and additional interconnects between the “leaves” of the trees is used to keep track of mesh refinement and the connectivity between solution blocks (see Fig. 4.1(a)). The blocks of the initial mesh are the roots of the forest which are stored in an indexed array data structure. Associated with each root is a separate “quadtree” data structure that contains all of the blocks making up the leaves of the tree created from the original parent blocks during mesh refinement. Each grid block corresponds to a node of the tree. Traversal of the multi-tree structure by recursively visiting the parents and children of solution blocks can be used to determine block connectivity. However, in order to reduce overhead associated with accessing solution information from adjacent blocks, the neighbours of each block are computed and stored, providing direct interconnects between blocks in the hierarchical data structure that are neighbours in physical space. One of the advantages of the hierarchical quadtree data structure is that it readily permits local mesh refinement. Local modifications to the multi-block mesh can be performed without re-gridding the entire mesh and re-calculating all solution block connectivities.

As described in previous chapters, the solution of the conservation equations by the CENO finite-volume method provides area-averaged solution quantities within quadrilateral computational cells and these cells are embedded in structured blocks consisting of  $N_{\text{cells}} =$

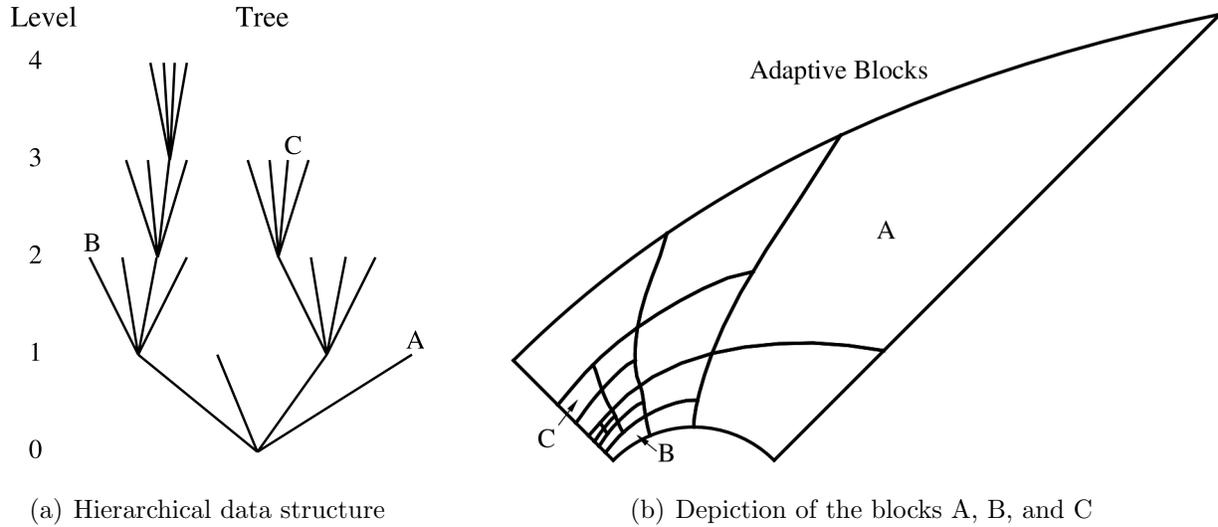


Figure 4.1: (a) Adaptive mesh refinement quad-tree data structure, and (b) associated self-similar solution blocks of the quadrilateral body-fitted mesh with different resolution levels.

$N_x \times N_y$  cells, where  $N_x$  and  $N_y$  are even, but not necessarily equal integer values. Solution data associated with each block are stored in indexed array data structures and it is therefore straightforward to obtain solution information from neighbouring cells within blocks.

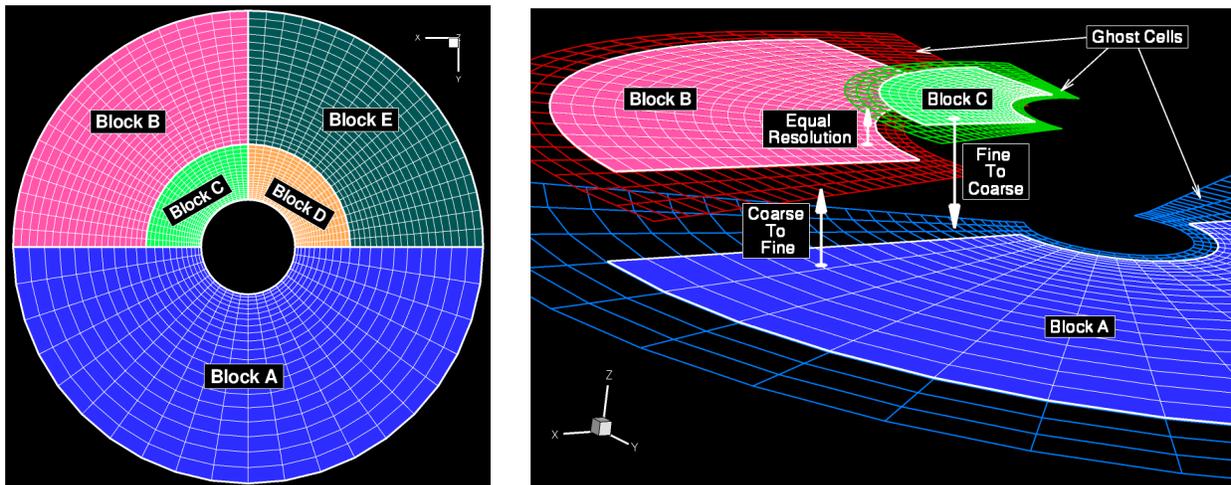
The hybrid CENO solution reconstruction procedure is used in conjunction with standard multi-grid-type restriction and prolongation operators to evaluate the solution on all blocks created by the coarsening and division processes, respectively. Details of the application of CENO reconstruction to multi-block mesh and the high-order solution transfer between AMR grids are given in Sect. 4.2 and Sect. 4.3, respectively.

In order that the finite-volume scheme can be applied to all blocks in a more independent manner, some solution information is shared between adjacent blocks having common interfaces. This information is stored in additional layers of overlapping “ghost” cells associated with each block. The number of ghost cells required depends on the accuracy of the scheme and follow the same guidelines laid out in Sect. 3.6.3. An illustration of the additional layers of ghost cells for a refined O-grid mesh and the types of existing inter-block communication are depicted in Fig. 4.2. As can be seen in Fig. 4.2(b), at interfaces between blocks of equal resolution, such as B and C, a one to one correspondence exists and thus, the ghost cells are simply assigned the solution values associated with the appropriate interior cells of the adjacent blocks. At resolution changes, such as those occurring between the coarse block

A and the refined blocks B and C, as shown in Fig. 4.2(b), either a coarse to fine or a fine to coarse solution transfer is required depending on whether the ghost cells belong to the coarse or the fine adaptive block. In these situations, restriction and prolongation operators, similar to those used in block coarsening and division, are employed to evaluate the ghost cell values, the design of which is based on solution conservation and accuracy rationales. Within the AMR approach, additional inter-block communication is also required at interfaces with resolution changes to strictly enforce the flux conservation properties of the finite-volume scheme [86, 87]. In particular, the interface fluxes computed on more refined blocks are used to correct the interface fluxes computed on coarser neighbouring blocks and ensure the solution fluxes are conserved across block interfaces.

## 4.2 CENO Scheme on Body-Fitted Multi-Block Mesh

The high-order CENO algorithm described in the previous chapter for a single block is extended herein to multi-block meshes by storing in each constituent solution block enough



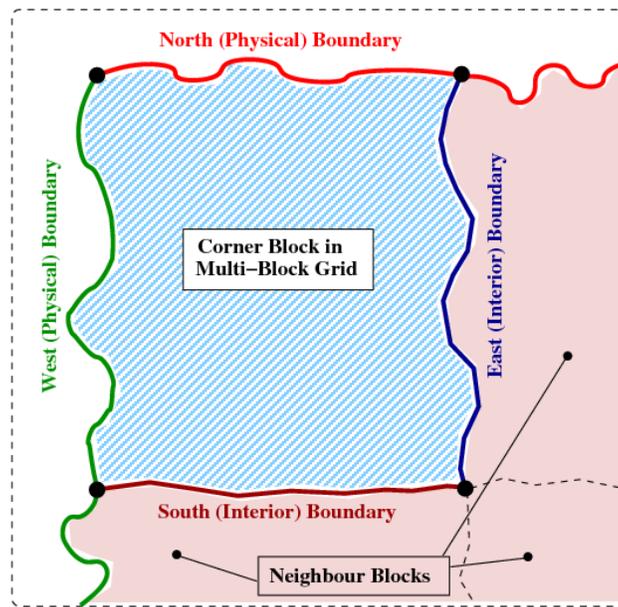
(a) Planar view of refined O-grid mesh with one coarse and four refined blocks.

(b) Exploded view of the blocks A, B and C, showing the overlapping ghost cell layers of each block and representing schematically the types of information transfer occurring between interior and ghost cells.

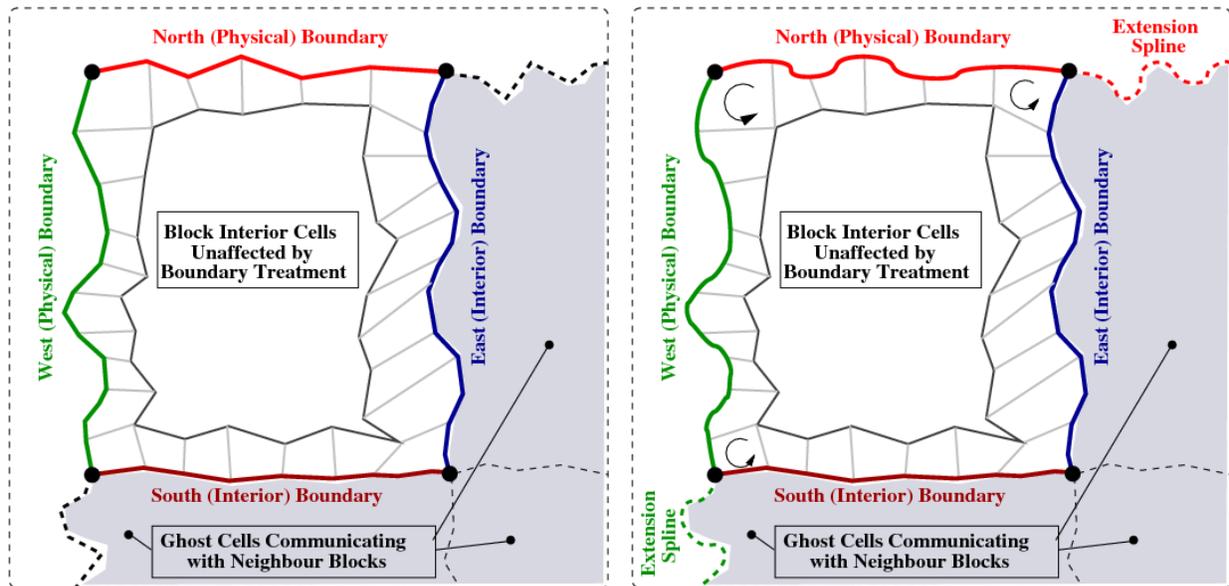
Figure 4.2: (a) Illustration of refined multi-block O-grid and (b) associated types of inter-block communication (i.e., fine to coarse, coarse to fine, and equal resolution).

information to allow the performance of the high-order solution reconstruction and residual calculation in each computational cell without intermediate data transfer from neighbouring blocks. Since there is no inter-block communication during the high-order residual evaluation, the CENO algorithm can be directly applied to each block independently in the way described in Sect. 3.6. The information incorporated in the overlapping layers of ghost cells is in general sufficient to allow an independent high-order reconstruction and residual evaluation in each cell of an interior block, but additional information must be typically exchanged between the neighbouring blocks near physical boundaries to carry out the reconstruction process. The additional exchanged data is primarily related to the geometry of the curved boundaries and the boundary conditions imposed in the adjacent blocks. Note that this data is not exchanged at every iteration and/or time step. In fact, it is only updated when the multi-block topology near physical boundaries is modified, such as during mesh adaptation.

To see why the exchange of this additional information is required for the application of the high-order solution procedure to a boundary block, it is instructive to look at the differences depicted in Fig. 4.3 between a low- and a high-order representation of a corner block in a multi-block mesh. As can be seen in Fig. 4.3(b), a low-order representation of the “true” physical domain boundaries, shown in Fig. 4.3(a), requires only information about the location of all grid nodes along the boundary, including those associated with the ghost cells. This is a consequence of the fact that the faces of the quadrilateral control volumes can be obtained by connecting the adjacent nodes with straight segments. Thus, this knowledge is typically sufficient to develop a solution procedure up to 2nd-order accuracy. In contrast, the edges of the high-order boundary elements shown in Fig. 4.3(b) cannot be constructed based only on the node locations and information for the boundary splines is required. The extra pieces of information required by the high-order reconstruction procedure applied to ghost cells adjacent to physical boundaries are provided by storing extension (ghost) boundaries. Hence, ghost boundaries provide local information about the nearby representation of physical boundary splines shared by adjacent blocks, and this role is very similar to that of the overlapping layers of ghost cells. Not only are these ghost boundaries useful for computing the geometric properties of the ghost cells, but they also provide essential information about the way in which the BCs are imposed and consequently, they influence the choice of the reconstruction supporting stencil in the neighbourhood and the type of the reconstruction (i.e., constrained or unconstrained) in their adjacent ghost cells.



(a) “True” physical domain



(b) Low-order domain representation

(c) High-order domain representation

Figure 4.3: (a) Illustration of the “true” domain and its division into multiple blocks; and (b) the low- and (c) the high-order mesh representations for the corner block. Straight line segments are sufficient for representing interior block boundaries. The extension (ghost) boundaries provides complete information about the properties of the neighbouring splines.

As anticipated, the main complication arising in the high-order CENO reconstruction on multi-block mesh is the formulation of an algorithm that provides *identical* or *consistent* supporting stencils for the overlapping ghost and interior cells of adjacent blocks near physical boundaries for all possible combinations of methods to impose BCs (i.e., via ghost cells or constrained reconstruction). To make this point clear, let's assume that the west block boundary in Fig. 4.3(c) requires constrained reconstruction (i.e., an *opaque and constrained* boundary) and its south extension boundary (i.e., the west block boundary of the south neighbour) requires BCs specified by ghost cells (i.e., a *transparent and unconstrained* boundary). Obviously, both ghost and interior cells in the proximity of the SW corner of this block require a special treatment. Their reconstruction supporting stencil must be chosen to obey the conditions imposed by the nearby splines and at the same time must remain a valid stencil. Note also that the same conditions must be mirrored in the south neighbour block where the SW corner is substituted for the NW cell. Depending on the type of the boundaries that meet in a corner there are 16 distinctive cases that can exist but after the non-realizable situations are eliminated only 8 are left. These realizable combinations are illustrated schematically for a SW block corner in Fig. 4.4.

It should be clear from Fig. 4.4 that the top four cases do not present any problems for formulating a valid reconstruction stencil if the rules outlined in Sect. 3.6.3 for opaque boundaries are followed (i.e., do not use cells beyond the constrained boundary and enlarge the stencil as required in the opposite direction to that of the boundary). The bottom four cases are made consistent by setting the unconstrained extension boundary closest to the constrained boundary as *unconstrained and opaque* boundary and follow similar rules as those

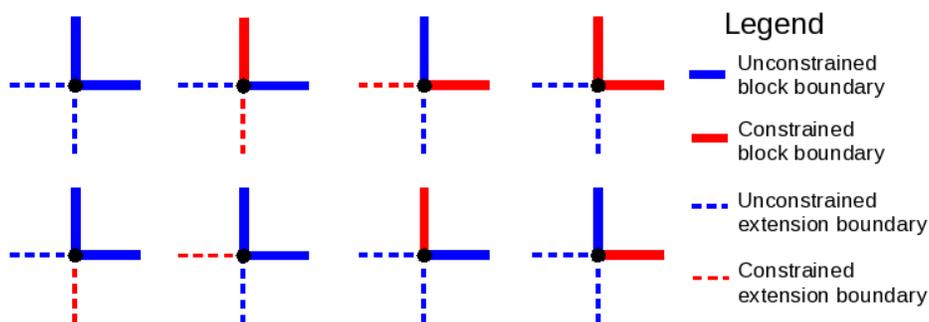


Figure 4.4: *Representation of the valid combinations of boundary types for the SW block corner.*

previously mentioned. Additionally, the process of enlarging the stencil is subject to ensuring that *identical* stencils can be obtained in the adjacent blocks. The implementation improves the computational efficiency of the scheme on blocks with constrained reconstruction by marking the type of the reconstruction (i.e., no reconstruction, central stencil reconstruction, modified stencil unconstrained reconstruction and constrained reconstruction) in each constituent computational cell as a preprocessing step.

### 4.3 High-Order Solution Transfer on AMR Grids

In the refinement process of a grid block the geometry of the newly created cells belonging to the offspring is obtained by dividing the domain of each coarse interior cell into four fine cells denoted as *I*, *II*, *III* and *IV*, as illustrated in Fig. 4.5. The new nodes represent the midpoint of each coarse face and the weighted average of the four coarse corners. Note that a body-fitted mesh is obtained herein by ensuring that the inserted points are on the boundary splines representing the physical curved boundaries and this approach is taken for both low- and high-order boundary representations. Note also that the utilization of high-order boundary elements for curved boundaries, as seen in Fig. 4.5(b), ensures naturally the equality of the coarse element area,  $A_{\Omega}$ , and the summation area of the offspring (i.e.,  $A_I + A_{II} + A_{III} + A_{IV}$ ) thereby avoiding the necessity for correction algorithms during

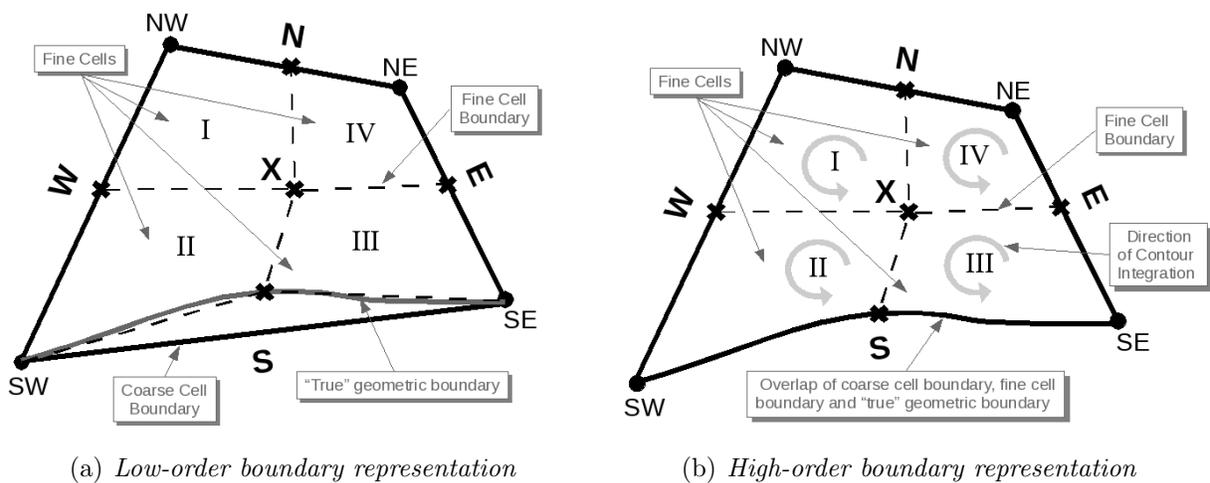


Figure 4.5: *Depiction of a coarse cell division into four fine cells for low- and high-order block boundary elements.*

solution distribution procedure. Based on this property the following solution conservation equation can be written for each average conserved solution quantity,  $\bar{u}_\Omega$ , of the coarse cell

$$\bar{u}_\Omega A_\Omega = \bar{u}_I A_I + \bar{u}_{II} A_{II} + \bar{u}_{III} A_{III} + \bar{u}_{IV} A_{IV} . \quad (4.1)$$

In order to maintain the high-order solution accuracy for combinations of CENO algorithm with the aforementioned block-based AMR approach accurate solution transfer operators must be formulated. Note that the procedure for correcting the fluxes across block interfaces with different grid resolution always ensures the conservation properties of the numerical scheme but it does not necessarily ensure high-order accuracy of the solution. Therefore, accurate solution transfer between the overlapping interface cells is required for steady-state calculations and, additionally, between the interior cells of the parent and children blocks in unsteady flow AMR simulations.

The restriction and prolongation operators used herein are derived from the conservation property of the solution content for overlapping cell domains expressed as Eq. (4.1). Additionally, a high-order accurate solution transfer from the coarse cell to the fine cells (i.e., prolongation) is provided here by incorporating information about the solution distribution over the coarse cell domain. In particular, to distribute the average solution quantity among offspring with high-order accuracy the high-order polynomial reconstructions of all solution variables in the coarse cell are integrated over the domain of each fine cell, which reads for the domain  $I$

$$\bar{u}_I = \frac{1}{A_I} \iint_{\mathcal{A}_I} u_{\text{coarse}}^k(\vec{X}) dx dy , \quad (4.2)$$

where the surface integral is computed exactly with an appropriate-order Gauss quadrature contour integration as indicated in Fig. 4.5(b) and explained in Appendix A.

It should be clear that the prolongation operator described above automatically satisfies Eq. (4.1) if the interpolant  $u_{\text{coarse}}^k$  represents the reconstruction of a conserved quantity of the governing equation set. However, this type of solution distribution does not enforce explicitly the positivity conditions, such as non-negative pressure and density, and consequently, in case they are violated, a direct injection approach (i.e.,  $\bar{u}_I = \bar{u}_{II} = \bar{u}_{III} = \bar{u}_{IV} = \bar{u}_\Omega$ ) is used herein.

For reconstructions performed with derived quantities such as primitive solution states,  $\bar{W}$ , the approach considered in this work is to compute the average conserved state of each fine cell by correcting the average quantities obtained from integration of the coarse recon-

struction polynomials with any non-zero solution error distributed uniformly over the coarse cell domain. Thus, the average conserved solution state for a fine cell  $s$  is given by

$$\bar{U}_s = \bar{U}'_s + \Delta\bar{U} = f(\bar{W}'_s) + \frac{1}{A_\Omega} \left[ \bar{U}_\Omega A_\Omega - \sum_{i=I}^{IV} f(\bar{W}'_i) A_i \right], \quad (4.3)$$

where  $f(W)$  represent the mapping function of the derived quantities into the conserved ones. As in the previous case, direct injection is used if non-physical values are encountered.

## 4.4 Refinement Criteria Based on High-Order Solution

In previous work, the coarsening and division of blocks was directed using multiple physics-based refinement criteria [92, 96, 98, 113]. In this work, an  $h$ -refinement criterion based on the solution smoothness indicator is defined and used to control refinement of the body-fitted multi-block AMR mesh. The form considered herein for the CENO-based refinement criterion is:

$$\mathcal{R}_c = e^{-\frac{\max(0, \mathcal{S})}{\mathcal{U}_s * \mathcal{S}_c}} \quad (4.4)$$

where  $\mathcal{U}_s$  is a scaling coefficient. Based on the range of  $\mathcal{S}$  and the above definition for the refinement criterion, it should be evident that  $\mathcal{R}_c$  always takes values in the range  $(0, 1]$ . Each block is then assigned the maximum  $\mathcal{R}_c$  value of all cells within the block and in order to decide which blocks are to undergo mesh refinement, refining and coarsening thresholds are used. The coefficient,  $\mathcal{U}_s$ , is introduced for convenience and provides an easy way of adjusting the values of the coarsening and refining thresholds relative to the value of the cutoff  $\mathcal{S}_c$ . For multiple reconstructed variables a unique  $\mathcal{R}_c$  is obtained by taking the maximum value among all or a reduced set of solution variables. Note that the proposed refinement criterion is non-dimensional and consequently, the relative importance of different variables for AMR can be easily assessed based only on their relative smoothness.

Hence, the solution accuracy can be significantly improved by directing mesh adaptation based on the proposed refinement criterion only as long as there is a tight correlation between the discretization error and the local solution smoothness as measured by the smoothness indicator. Note that many dominant local solution features, such as unresolved and/or poorly resolved smooth solution content, shear layers, and discontinuities/shockwaves, may exhibit such a correlation at least for a range of mesh resolutions, if not continuously as in the

case of solution discontinuities. Over resolving of non-smooth solution features (e.g., discontinuities) can be avoided by specifying a maximum allowable level of mesh refinements. Since many flows are characterized by a direct relation between the final accuracy of the solution and the level with which the aforementioned types of dominant solution features are resolved, it should be clear that using smoothness-based mesh adaptation, at least in the initial stage of the solution procedure, can appropriately identify regions required for mesh refinement. However, this may not necessarily be the case for a desired targeted solution accuracy, in which case it would be more appropriate to devise a combined mesh adaptation based on both smoothness and *a posteriori* error estimates, such as the one described below. Note that instead of competing with each other, the two approaches are rather complimentary. Smoothness-based adaptation can provide adequately resolved solutions for which error estimates are then reliable and the solution can be refined to the desired level of accuracy.

Another way to exploit the high-order solution reconstruction for adaptive mesh refinement is to construct *a posteriori* error estimates based on different  $p$ -th order polynomial approximations similar to the formulations proposed by Barth and Larson [156]. However, this approach has been only briefly explored in this work by assessing the effectiveness of estimating the error magnitude and location introduced by low-order reconstructions of analytical functions when the “true” function representation is replaced with a high-order polynomial interpolant. In particular, it has been confirmed that cubic and quartic polynomials provide a reliable way of approximating the “true” analytical solution and can be effectively used for truncation error estimation in conjunction with linear or quadratic reconstructions.

## 4.5 Domain Decomposition and Parallel Implementation

Although the block-based AMR approach described above is somewhat less flexible and incurs some inefficiencies in solution resolution as compared to a cell-based approaches (i.e., for the same solution accuracy, generally more computational cells are introduced in the adapted grid), the block-based method offers many advantages over cell-based techniques when parallel implementation of the solution algorithm is considered and computational performance issues are taken into account.

Domain decomposition involves decomposing a computational mesh and distributing the sub-meshes among the processors of a multi-processor architecture. In particular, the multi-

block quadrilateral mesh and quadtree data structure lends itself naturally to geometric decomposition as the solution blocks can be easily distributed to the processors, with more than one block permitted on each processor as shown in Fig. 4.6. Thereby, it should enable efficient and scalable implementations of the CENO FV scheme on distributed-memory homogeneous multi-processor (identical processors) architectures, as previously demonstrated by various 2nd-order algorithms [92, 96, 98, 113] and illustrated by the parallel efficiency study shown in Sect. 5.8.2 of numerical results.

The parallel implementation of the block-based AMR scheme has been carried out using the C++ [157] version of the MPI (message passing interface) library [158]. Additionally, C++ templates are used extensively throughout our parallel AMR and high-order CENO implementation thereby providing a flexible and unique framework for all flow solvers, avoiding code redundancy and improving software maintainability. Inter-processor communication is mainly associated with block interfaces and involves the exchange of ghost-cell solution values and conservative flux corrections at every stage of the multi-stage time integration procedure. Consequently, the amount of exchanged data between processors is proportional to the total number of ghost cells in a block whereas the computational work per block is varying with the number of interior cells,  $N_{\text{cells}} = N_x \times N_y$ . As the ratio of the number of ghost cells to the interior cells is decreasing inversely proportional with the square of  $N_x$  and  $N_y$ , it results that the parallel efficiency of the high-order scheme is very dependent on the number of interior cells and it rapidly increases as the blocks get larger and larger.

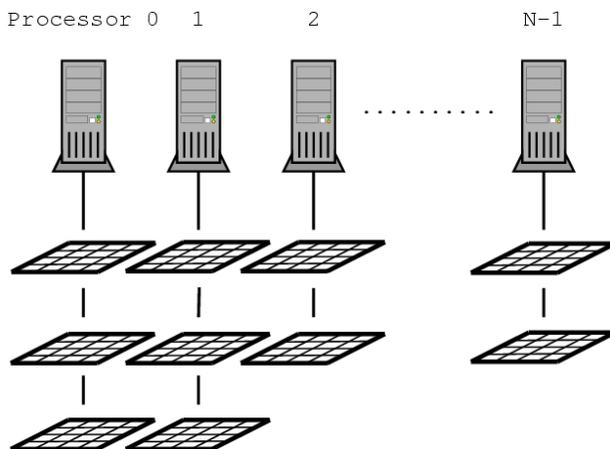


Figure 4.6: *Block distribution among the processors of a multi-processor architecture.*

To improve the efficiency of the parallel communication, message passing of the ghost-cell values and flux corrections is performed by consolidating inter-processor messages.

## 4.6 Computational Resources Used for Numerical Simulations

Computational resources for performing all of the calculations reported herein were provided by the SciNet High Performance Computing Consortium at the University of Toronto and Compute/Calcul Canada through funding from the Canada Foundation for Innovation (CFI) and the Province of Ontario, Canada.

All of the computations were carried out on two clusters, the General Purpose Cluster (GPC) and the Tightly Coupled System (TCS). The GPC consists of Intel Xeon E5540 (2.53 GHz) nodes whereas the TCS cluster is based on IBM Power6-575 (4.7 GHz) nodes. Both systems are connected with a high-speed, low-latency, non-blocking 4x-DDR Infiniband interconnect.

# Chapter 5

## NUMERICAL RESULTS

*“Results! Why man, I have gotten a lot of results. I know several thousand things that won’t work.”*

by Thomas A. Edison (1847-1931), American inventor

To demonstrate the capabilities of the proposed high-order CENO scheme described in the previous chapters, numerical results are presented in this chapter for solution reconstruction in one- and two-space dimensions, as well as for various one- and two-dimensional flow problems governed by the advection-diffusion, the Euler and the Navier-Stokes equations. Moreover, the predictive capabilities of the CENO scheme are illustrated with results on both fixed and AMR meshes. Additionally, in order to facilitate the interpretation of the numerical results, the next section provides a summary of the methods used for analyzing the proposed high-order algorithm.

### 5.1 Assessment of Properties of CENO Discretization

*“Wir müssen wissen. Wir werden wissen.”*

*We must know. We will know.*

by David Hilbert (1862-1943), German mathematician (his tombstone epitaph)

The process of *validation* and *verification* of CFD solutions and techniques currently represents an important component of establishing the *credibility* of newly proposed CFD algorithms in the context of computational science and engineering. Moreover, with powerful computers becoming more common, credibility of simulation techniques must be established through rigorous and systematic error quantification [114, 159]. The purpose of this section is to describe the investigation techniques used for the analysis of the CENO algorithm and for determining the discretization properties of this numerical scheme.

The verification of the proposed CENO algorithm and the analysis of its discretization properties have been thoroughly investigated over the course of this research following the methods established in the literature, such as those described by Roache [114] and Zingg [160]. Thus, the techniques considered in this work include: comparisons between high- and low-order solutions for the same grid resolution; systematic grid convergence studies carried out for a large range of problems based on either analytic solutions, solution invariants or accurate converged solutions obtained on very fine meshes; solution visualization with high-resolution; quantification of errors on highly distorted meshes; numerical investigations regarding the recovery of mathematical principles; and, last but not least, comparisons with the prediction of other similar techniques and/or experimental values. Several details regarding the application of the aforementioned techniques in this work are now discussed.

Accurate visualization is crucial for assessing the solution quality of CFD simulations. It is especially important for high-order methods, where fine solution details reflected in the simulation data may be lost in the process of visualization with low-resolution methods. It is a very common problem that sharp features captured by high-order algorithms get “smeared out” by the low interpolation techniques used in most of the current plotting packages, determining erroneous interpretations of the numerical results and the capabilities of the numerical scheme [161, 162]. At the very least, one should plot the simulation data with higher mesh resolution in order to reproduce the high-order solution variations with increased fidelity. Another choice is to employ more accurate and efficient, but also more complex, visualization techniques that both reproduce and exploit the mathematical structure of high-order methods, such as the ray-tracing procedure described by Meyer *et al.* [162]. The approach considered in this work was to enhance the high-order solution visualization by including additional plotting points. For pure reconstruction studies, a subgrid mesh of arbitrary resolution was superimposed over the geometry of each quadrilateral element and thus, very accurate visualizations were possible by evaluating the polynomial reconstruction at the nodes of the subgrid mesh. However, such an approach is not practical when performing realistic flow simulations due to the generation of large volumes of data and therefore, the compromised solution adopted herein is to plot the reconstruction values at *nine* points in each quadrilateral element, which were chosen as follows: the four vertices, the four face midpoints and the centroid. Note that even with the relatively reduced set of points used here, it is still possible to visualize the solution discontinuities across element

interfaces and equally important, to judge “continuous to the eye” profiles more accurately.

Accurate visualization represents only one element in analyzing high-order methods, and far more convincing results are provided by systematic grid convergence studies with error quantification [114, 160]. One of the requirements for performing accurate convergence studies is to generate sequences of refined meshes which to guarantee a uniform and smooth mesh-spacing variation throughout the domain. This condition has been satisfied in this work by either employing transfinite interpolation (TFI) algorithm [163] for mapping uniform grids of different resolution in a computational domain to the physical domain with irregular boundaries used for numerical studies, or by including finer nodes in between the nodes of the coarser mesh in a systematic way. Note that an alternative approach to the latter technique is to generate the sequence of meshes by coarsening, as outlined in [160]. Furthermore, the curved boundaries of the computational domains considered in this work are all defined with a sufficient number of control points at each grid resolution level so as to avoid any discretization errors that may occur due to inaccuracy of the geometry representation.

Critical to convergence studies is the error quantification process itself. Thus, for exact solutions and/or solution invariants the numerical error of the scheme was assessed in the  $L_1$ ,  $L_2$  and  $L_\infty$  norms as follows:

$$L_1 = |E|_1 = \frac{1}{\mathcal{A}_T} \sum_{i,j} \iint_{\mathcal{A}_{i,j}} \left| u_{i,j}^k(x, y) - u_{\text{exact}}(x, y) \right| da, \quad (5.1)$$

$$L_2 = |E|_2 = \left\{ \frac{1}{\mathcal{A}_T} \sum_{i,j} \iint_{\mathcal{A}_{i,j}} \left[ u_{i,j}^k(x, y) - u_{\text{exact}}(x, y) \right]^2 da \right\}^{1/2}, \quad (5.2)$$

$$L_\infty = |E|_\infty = \max_{i,j} \left\{ \frac{1}{\mathcal{A}_{i,j}} \iint_{\mathcal{A}_{i,j}} \left| u_{i,j}^k(x, y) - u_{\text{exact}}(x, y) \right| da \right\}, \quad (5.3)$$

where  $i$  and  $j$  indices describe all interior cells of the structured mesh,  $u_{i,j}^k(x, y)$  represents the polynomial reconstruction of the predicted solution and  $u_{\text{exact}}(x, y)$  is the exact solution at the error calculation point. Note that the error norms defined in Eq. (5.1)–(5.3) can be applied to any exact solution field, regardless of whether it represents a reconstructed variable or a derived quantity of the variables that are reconstructed. An example of solution invariant used for accuracy assessment herein is provided by the constancy of the entropy for isentropic flows. The accurate calculation of the surface integrals in Eq. (5.1)–(5.3) is performed herein with the adaptive Gauss-Lobatto quadrature procedure outlined in Appendix A, in which at least eight exact digits are sought in the integration result.

For problems in which accuracy assessment is based on errors in functional estimation, such as prediction of aerodynamic lift and drag coefficients, accurate and efficient functional estimation is obtained by integrating the reconstructions of relevant dependent flow quantities, such as fluid pressure for lift and/or normal velocity gradient for drag, along the specified geometric surfaces, which in general are curved. The curvilinear function integration procedure used herein is a Gauss quadrature rule of five points for each quadrilateral edge, the details of which are provided in Appendix A.

The remainder of this chapter presents a large volume of numerical experiments which provide credibility to the proposed CENO algorithm and illustrate the predictive capabilities and performance of the current implementation for problems in one and two-space dimensions.

## 5.2 Function Reconstruction on Fixed Mesh

Several representative examples of one- and two-dimensional function reconstruction have been included herein to demonstrate the accuracy and robustness of the CENO reconstruction approach which lies at the core of the proposed high-order finite-volume scheme. Moreover, comparisons between the CENO and the original ENO reconstructions are provided for both smooth and discontinuous one-dimensional functions.

### 5.2.1 One-Dimensional Smooth Trigonometric Function

To assess the accuracy of the hybrid CENO scheme, reconstructions of the smooth trigonometric function  $f(x) = \sin(2x) + 2\cos(x)$  on cell-centred one-dimensional uniform meshes are compared to the exact solution. In Fig. 5.1, the cubic ( $k=3$ ) reconstruction obtained with 32 computational cells for  $-5 < x < 5$  is shown together with the exact function. It can be seen in the figure that this smooth function features multiple extrema and inflection points which can be accurately captured only with polynomials of appropriate order. Low-order methods cannot properly resolve such solution features. The 3rd-order CENO reconstruction resolves properly all smooth extrema of the function on the current mesh and consequently, the reconstructed polynomial is deemed smooth in every computational cell.

The  $L_1$ ,  $L_2$ , and  $L_\infty$  norms of the reconstruction error associated with hybrid CENO

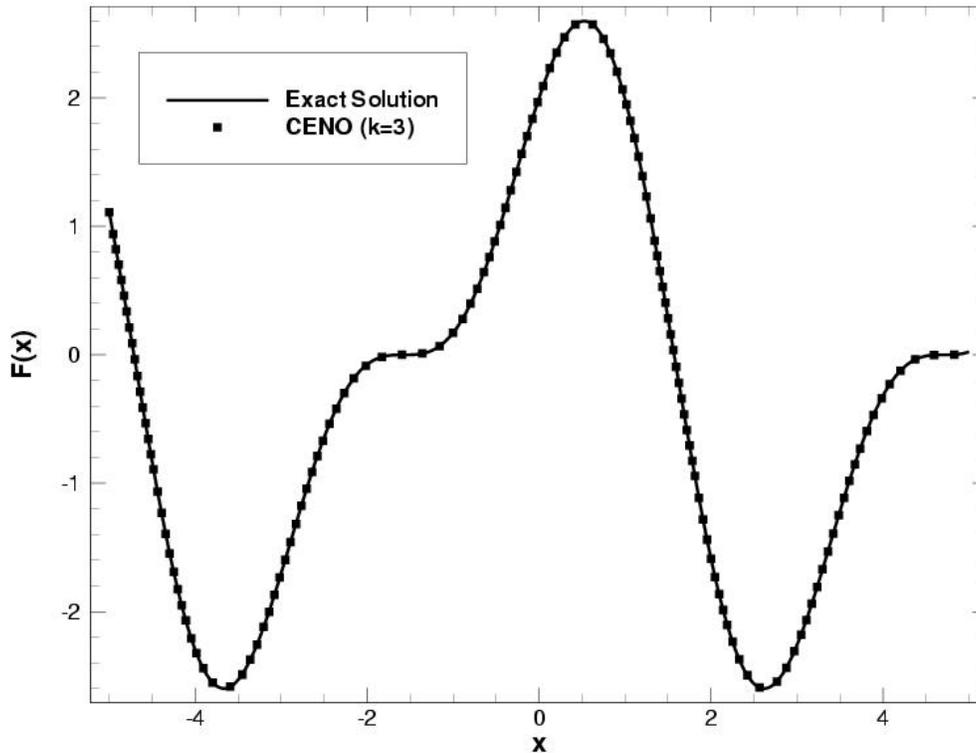
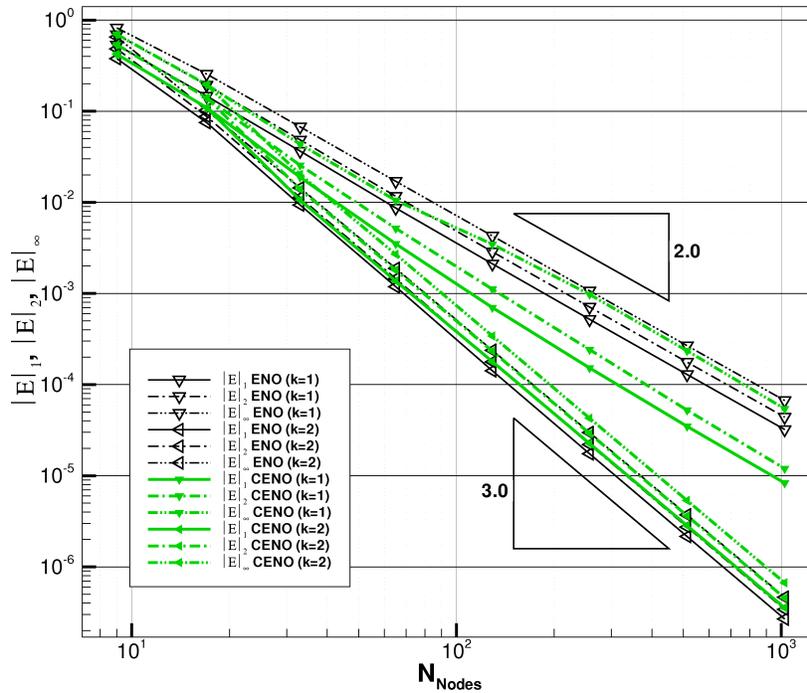
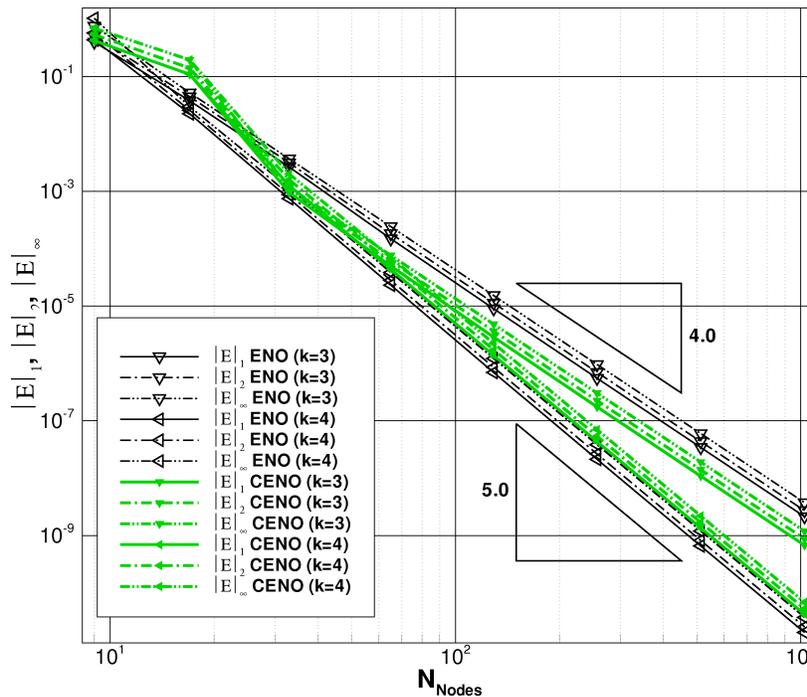


Figure 5.1: *Piecewise cubic ( $k=3$ ) hybrid CENO reconstruction of the smooth trigonometric function  $f(x) = \sin(2x) + 2\cos(x)$  on a one-dimensional uniform mesh of 32 computational cells is compared to the exact function for  $-5 < x < 5$ .*

reconstruction are shown in Fig. 5.2(a) and Fig. 5.2(b). Results are given for piecewise linear ( $k=1$ ), quadratic ( $k=2$ ), cubic ( $k=3$ ), and quartic ( $k=4$ ) reconstruction and compared to reconstruction errors obtained using the original ENO scheme of Harten *et al.* [18]. It is evident that the reconstruction errors of the CENO scheme are quite comparable to those of the ENO approach, and the expected theoretical asymptotic convergence rates of the 2nd-, 3rd-, 4th-, and 5th-order accurate methods are achieved (the order of accuracy of  $k$ -exact CENO schemes is  $k+1$  for smooth resolved functions). Note also that for under-resolved meshes, the switching to limited linear reconstruction employed by the CENO scheme as a mechanism to preserve monotonicity produces absolute errors that are similar to those obtained by selecting the smoothest stencil in the ENO approach.



(a) Error norms for piecewise linear and quadratic



(b) Error norms for piecewise cubic and quartic

Figure 5.2:  $L_1$ ,  $L_2$ , and  $L_\infty$  norms of the reconstruction errors for ENO and hybrid CENO reconstructions of the function shown in Fig. 5.1 as a function of the number of computational cells. (a) Piecewise linear ( $k=1$ ) and quadratic ( $k=2$ ) reconstructions, and (b) piecewise cubic ( $k=3$ ) and quartic ( $k=4$ ) reconstructions are shown.

### 5.2.2 One-Dimensional Smooth Exponential Function with Large Gradients

A similar set of numerical results are shown in Fig. 5.4 for the reconstruction of the smooth function  $f(x) = e^{-4x} \sin(5x)$  on the domain  $-5 < x < 5$ . Results are depicted for cubic ( $k = 3$ ) and quartic ( $k = 4$ ) CENO reconstructions and compared to ENO reconstructions. This function features a very large peak near  $x = -4.5$  (refer to Fig. 5.3) which determines large oscillations to non-monotone reconstructions when it is under-resolved as shown in Fig. 3.3. The proposed CENO scheme is able to handle this feature in an accurate and also robust manner. For meshes containing less than 40-50 cells, the function is under-resolved and monotonic piecewise linear reconstruction of the peak is obtained. As more cells are added, the function is then resolvable on the mesh and a smooth 4th-order reconstruction of the function is achieved with the expected asymptotic convergence rate of this 5th-order accurate scheme. Again, the reconstruction errors of the CENO scheme are very comparable to those of the ENO approach.

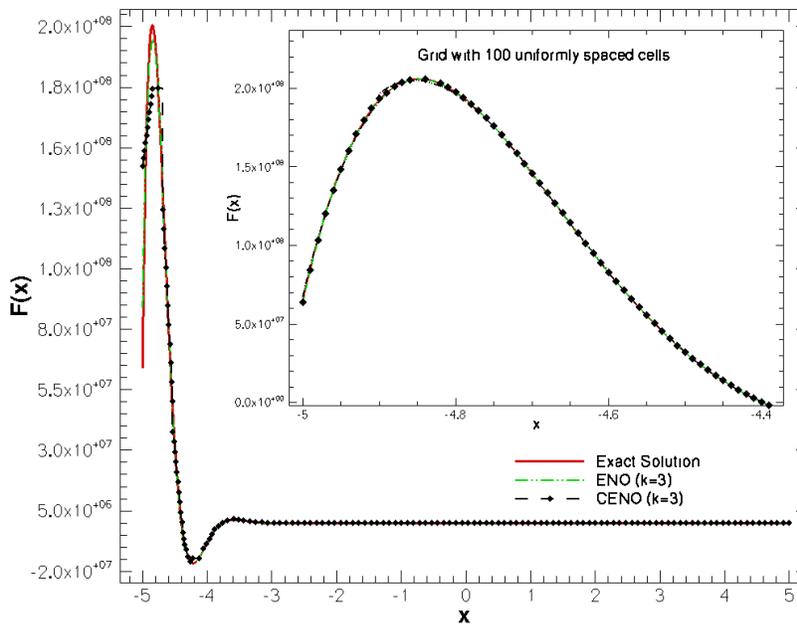


Figure 5.3: *Reconstruction of  $f(x) = e^{-4x} \sin(5x)$  on cell-centred one-dimensional uniform mesh for  $-5 < x < 5$  containing 64 and 100 computational cells. Results obtained using piecewise cubic ( $k=3$ ) ENO and hybrid CENO schemes are compared to the exact function.*

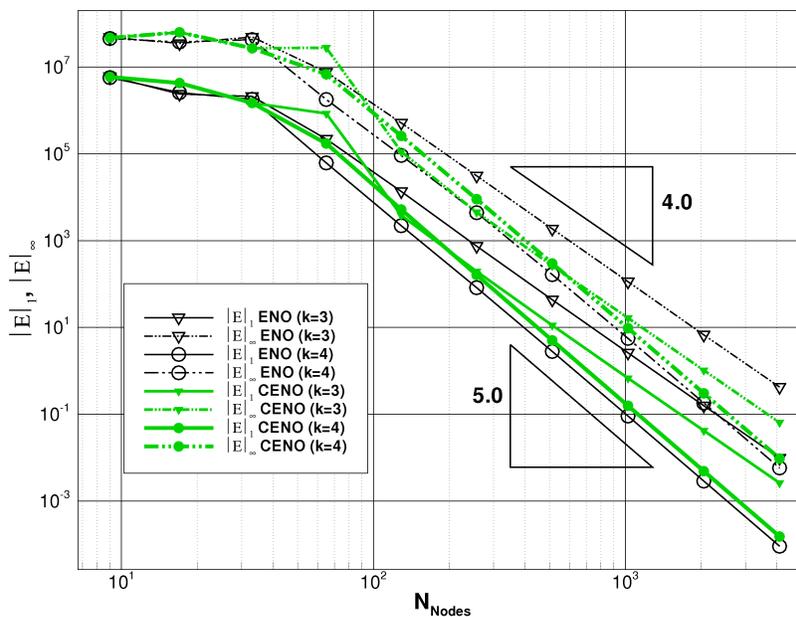


Figure 5.4:  *$L_1$  and  $L_\infty$  norms of the reconstruction errors for ENO and hybrid CENO reconstructions of the function shown in Fig. 5.3 versus the number of computational cells (piecewise cubic ( $k=3$ ) and quartic ( $k=4$ ) reconstructions are shown).*

### 5.2.3 One-Dimensional Discontinuous Step Function

The robustness of the CENO approach is now illustrated by considering the reconstruction of a discontinuous step function on a one-dimensional uniform mesh for  $-1 < x < 4$  containing 11 computational cells. Results obtained using piecewise linear ( $k = 1$ ), quadratic ( $k = 2$ ), cubic ( $k = 3$ ), and quartic ( $k = 4$ ) ENO and hybrid CENO schemes are compared to a piecewise constant representation and the exact function (refer to Fig. 5.5). It is quite evident that the discontinuous solution is not resolvable on any mesh and the hybrid CENO scheme is able to recognize this and reverts to a monotonic piecewise linear representation of the solution. Note that the solutions obtained with the CENO reconstruction for all reconstruction orders near discontinuity overlap each other, which is a consequence of the switching to a limited linear reconstruction in that region.

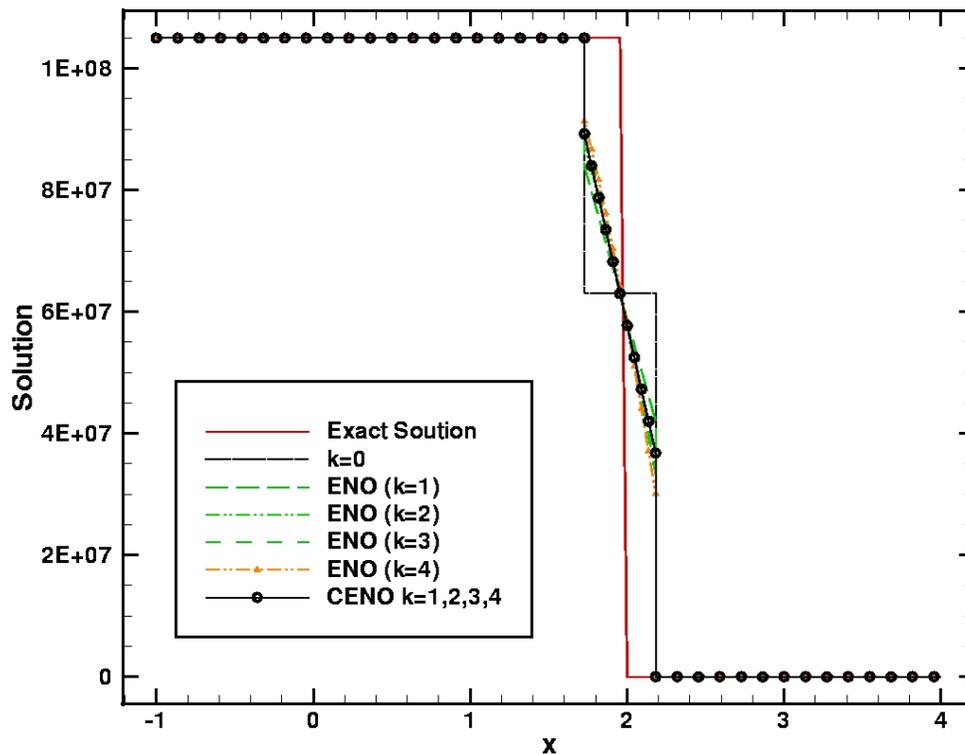


Figure 5.5: *Reconstruction of a discontinuous step function on a one-dimensional uniform mesh for  $-1 < x < 4$  containing 11 computational cells. Results obtained using piecewise quadratic ( $k = 2$ ), cubic ( $k = 3$ ), and quartic ( $k = 4$ ) ENO and hybrid CENO schemes are compared to a piecewise constant representation and the exact function.*

### 5.2.4 Two-Dimensional Smooth Trigonometric Function

The properties of the CENO reconstruction for two-dimensional smooth solution variations are now illustrated with the trigonometric function  $u(x, y) = 1.1 + \cos(\pi x^2 + 4\pi y)$  and the domain defined by the Ringleb's flow [164] solution between the streamlines corresponding to  $k = 0.75$  and  $k = 1.5$ , and the iso-velocity contour  $q = 0.5$  (see Sect. 5.6.3 for more details about the Ringleb's flow). The exact solution for this problem and the quartic ( $k=4$ ) CENO reconstructed solution on a mesh with  $40 \times 40$  cells are shown in Fig. 5.6. As can be easily observed, the quartic CENO reconstruction captures all the smooth extrema of the function very well on this relatively coarse mesh which is shown in Fig. 5.7(a), generating a solution profile that is visually indistinguishable from the exact solution. Note also that the solution gradients are not at all aligned with the mesh in this test.

To quantify the errors of the CENO reconstruction procedure for this problem convergence studies have been carried out with different parameters and two types of meshes ranging from  $10 \times 10$  to  $160 \times 160$  computational elements. The first sequence of meshes was represented by regular grids similar to the one shown in Fig. 5.7(a) whereas the second sequence was generated by disturbing randomly the nodes of the meshes in the first category. The disturbed mesh with  $40 \times 40$  computational cells used for this study is shown in Fig. 5.7(b). To evaluate the accuracy of the CENO procedure the cubic ( $k = 3$ ) and quartic ( $k = 4$ )

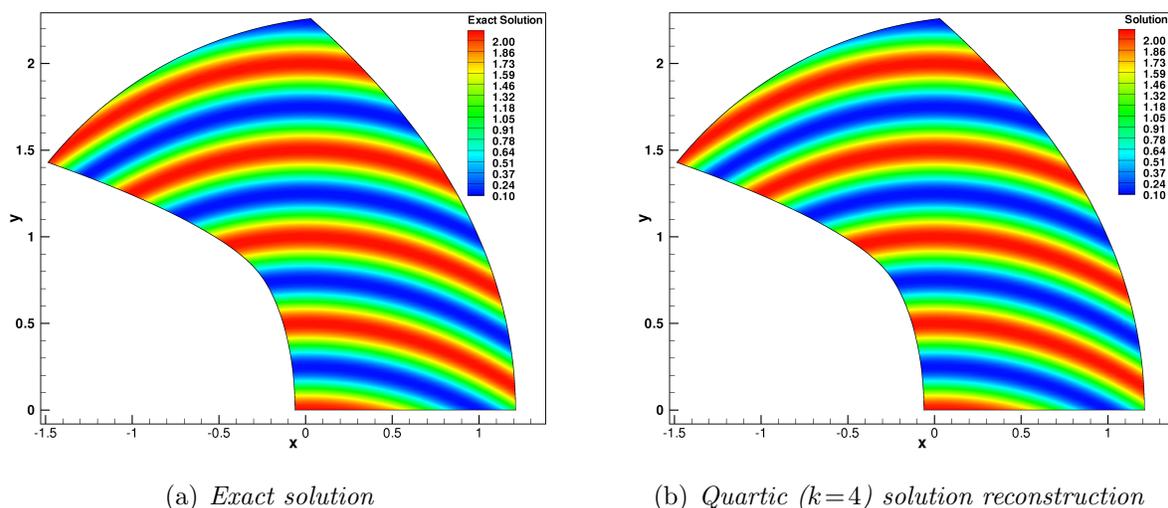


Figure 5.6: *Exact solution of function  $u(x, y) = 1.1 + \cos(\pi x^2 + 4\pi y)$  and the quartic solution reconstruction on a regular mesh with  $40 \times 40$  computational cells.*

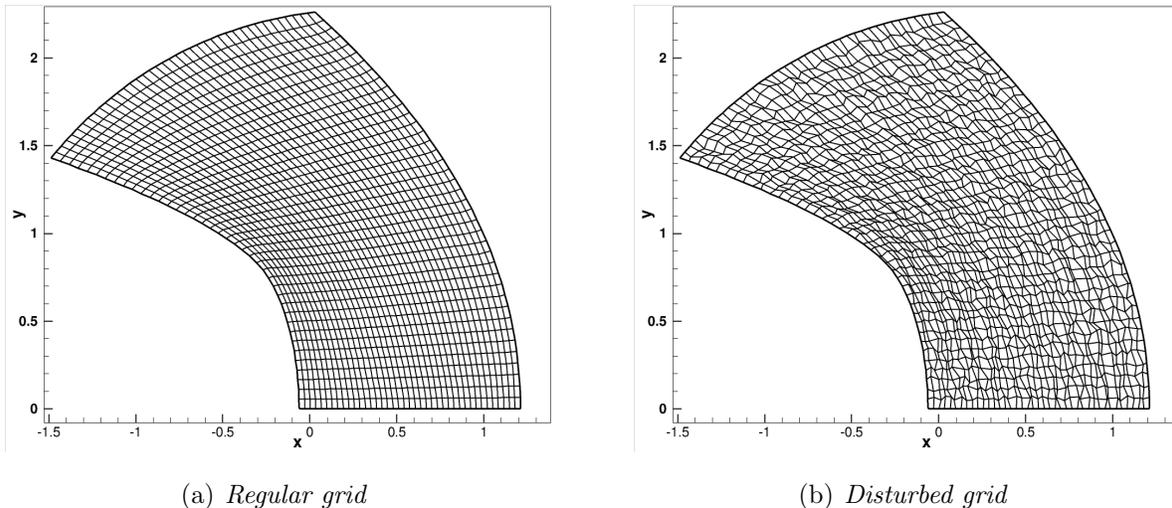
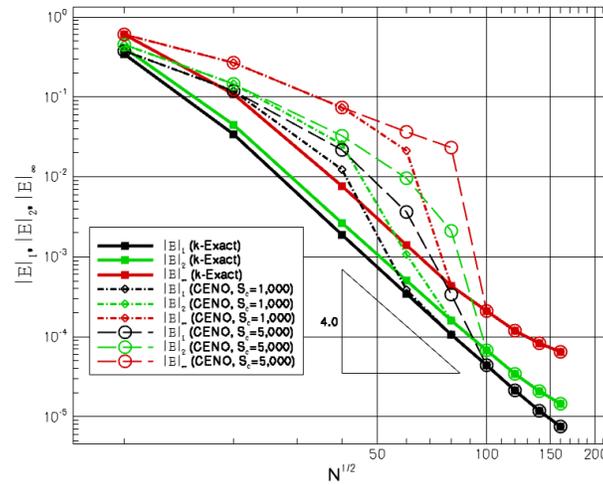


Figure 5.7: The regular and disturbed body-fitted meshes with  $40 \times 40$  cells used in the convergence study of the function depicted in Fig. 5.6.

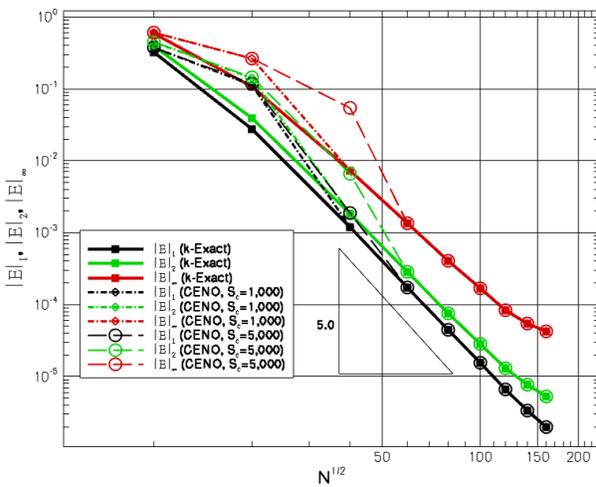
reconstructions have been considered in three cases: A) using only  $k$ -exact reconstruction, B) using CENO reconstruction with a pass/no-pass cutoff value  $\mathcal{S}_c = 1,000$  and C) using CENO reconstruction with  $\mathcal{S}_c = 5,000$ . In Fig. 5.8 the corresponding convergence history in all these situations is presented for  $L_1$ ,  $L_2$ , and  $L_\infty$  error norms. The results verify that the theoretical convergence orders based on different error norms which are four for the cubic and five for the quartic reconstruction, respectively, are achieved at least in the asymptotic limit. Additionally, the error norms obtained on the disturbed meshes are very similar to those corresponding to the regular grids, although the absolute error is slightly larger, and they are both very close to the asymptotic convergence rates indicated with the triangles depicted in the figures.

The influence of the cutoff value,  $\mathcal{S}_c$ , can also be inferred from the plots. First, note that all considered reconstructions produce similar errors on a mesh with  $10 \times 10$  cells regardless of the cutoff value which confirms the fact that this mesh is well under-resolved. Next, a “transition” regime occurs in which not all cell reconstructions are considered resolved based on the cutoff value and differences between the  $k$ -exact and the CENO errors are observed. The extent of this transition regime is a function of the cutoff value but also of the order of the reconstruction, as can be seen by comparing the plots in Fig. 5.8(a) and (b). Finally, after a minimum resolution has been attained and all cell reconstructions are deemed to be smooth, the solution maintains only the character of the  $k$ -exact reconstruction and becomes

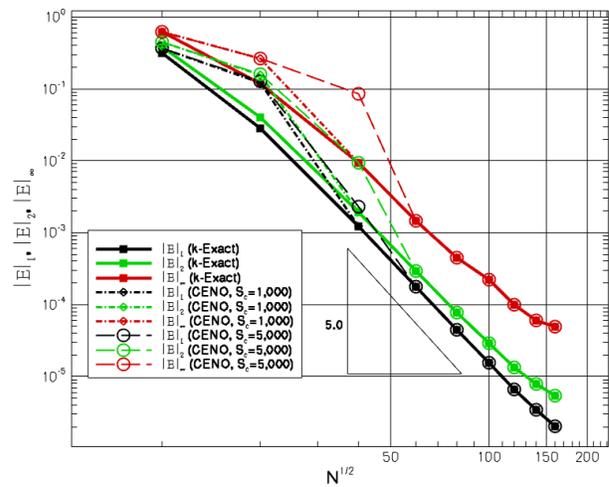
insensitive to cutoff values in the recommended range. Note that for the complex solution variation considered herein and a cubic reconstruction, the minimum resolution corresponds to around 60 to 100 cells in one-space dimension. However, the quartic reconstruction is able to capture more accurately the inflections of the tested function with fewer cells and consequently, enters into the final regime at a minimum resolution of around 40 to 60 cells.



(a) Cubic ( $k=3$ ) reconstruction on regular mesh



(b) Quartic ( $k=4$ ) reconstruction on regular mesh



(c) Quartic reconstruction on disturbed mesh

Figure 5.8: Cubic and quartic convergence history for  $L_1$ ,  $L_2$ , and  $L_\infty$ -norms of the reconstruction error of the function  $u(x, y) = 1.1 + \cos(\pi x^2 + 4\pi y)$ . Three cases are considered based on the cutoff value,  $S_c$ : A)  $k$ -exact, B)  $S_c=1,000$  and C)  $S_c=5,000$ .

### 5.2.5 Two-Dimensional Abgrall's Function

The reconstruction of a non-smooth function in two dimensions by the proposed hybrid CENO scheme is considered next. Reconstruction of the function examined in previous studies by Abgrall [165] and Ollivier-Gooch [30] is considered. A contour plot of Abgrall's function is given in Fig. 5.9(a). A similar contour plot of a fourth-order ( $k = 4$ ) CENO reconstructed solution of this function for a stretched  $80 \times 80$  non-uniform mesh is given in Fig. 5.9(b) and cross sections of the reconstructed solutions using the 2nd-order ( $k = 2$ ), 3rd-order ( $k = 3$ ), and 4th-order ( $k = 4$ ) CENO schemes for the  $80 \times 80$  non-uniform mesh are depicted in Fig. 5.10 and compared to the exact representation of the function as well as those for simple piecewise constant reconstruction. It is evident that the CENO reconstruction accurately represents the Abgrall's function in smooth regions while providing virtually non-oscillatory reconstructions near discontinuities.

A quantitative assessment of the error for this problem has also been performed. Figure 5.11 shows the variation of the  $L_1$  error norm corresponding to piecewise constant ( $k = 0$ ), linear ( $k = 1$ ) and cubic ( $k = 3$ ) reconstructions for a sequence of Cartesian meshes ranging from  $50 \times 50$  to  $400 \times 400$  computational cells. The results show that, regardless of the reconstruction order, the CENO reconstruction procedure always generates a first-order

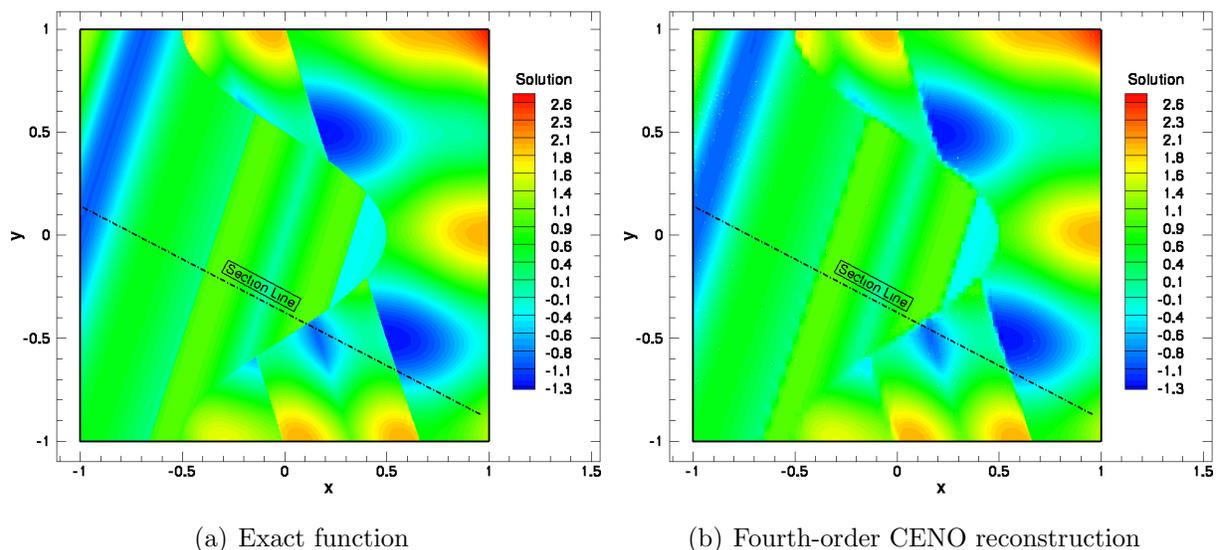


Figure 5.9: (a) The exact solution of the non-smooth function of Abgrall; (b) contours of a fourth-order ( $k = 4$ ) CENO reconstructed solution for a stretched  $80 \times 80$  non-uniform mesh.

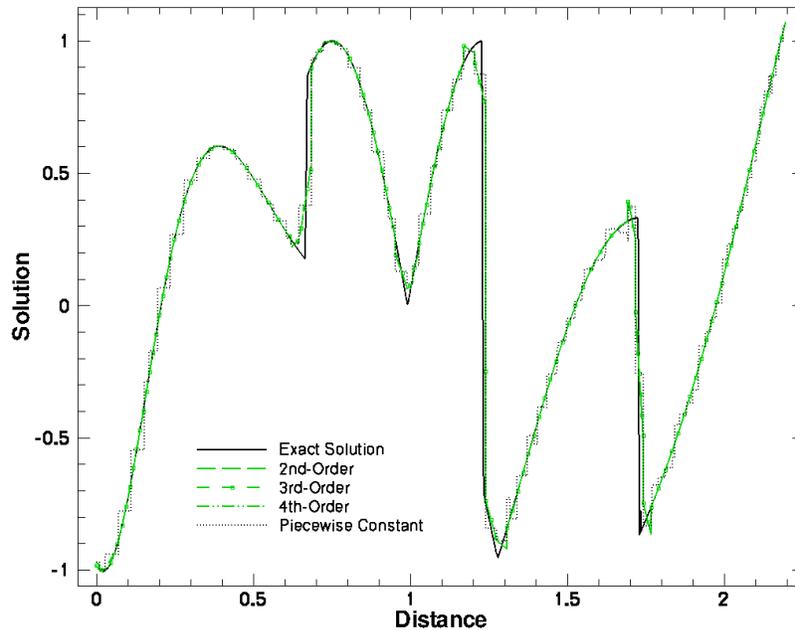


Figure 5.10: *Cross sections of 2nd-order ( $k = 2$ ), 3rd-order ( $k = 3$ ), and 4th-order ( $k = 4$ ) CENO reconstructed solution of the Abgrall function along the line shown in Fig. 5.9(a)–(b).*

accurate solution for this problem. This fact is expected due to the presence of discontinuous solution data in the reconstructed function and due to the design of the CENO scheme. However, a relatively significant reduction in the magnitude of the error for a particular mesh resolution can also be observed (i.e., the error curves are shifted downwards) as the order of the reconstruction is increased. Obviously, this effect is due to a better representation of the function in the smooth regions by higher-order polynomials and translates in fewer computational cells required to obtain a desired accuracy. In particular, the piecewise cubic reconstruction requires about 40% fewer computational cells than the linear one for the arbitrary desired accuracy indicated in the figure, which represents a significant number considering the fact that Abgrall's function exhibits an abundance of discontinuous regions.

### 5.3 Function Reconstruction with AMR

Reconstruction of Abgrall's function is now re-considered in conjunction with AMR. A contour plot of a third-order ( $k = 3$ ) CENO reconstructed solution of this function on an initial uniform  $10 \times 10$  Cartesian mesh consisting of 1 block and 100 computational cells is shown

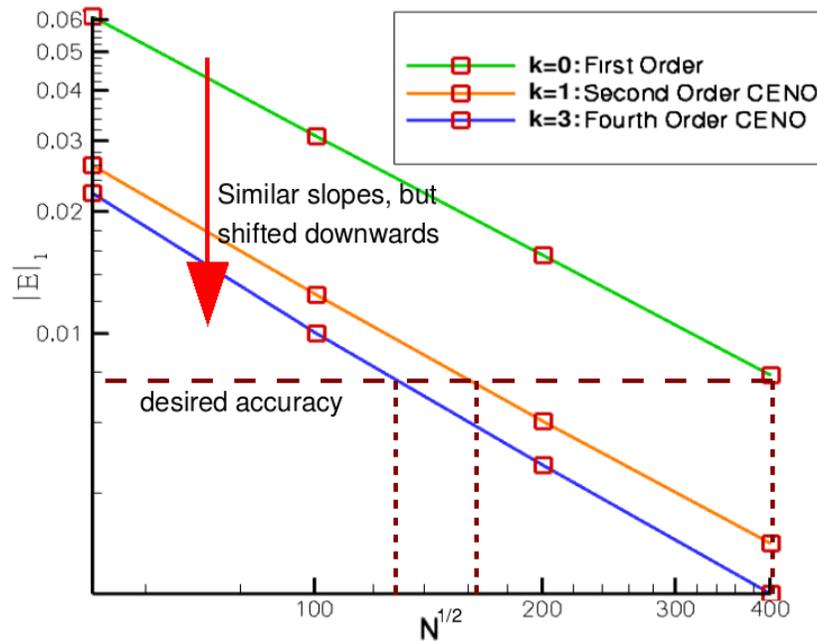
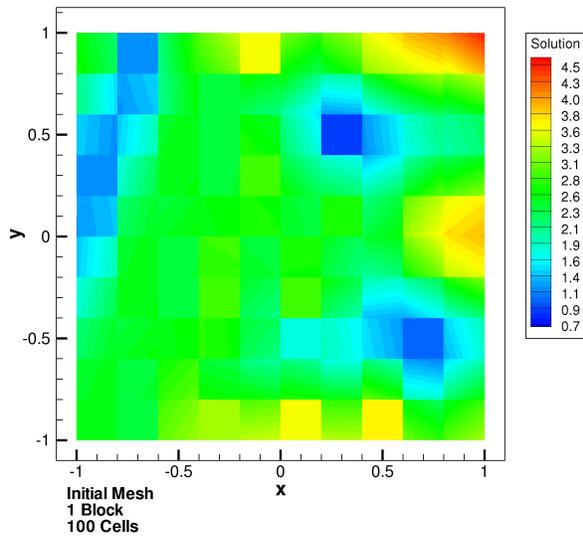
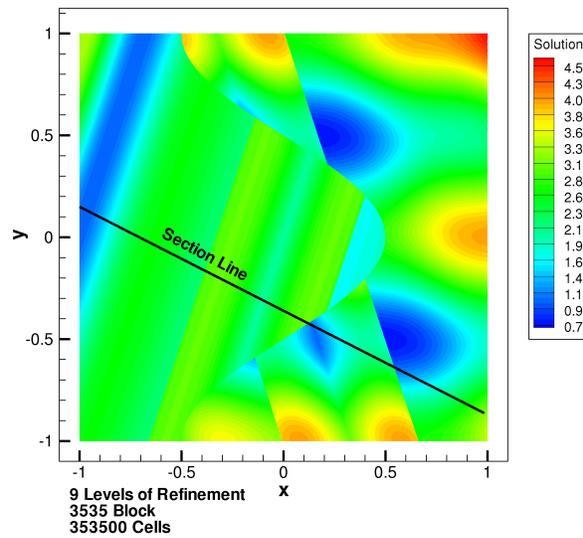


Figure 5.11:  $L_1$  norm of the reconstruction error obtained for Abgrall's function with piecewise constant ( $k=0$ ), linear ( $k=1$ ) and cubic ( $k=3$ ) reconstructions versus the number of computational cells.

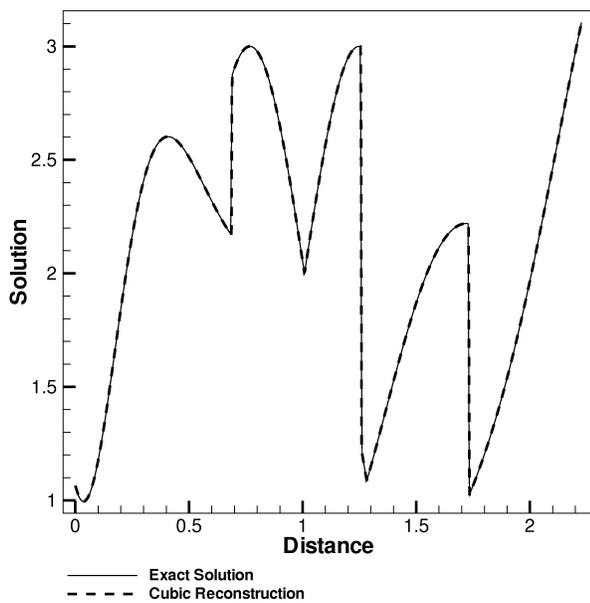
in Fig. 5.12(a). The initial  $10 \times 10$  mesh is given in Fig. 5.13(a). A sequence of refined mesh is then generated using the  $h$ -refinement criterion defined in Sect. 4.4. The Abgrall function is re-imposed on each mesh level before the mesh is further refined. A sequence of eight mesh refinements was carried out and the resulting multi-block AMR meshes are depicted in Fig. 5.13. The final mesh consisted of 3,535 blocks  $10 \times 10$  and 353,500 cells with a refinement efficiency of  $\eta = 0.946$ . The reconstructed solution on the finest mesh using the 4th-order CENO scheme is shown in Fig. 5.12(b). A cross section of the cubic CENO reconstructed solution along the section line shown in Fig. 5.12(b) is depicted in Fig. 5.12(c) and compared to the exact representation of the Abgrall function. In addition, the regions in which the hybrid CENO scheme uses the unlimited cubic and limited linear reconstruction procedures are shown in Fig. 5.12(d). It can be seen that the regions where limited piecewise reconstruction is used correspond to regions where Abgrall's function is either not  $\mathcal{C}^0$  or not  $\mathcal{C}^1$  continuous. The refinement of the AMR mesh is automatically directed toward these regions. Another important observation is that the proposed  $h$ -refinement criterion is capable of identifying all of the discontinuous regions, regardless of the strength of the solution jumps.



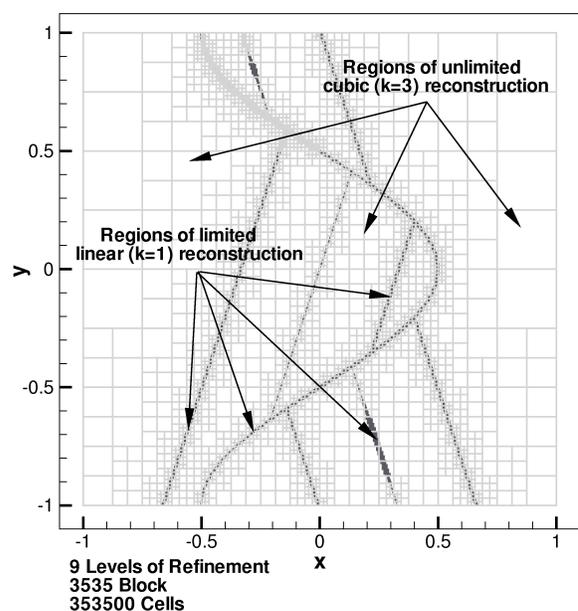
(a) Solution on the initial mesh



(b) Solution on the final mesh



(c)



(d) Distribution of unlimited and limited regions

Figure 5.12: (a) Contour plot of a third-order ( $k = 3$ ) CENO reconstructed solution of the Abgrall's function on an initial uniform  $10 \times 10$  Cartesian mesh consisting of 1 block and 100 computational cells; (b) the cubic reconstructed solution on the mesh obtained after eight mesh refinements which consists of 3,535 blocks and 353,500 cells; (c) Cross section of the cubic CENO reconstructed solution is compared against the exact solution of the Abgrall's function along the section line shown in (b); and (d) regions in which the CENO scheme uses unlimited cubic and limited linear reconstruction.

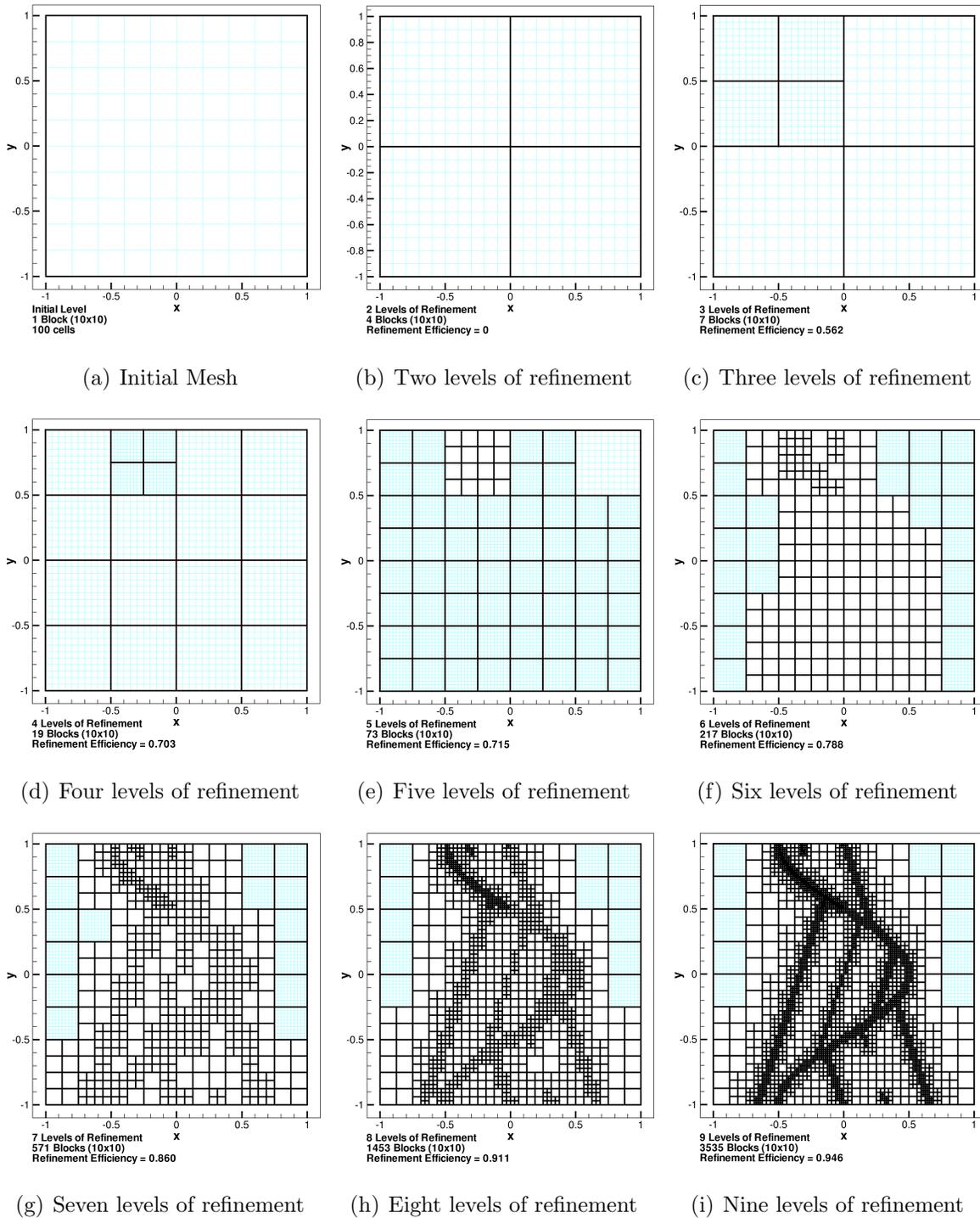


Figure 5.13: (a)-(i) Sequence of AMR meshes generated by the block-based refinement algorithm for the refinement of the Abgrall's function (refer to Fig. 5.12). The initial mesh (a) consisted of 1 block and 100 computational cells. The final mesh (i) had 3,535 blocks  $10 \times 10$  and 353,500 cells with a refinement efficiency of  $\eta = 0.946$ .

## 5.4 Solution of the Advection-Diffusion Equation on Fixed Mesh

The accuracy and robustness of the hybrid CENO algorithm is now assessed based on solutions to the advection-diffusion equation discussed in Sect. 2.1 obtained on fixed meshes.

### 5.4.1 Circular Advection of Smooth Inflow Variation

To begin the assessment of the accuracy of the CENO scheme for the advection-diffusion equation, the problem of circular advection with a constant angular velocity and a smooth inflow solution variation  $u(x, 0) = e^d \sin^6(\pi d)$  if  $d \in [0 : 1]$ , otherwise 0, where  $d = x - 0.3$ , is first considered. The BCs for the problem are shown in Fig. 5.14(a). The predicted solution on a  $80 \times 80$  Cartesian mesh using the 4th-order ( $k = 4$ ) CENO is shown in Fig. 5.14(a). The  $L_1$ ,  $L_2$ , and  $L_\infty$  norms of the error in the predicted solution for both 4th-order versions ( $k = 3$  and  $k = 4$ ) of the proposed CENO finite-volume scheme obtained on a series of grids ranging in size from  $40 \times 40$  to  $200 \times 200$  are given in Fig. 5.14(b) for this smooth problem. As the mesh is refined, the slopes of the  $L_1$ -,  $L_2$ - and  $L_\infty$ -norms for both 4th-order schemes approach in the asymptotic limit -4.53, -4.55 and -4.58 for the cubic reconstruction ( $k = 3$ ), and -4.94 in all error norms for the quartic reconstruction ( $k = 4$ ), respectively, indicating that the expected theoretical accuracy has been achieved in each case.

### 5.4.2 Circular Advection of Discontinuous Inflow Variation

To test the robustness of the CENO scheme in the presence of large gradients or discontinuities, a similar problem but with a top-hat inflow function was used. The results shown in Fig. 5.15(a)–(b) demonstrate that the hybrid CENO algorithm can obtain monotone solutions by switching to a limited linear reconstruction in the regions detected as non-smooth.

### 5.4.3 Numerical Investigation of Elliptic Operator Discretization

Prior to discussing results that concern the advection-diffusion equation with non-zero diffusion operator, it is important to consider some of the properties of the proposed elliptic discretization as defined by Eq. (3.38)–(3.41). Clearly, computing the interface gradients

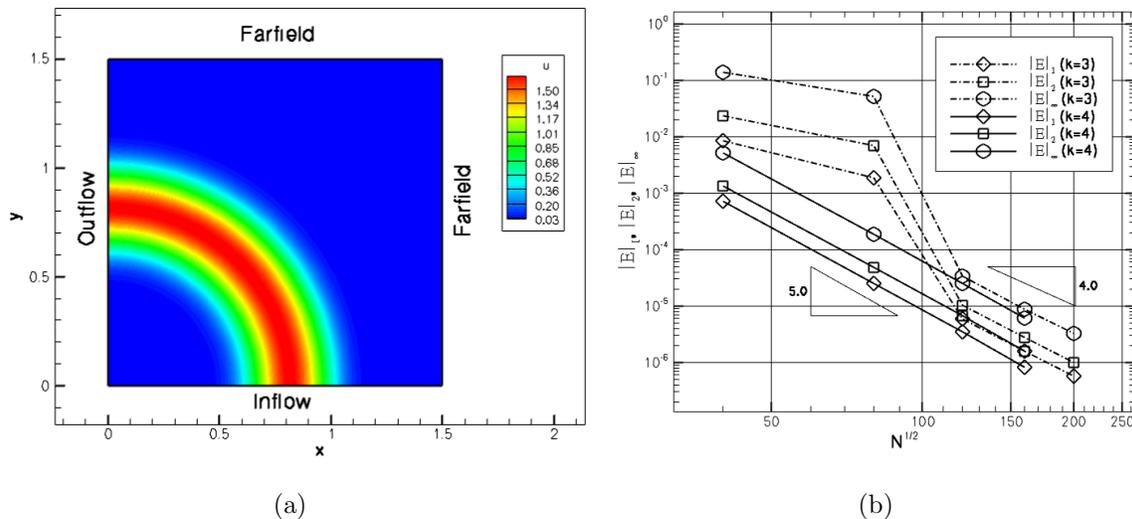


Figure 5.14: (a) Predicted 4th-order ( $k=4$ ) CENO solution to the circular advection of the smooth inflow variation defined in the text obtained on an  $80 \times 80$  Cartesian mesh; (b)  $L_1$ ,  $L_2$  and  $L_\infty$  error norms in the solution for cubic and quartic CENO reconstructions.

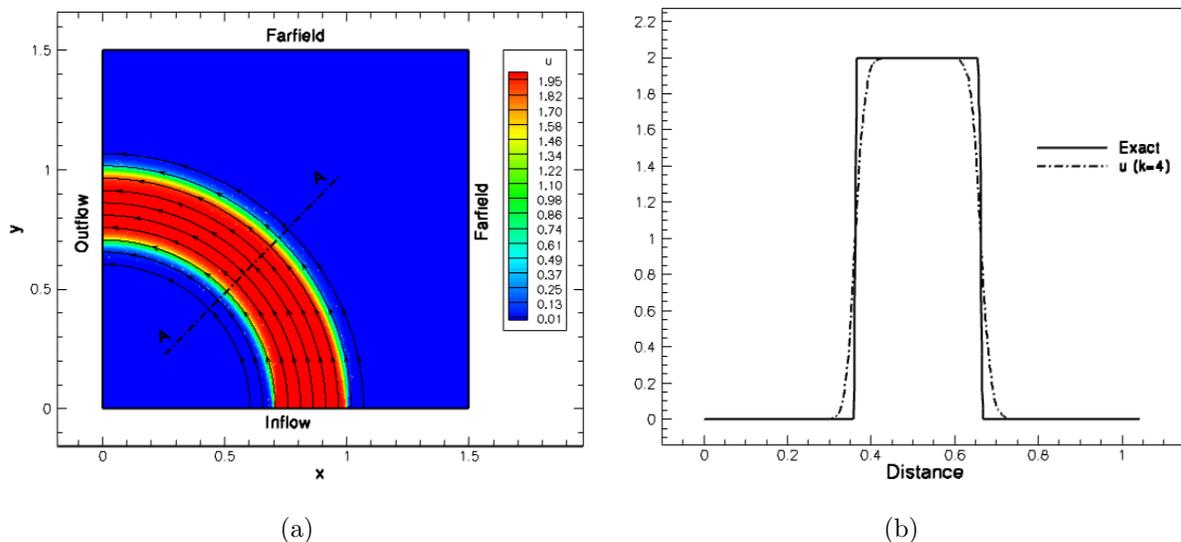


Figure 5.15: (a) Predicted solution for the circular advection of top-hat inflow variation obtained using a 4th-order ( $k=4$ ) CENO scheme on a Cartesian mesh with  $150 \times 150$  computational cells. Several velocity streamlines as well as the cross-section line **A-A** are also shown; (b) Numerical and exact solution profiles along the cross-section line **A-A**.

by averaging the reconstructions used for the discretization of hyperbolic operators has important computational advantages but it also raises questions about its usability, as similar formulations with second-order discretizations lead to several problems such as odd-even

solution decoupling [3, 64]. To shed some light on whether the current elliptic discretization suffers from any of the problems reported in the literature, a series of investigations have been performed for different reconstruction orders. While the accuracy of Eq. (3.38) can be readily observed to be  $k$ -order accurate based on the design of the scheme, other properties of the resulting discretization such as positivity (related to local satisfaction of a discrete maximum principle) or odd-even solution decoupling can be only inferred from more careful analysis.

For these purposes, it is convenient to apply the proposed elliptic discretization to the Laplace operator,  $L(u) = \nabla^2 u$ , and analyze the influence coefficient (i.e., the weight) of each entry in the supporting stencil [64]. Thus, the resulting discrete version of the local Laplacian,  $\tilde{L}(u) = \sum_{n=0}^N \alpha_n \bar{u}_n$ , represents a linear combination of average solutions,  $\bar{u}_n$ , pertaining to cells with influence on the residual computation,  $\mathcal{R}_0$ , for the Laplace equation,  $\tilde{L}(u) = 0$ , in the cell of index  $n = 0$ . Hence, the sufficient condition for the discrete Laplacian based on an  $N$ -point stencil to satisfy a discrete version of the maximum principle, that is

$$\min(\bar{u}_1, \bar{u}_2, \dots, \bar{u}_N) \leq \bar{u}_0 \leq \max(\bar{u}_1, \bar{u}_2, \dots, \bar{u}_N), \quad (5.4)$$

is to require that all coefficients satisfy  $\alpha_n \geq 0$  for  $n \in [1, N]$ . Note that for a given elliptic discretization scheme and a supporting stencil the influence coefficients,  $\alpha_n$ , and consequently the positivity of the scheme, depend only on the local mesh geometry and not on the actual solution. Since these weights represent partial derivatives of the residual  $\mathcal{R}_0$  with respect to average solutions  $\bar{u}_n$ , a convenient and more generic way to compute them for arbitrary mesh geometries is by a finite-differencing approximation. As proposed by Coirier [64], the positivity and stability of the scheme can then be characterized in terms of  $\alpha_0$  and  $\tilde{\alpha}_{\min}$  where

$$\tilde{\alpha}_{\min} = \frac{\min(\alpha_n, 0)}{\sqrt{\frac{1}{N} \sum_n \alpha_n^2}}, \quad n \in [1, N]. \quad (5.5)$$

Ideally,  $\alpha_0 < 0$  for stability and  $\tilde{\alpha}_{\min} = 0$  for positivity [65]. Moreover, it is important to assess whether or not the dangerous situation of odd-even solution decoupling may occur as a manifestation of checkerboard patterns of coefficients  $\alpha_n = 0$  (i.e., the  $n$ -neighbour cell does not influence the Laplacian residual  $\mathcal{R}_0$ ).

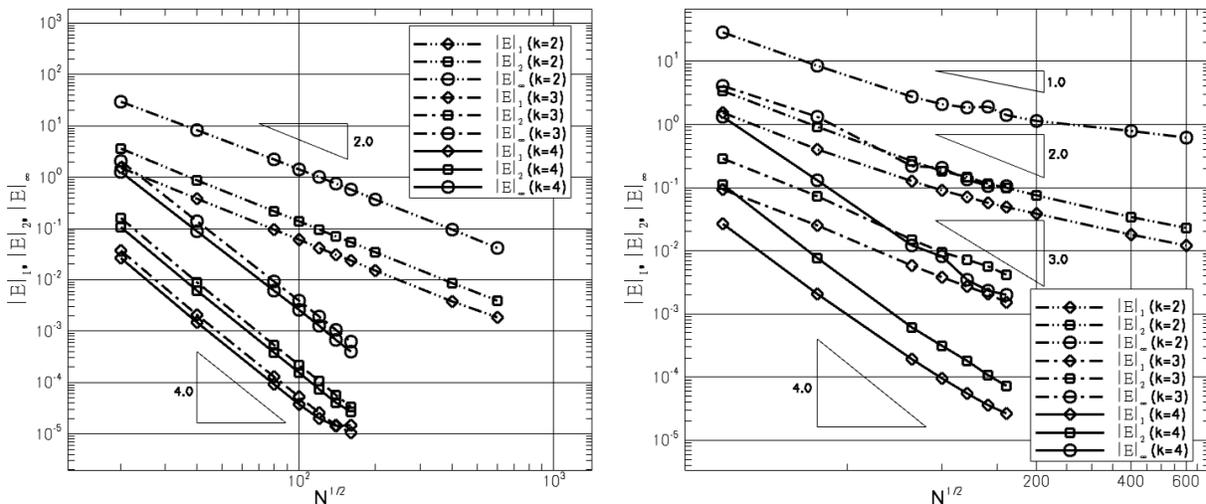
In the current work, different mesh topologies were analyzed including Cartesian, stretched,

and randomly disturbed quadrilateral grids. Analysis of the proposed discretization procedure has shown that odd-even solution decoupling does not occur. In terms of the stability and positivity, it was found that  $\alpha_0 < 0$  (i.e., the schemes are stable) but also  $\tilde{\alpha}_{\min} < 0$  for discretizations of all order, unfortunately implying that, while stable, none of the discretizations satisfy the discrete maximum principle. This result agrees with the general perception that, for finite-volume discretizations, accuracy and positivity are essentially conflicting properties [64]. Note that for square Cartesian meshes, values for  $\tilde{\alpha}_{\min}$  are found to be -0.823 for  $k=2$ , -0.362 for  $k=3$  and -0.854 for  $k=4$  when inverse distance geometric weighting is used in the  $k$ -exact reconstruction. However, the positivity can be improved by using an inverse distance squared geometric weighting, for which  $\tilde{\alpha}_{\min}$  was found to be -0.051 for  $k=2$ , -0.247 for  $k=3$  and -0.324 for  $k=4$ . For non-Cartesian meshes, large variations in the value of  $\tilde{\alpha}_{\min}$  are possible ( $-5 < \tilde{\alpha}_{\min} < 0$ ), depending on the regularity and topology of the mesh. It should be mentioned that for the test problems and the computational meshes discussed in this results section, the lack of strict positivity of the elliptic discretization did not seem to represent an obvious issue, at least for the range of cases considered.

#### 5.4.4 Residual-Based Accuracy Assessment for Poisson Equation

To investigate the accuracy of the proposed algorithm for discretization of elliptic operators, convergence studies based on the residual error to Poisson equation  $\vec{\nabla} \cdot (\vec{\nabla} u) = a e^{\beta u}$  have been performed using the exact solution  $u(x, y) = \frac{1}{\beta} \left[ \ln \left( \frac{8C}{a\beta} \right) - 2 \ln |(x+A)^2 + (y+B)^2 - C| \right]$ , where  $A = 2.0$ ,  $B = 1.0$ ,  $C = 2.0$ ,  $a = 2.5$  and  $\beta = 0.001$ . The domain in this study was the rectangular box defined by  $0.5 \leq x, y \leq 4.5$ . The  $L_1$ ,  $L_2$  and  $L_\infty$  norms of the flux integral (i.e. the residual) errors obtained for a sequence of Cartesian and randomly disturbed meshes are shown in Fig. 5.16(a) and (b), respectively, for quadratic ( $k=2$ ), cubic ( $k=3$ ) and quartic ( $k=4$ ) interpolants. The correspondent slopes of the  $L_1$ -,  $L_2$ - and  $L_\infty$ -norms were determined based on all measured error values and are shown in Table 5.1 for both Cartesian and disturbed meshes.

The results of Fig. 5.16 and Table 5.1 show that both cubic and quartic interpolants produce a 4th-order scheme for Cartesian meshes in all error norms, whereas the quadratic reconstruction generates a 2nd-order one. The result for  $k=3$  in Fig. 5.16(a) demonstrates that due to error cancellations a cubic interpolant is also able to generate a 4th-order scheme for elliptic operator discretizations. However, an analysis of the error norms in Fig. 5.16(b)



(a) Error norms for Cartesian meshes

(b) Error norms for distorted meshes

Figure 5.16:  $L_1$ ,  $L_2$  and  $L_\infty$  norms of the residual error to Poisson equation  $\vec{\nabla} \cdot (\vec{\nabla} u) = a e^{\beta u}$  for piecewise quadratic ( $k=2$ ), cubic ( $k=3$ ) and quartic ( $k=4$ ) reconstructions versus the number of computational cells of (a) Cartesian meshes and (b) randomly distorted meshes.

Table 5.1: The slope values corresponding to each convergence error plot shown in Fig. 5.16.

Mesh	Reconstruction Order	$L_1$	$L_2$	$L_\infty$
Cartesian Grid	$k=2$	1.99	2.01	1.93
	$k=3$	3.98	4.07	3.89
	$k=4$	3.77	4.02	3.87
Distorted Grid	$k=2$	1.41	1.47	1.10
	$k=3$	1.99	2.05	1.86
	$k=4$	3.35	3.53	3.18

reveals that the error cancellation effect does not occur on the randomly disturbed meshes and as such, the error norms of cubic and quartic reconstructions differ by one. Unfortunately, this is not a rigorous convergence study due to the random disturbance of the nodes and as a result, the error norms of both reconstructions are somewhat lower than the expected theoretical ones. However, the numerical results provide support for the point made concerning the effects of error cancellation.

### 5.4.5 Heat Conduction Equation with Source Term

The predictive capability of the proposed algorithm was also determined for solutions to the Poisson equation  $\nabla^2 u = 6u$ , using the boundary conditions and the domain shown in Fig. 5.17(a) by finding the converged steady-state solution to the equation  $\frac{\partial u}{\partial t} = \nabla^2 u - 6u$ . Although a general exact solution exists for the Poisson equation shown above, the boundary conditions imposed for this problem makes the solution independent of the  $y$ -direction and consequently, an analytic solution can be obtained by solving a second-order ODE with constant coefficients. Hence, the analytic solution for this particular problem is:

$$u(x, y) = \frac{\left( e^{\frac{\sqrt{6}}{2}(1+2x)} - e^{-\frac{\sqrt{6}}{2}(1+2x)} \right) e^{\sqrt{6}}}{e^{2\sqrt{6}} - 1}. \quad (5.6)$$

Figure 5.17(b) shows the 4th-order solution ( $k = 4$ ) obtained with the high-order algorithm and with Dirichlet boundary conditions implemented based on the exact solution on a disturbed computational grid with 100 cells. The result of the grid convergence studies carried out for this problem and presented in Fig. 5.18 demonstrates that the numerical solution converges with the theoretical order of accuracy in all  $L_1$ ,  $L_2$  and  $L_\infty$  norms for quadratic ( $k = 2$ ) and quartic ( $k = 4$ ) reconstructions. For linear ( $k = 1$ ) and cubic ( $k = 3$ ) approximations the  $L_1$ ,  $L_2$  and  $L_\infty$  norms of the solution error converge with a rate that is one order higher, due to error cancellations on uniform Cartesian meshes. Thus, for linear ( $k = 1$ ), quadratic ( $k = 2$ ) and cubic ( $k = 3$ ) reconstructions, the  $L_1$  norms of the solution error are 2.14, 2.21 and 4.16 respectively, and for quartic reconstruction ( $k = 4$ )  $L_1 = 4.37$ ,  $L_2 = 4.40$  and  $L_\infty = 4.25$ . For comparison, the error norms obtained for the 4th-order scheme ( $k = 4$ ) on distorted meshes are also shown in Fig. 5.18 and confirm that the theoretical order of accuracy is again obtained.

### 5.4.6 Solution of the Laplace Equation on Curved Boundaries

The numerical scheme was also investigated for solutions to the Laplace equation and geometries with curved boundaries such as the annulus domain shown in Fig. 5.19(a). Dirichlet boundary conditions were implemented along the domain contour based on the exact solution considered for this problem which was  $u(x, y) = e^{\mu x} (A \cos(\mu y) + B \sin(\mu y))$ , where  $A = 1$ ,  $B = 2$  and  $\mu = 1.5$ . The predicted solution obtained using the 4th-order ( $k = 4$ ) high-order

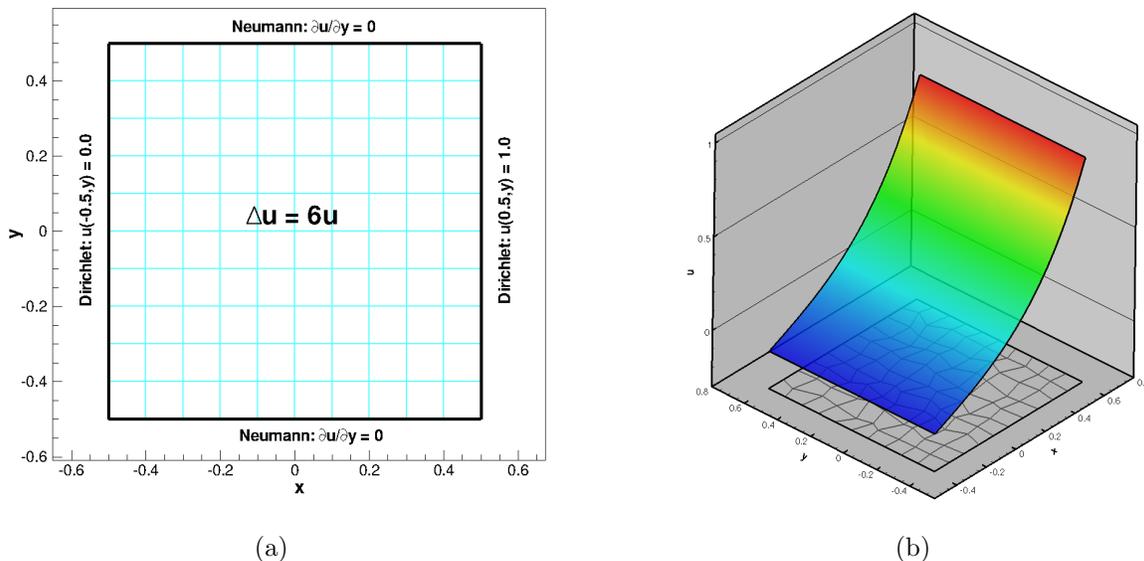


Figure 5.17: (a) Definition of the diffusion problem and (b) 4th-order solution to diffusion problem obtained on distorted computational grid.

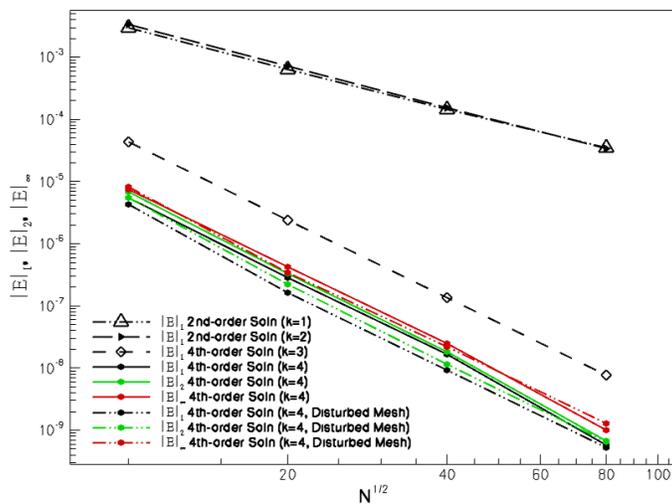


Figure 5.18:  $L_1$ ,  $L_2$  and  $L_\infty$  error norms for the heat conduction problem as a function of mesh density obtained using the high-order scheme with linear ( $k=1$ ), quadratic ( $k=2$ ), cubic ( $k=3$ ), and quartic ( $k=4$ ) interpolants for Cartesian meshes. Error norms for quartic polynomial are also shown for disturbed meshes.

scheme on a curvilinear mesh with  $40 \times 40$  cells is shown in Fig. 5.19(a). The  $L_1$ ,  $L_2$ , and  $L_\infty$  norms of the error in the predicted solution for cubic ( $k=3$ ) and quartic ( $k=4$ ) interpolants are given in Fig. 5.19(b) for this problem. The slopes of the  $L_1$ - and  $L_2$ -norms reach in the

asymptotic limit -3.86 and -3.85 for  $k=3$  and -3.86 and -3.81 for  $k=4$ , respectively. As can be easily observed, even if the slope of the cubic and quartic polynomials are basically the same and both very close to the theoretical value, there is about one order difference between the magnitudes of the errors, demonstrating the benefits of using quartic interpolants.

### 5.4.7 Two-Dimensional Channel Flow Problem

The application of the proposed CENO scheme is now considered for problems involving both advection and diffusion terms for a range of Péclet numbers. Solution of the advection-diffusion equation Eq. (2.1) with a constant velocity,  $\vec{V} = (v_0, 0)$ , and constant diffusion coefficient,  $\kappa(x, y) = \kappa_0 = 0.01$ , on the rectangular domain of length  $L=3$  and unit width was considered for three different Péclet numbers, depending on the value of  $v_0$ . The studied Péclet numbers correspond to  $Pe=0.1$ , at which diffusion dominates the flow, to  $Pe=1$ , at which advection and diffusion are equally represented and to  $Pe=10$ , which is representative for an advection dominated problem.

The boundary conditions for this problem, as shown in Fig. 5.20, are

$$u(x, 0) = u(x, 1) = 0, \quad u(0, y) = \sin(\pi y), \quad \text{and} \quad \frac{\partial u(L, y)}{\partial x} = 0. \quad (5.7)$$

A similar study was considered by Ollivier-Gooch and Van Altena for the evaluation of high-

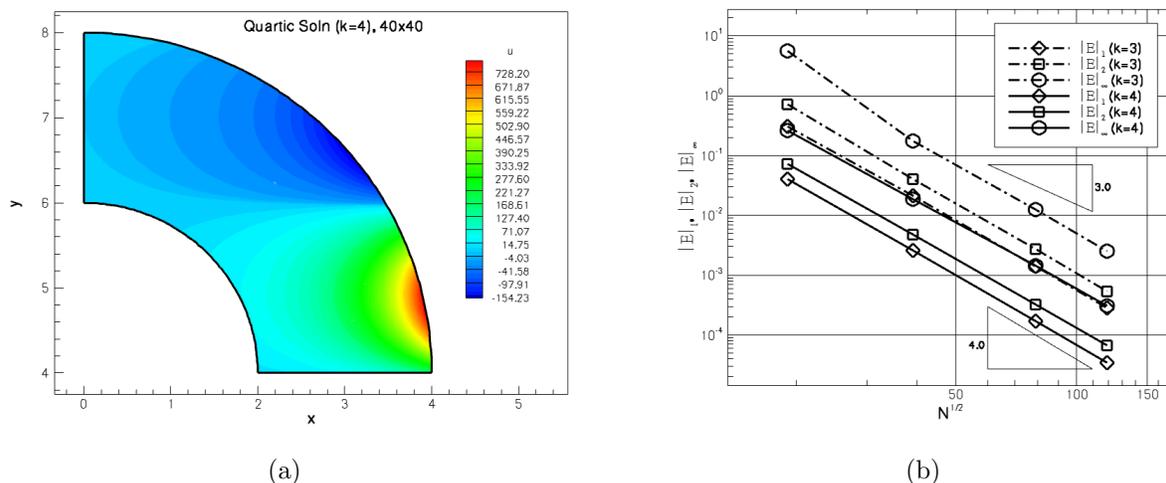


Figure 5.19: (a) Fourth-order ( $k=4$ ) solution to the Laplace equation on a mesh with  $40 \times 40$  computational cells. (b)  $L_1$ ,  $L_2$  and  $L_\infty$  norms of the error as a function of mesh density obtained with the cubic ( $k=3$ ) and quartic ( $k=4$ ) polynomial interpolants.

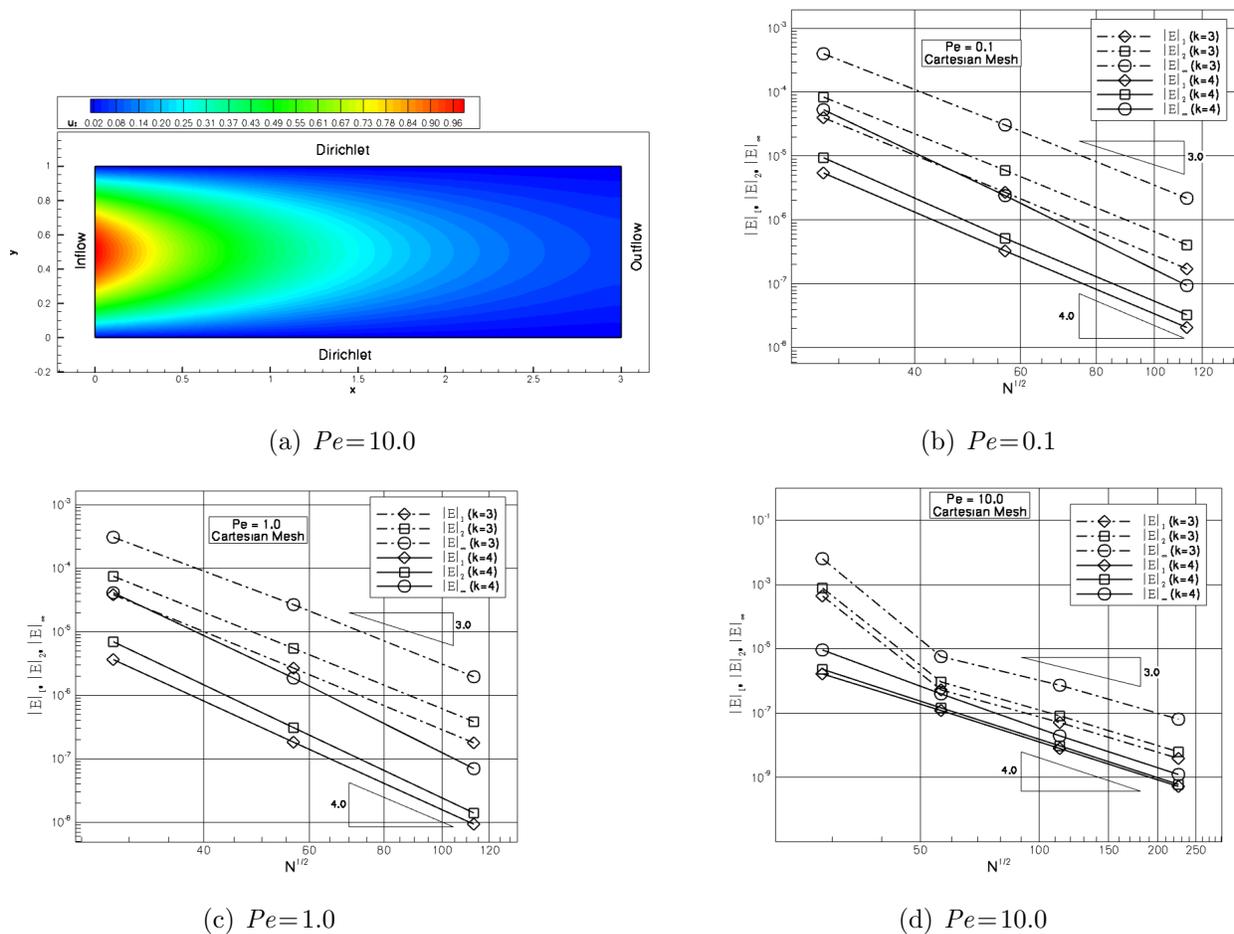


Figure 5.20: (a) Numerical prediction of channel flow problem on an  $80 \times 40$  Cartesian mesh. (b)-(d)  $L_1, L_2$  and  $L_\infty$  norms of the solution error as a function of mesh density obtained with cubic ( $k=3$ ) and quartic ( $k=4$ ) interpolants for different Péclet numbers.

order finite-volume schemes in [72]. The analytic solution to this problem can be determined by the method of separation of variables and can be arranged such that to avoid numerical problems for convection dominated flows in the following form:

$$u(x, y) = \frac{\sin(\pi y)}{(R [e^{L(R-1)}]^{r_1} - 1)} \left( R [e^{(RL+x-L)}]^{r_1} - e^{r_2 x} \right) \quad (5.8)$$

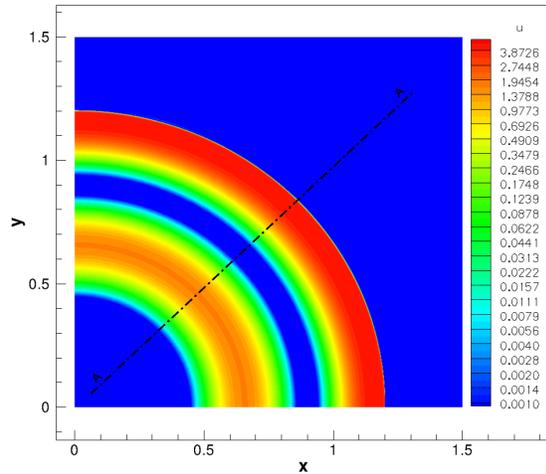
where  $r_{1,2} = \frac{v_0}{2\kappa_0} \pm \sqrt{\frac{v_0^2}{4\kappa_0^2} + \pi^2}$  and  $R = \frac{r_2}{r_1}$ .

The numerical solution obtained for  $Pe = 10$  on an  $80 \times 40$  Cartesian mesh is shown in Fig. 5.20(a) and the error norms for this advection-diffusion problem associated to each

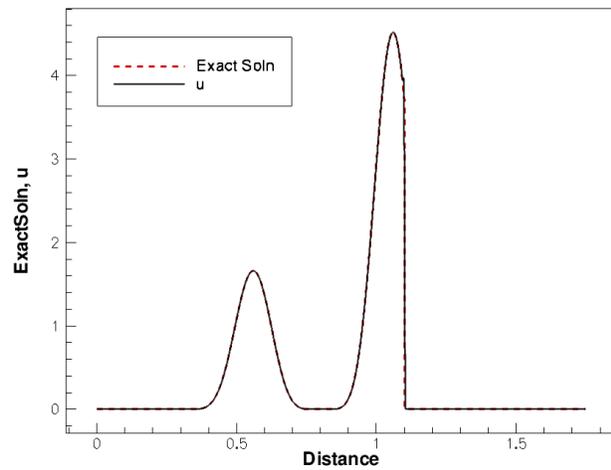
of the three Péclet numbers are shown in Fig. 5.20(b)–(d). The results show that the errors generated by the quartic polynomial are consistently lower than those of the cubic interpolant by at least one order of magnitude for all Péclet numbers and get the theoretical accuracy in all norms. Thus, the  $L_1$ - and  $L_2$ -norms for  $k=4$  are -4.02 and -4.08 for  $Pe=0.1$ , -4.30 and -4.46 for  $Pe=1.0$  and -3.92 and -3.95 for  $Pe=10.0$ , respectively. In the case of cubic interpolant, the error norms are -3.92 and -3.85 for  $Pe=0.1$ , -3.88 and -3.81 for  $Pe=1.0$  and -3.53 and -3.62 for  $Pe=10.0$ , respectively. It can be also seen in the error plots that, for the same accuracy level, the cubic interpolant requires almost twice as many computational cells as the quartic one. Taking into account that both polynomial reconstructions, quartic and cubic, use the same reconstruction stencil, the only extra cost associated with quartic reconstruction is to determine five additional variables during the least-squares reconstruction procedure.

## 5.5 AMR Solution of the Advection-Diffusion Equation

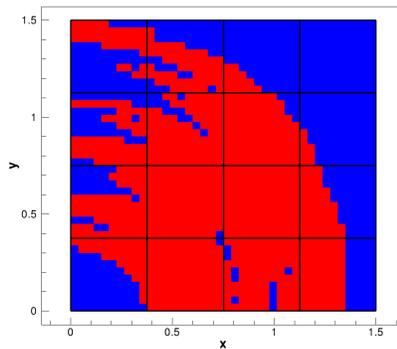
The application of the proposed high-order CENO finite-volume scheme to the solution of circular advection at constant angular velocity is again considered. In this case, the inflow function was chosen such that to test both the accuracy and robustness of the scheme as well as the capability of the high-order scheme in conjunction with AMR. The inflow variation was  $u(x, 0) = e^{2d} \sin^6(2\pi d)$  if  $d \in [0 : 0.8]$ , otherwise 0, where  $d = x - 0.4$ . The same boundary conditions as those shown in Fig. 5.14 were used for this problem. The predicted solution obtained using the 4th-order CENO scheme ( $k=3$ ) on a final mesh consisting of 2,911 blocks and 291,100 computational cells is shown in Fig. 5.21(a). The solution profile along the cross-section A-A is compared against the exact solution in Fig. 5.21(b). The initial mesh and the final mesh are depicted in Fig. 5.21(c) and (e), respectively. The results clearly show that the proposed AMR scheme in conjunction with the  $h$ -refinement criteria based on the smoothness indicator of the hybrid CENO reconstruction technique is capable of refining both under-resolved (in-accurate) and non-smooth regions of the solution and will not unnecessarily refine resolved solution content. The smooth peaks are all well captured by the high-order scheme whereas the solution discontinuity is well identified by the smoothness indicator and well resolved by the hybrid CENO scheme in conjunction with the AMR procedure.



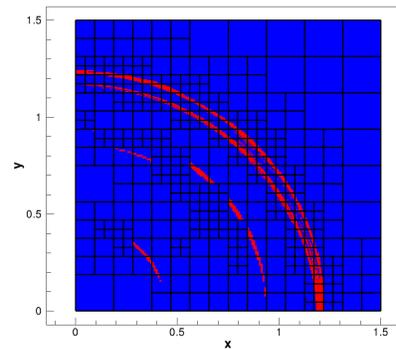
(a) Fourth-order CENO prediction



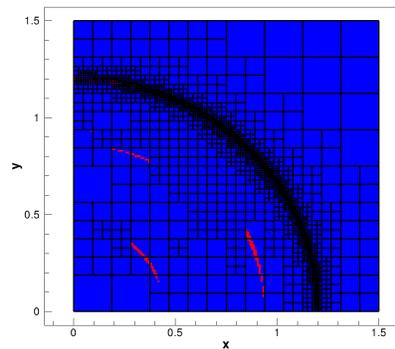
(b)



(c) Initial mesh



(d) Three levels of refinement



(e) Six levels of refinement

Figure 5.21: (a) Fourth-order ( $k = 3$ ) CENO prediction to circular advection of a smooth and discontinuous inflow variation on final mesh; (b) Comparison of numerical and exact solutions along the cross section A-A; (c) Initial mesh with 16  $10 \times 10$  blocks and corresponding regions in which CENO scheme uses limited linear reconstruction (shown in red); (d) Refined mesh after 3 level of refinement (d) Final refined mesh after 6 refinement levels with 2,911  $10 \times 10$  blocks and refinement efficiency of  $\eta = 0.955$ .

## 5.6 Solution of the Euler Equations on Fixed Mesh

The application of the proposed high-order CENO finite-volume scheme to the numerical solution of a hyperbolic system of nonlinear conservation laws has also been investigated. Here, numerical solutions of the one- and two-dimensional forms of the Euler equations of compressible gas dynamics described in Sect. 2.2 are considered.

### 5.6.1 One-Dimensional Blastwave Interaction Problem

The accuracy and robustness of the CENO scheme is investigated first with the classical blastwave interaction problem of Woodward and Colella [166]. Numerical results for this case are shown in Fig. 5.22 for both the 4th-order CENO scheme with cubic ( $k=3$ ) hybrid reconstruction and a 4th-order version of the original ENO scheme using locally defined characteristic variables. Results for a 2nd-order TVD scheme with limited piecewise linear reconstruction are also depicted in the figure. A uniform mesh containing 400 cells was used.

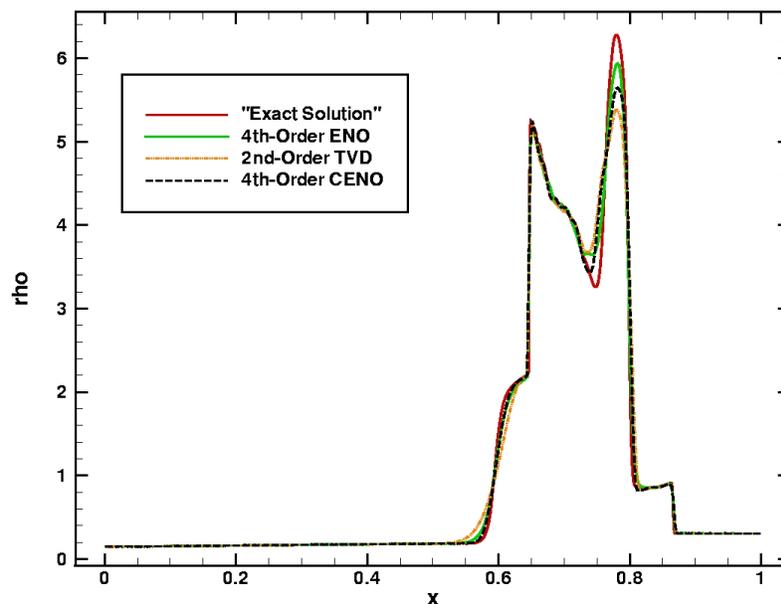


Figure 5.22: Numerical solutions of the one-dimensional blast wave interaction problem of Woodward and Colella [166]. Predictions of flow density at  $t = 0.038s$  obtained using the 4th-order CENO finite-volume scheme are depicted and compared to solutions obtained using a 4th-order ENO scheme and a 2nd-order TVD scheme with limited linear piecewise linear reconstruction for a uniform mesh with 400 cells.

The “exact solution” was generated with the 4th-order ENO algorithm on a mesh with 800 cells. It can be seen that the predictions of the CENO scheme are very similar to those of the ENO method, and both yield significantly improved results as compared to the TVD scheme for this very challenging problem involving strong nonlinear wave interactions.

### 5.6.2 Sod Shock-Tube Problem

Another classical test case considered herein is the Sod’s shock tube problem [167] on the domain  $[-1, 1]$  with the following initial condition

$$[\rho, u, p](x, 0) = \begin{cases} [1 & 0 & 1], & x < 0 \\ [\frac{1}{8} & 0 & \frac{1}{10}], & x > 0. \end{cases}$$

The predicted density profile by the 4th-order CENO scheme with 100 and 400 cells, respectively, at  $t = 0.5$ s is shown in Fig. 5.23. In this numerical simulation the 4th-order Runge-Kutta time-marching method and a  $CFL = 0.5$  were used together with a CENO spatial discretization based on the Roe numerical flux function, the Venkatakrisnan’s limiter

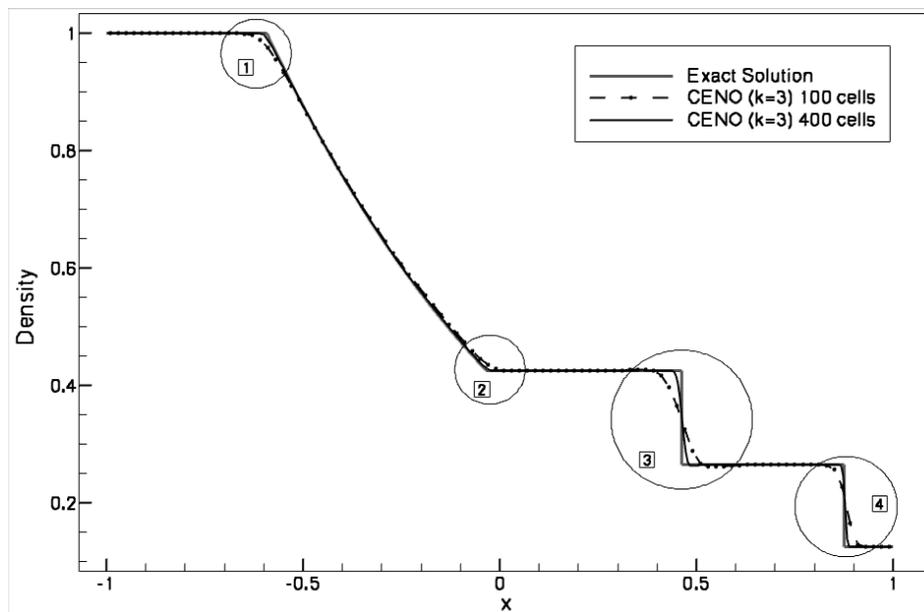


Figure 5.23: *The numerical prediction of the 4th-order CENO scheme with 100 and 400 cells is compared to the exact solution of the density variable for the Sod’s shock tube problem. Several regions of interest are identified for reference: (1) - rarefaction head, (2) - rarefaction tail, (3) - contact discontinuity, (4) - shockwave discontinuity.*

and the CENO cutoff value of  $S_c = 1, 500$ . The results show that the CENO scheme is capable of representing accurately the smooth rarefaction profile while providing a monotone and accurate representation of the solution discontinuities on different mesh resolutions.

To assess the predictive capabilities of the CENO scheme relative to other approaches, the CENO profiles in the regions defined in Fig. 5.23 are compared in Fig. 5.24 against those of a TVD scheme and of an ENO method using characteristic variables. The results demonstrate that CENO predictions are very similar to those of ENO and consistently more accurate than those of the TVD scheme. Moreover, slightly better results are obtained with CENO than with the ENO method near the rarefaction head and the contact discontinuity.

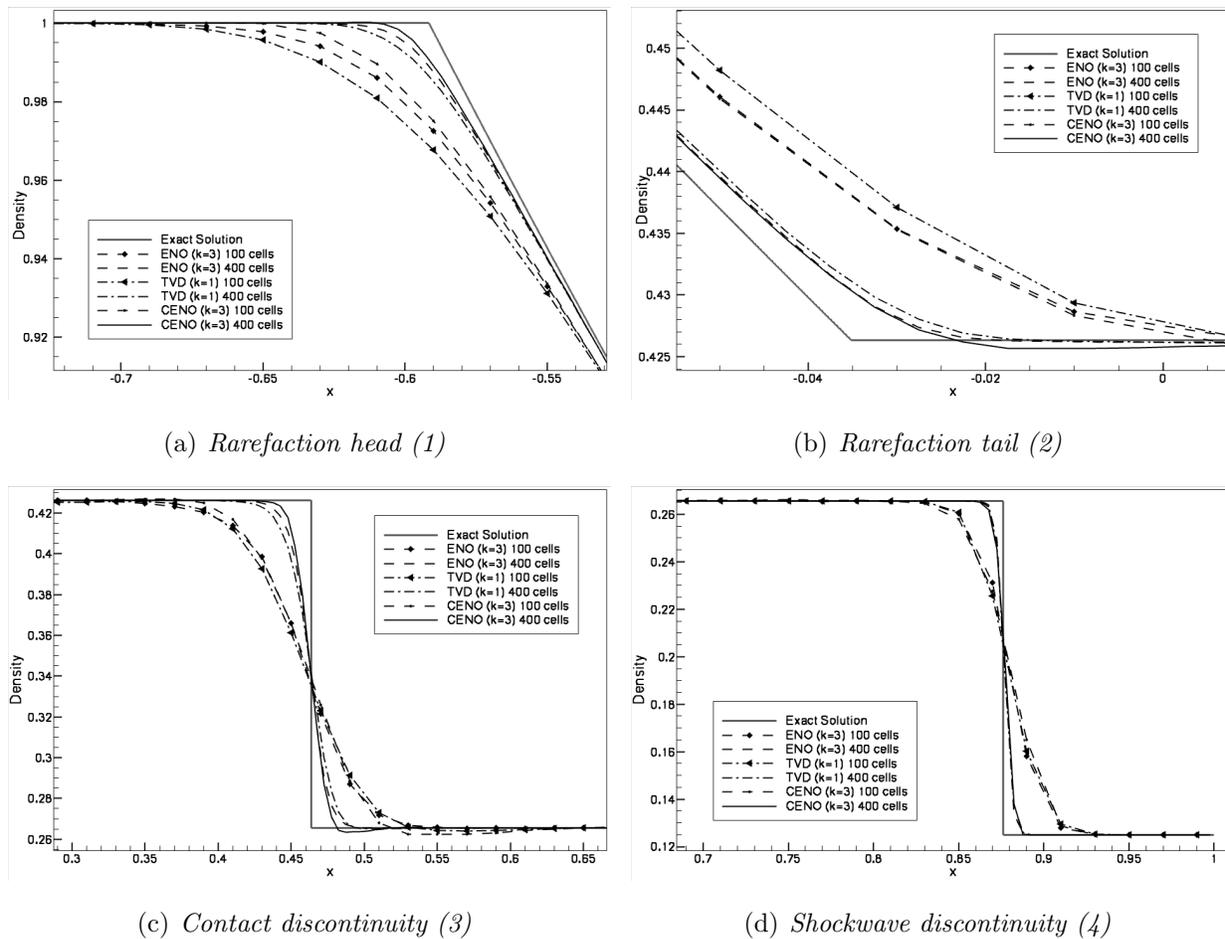


Figure 5.24: Predicted density profiles in the regions defined in Fig. 5.23 obtained with a 4th-order ENO, a 4th-order CENO and a 2nd-order TVD scheme on different mesh resolutions.

### 5.6.3 Ringleb Flow

The application of the proposed high-order CENO finite-volume scheme to the solution of the two-dimensional form of the Euler equations using body-fitted multi-block quadrilateral mesh has also been considered herein. The accuracy of the high-order spatial discretization scheme is demonstrated by first considering numerical predictions of Ringleb's flow. Ringleb's flow involves isentropic, expanding, irrotational flow between two streamlines and exact solutions for this smooth continuous flow field can be determined by analytical means [164, 168]. Therefore, this flow also represents a good test case to assess the influence of different sets of reconstructed variables to the accuracy of the CENO scheme. In particular, the head-to-head comparison of CENO schemes based on reconstruction of primitive and conserved variables, respectively, has been performed herein by carrying out systematic grid convergence studies and comparing the solution accuracies in the density variable for each approach.

As a first case, an entirely subsonic flow solution is considered for which the domain is defined by streamlines corresponding to  $k_{min} = 0.4$  and  $k_{max} = 1.0$  and by the subsonic inflow boundary corresponding to  $q = 0.3$ . In this case, inflow, outflow, and streamline

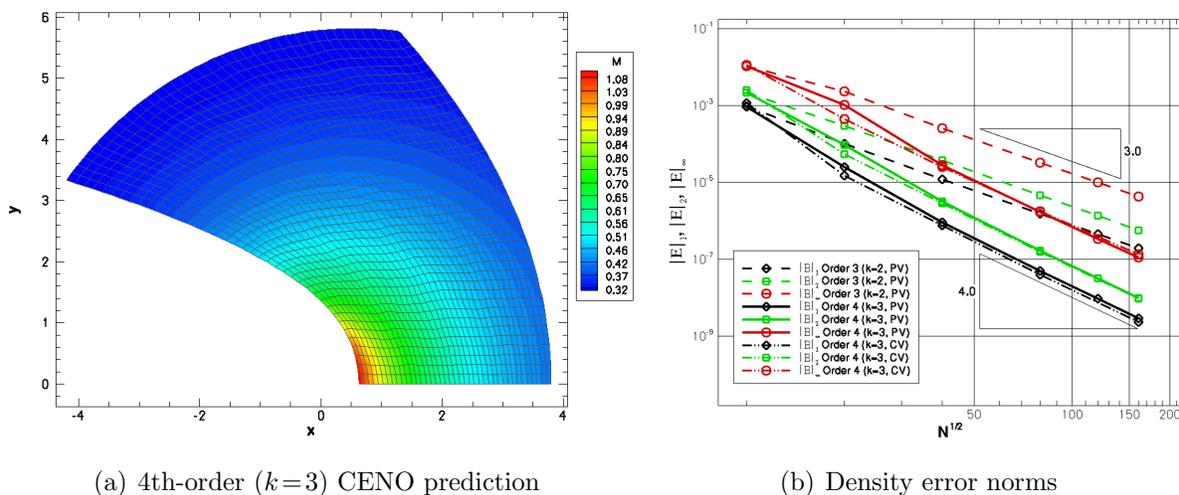


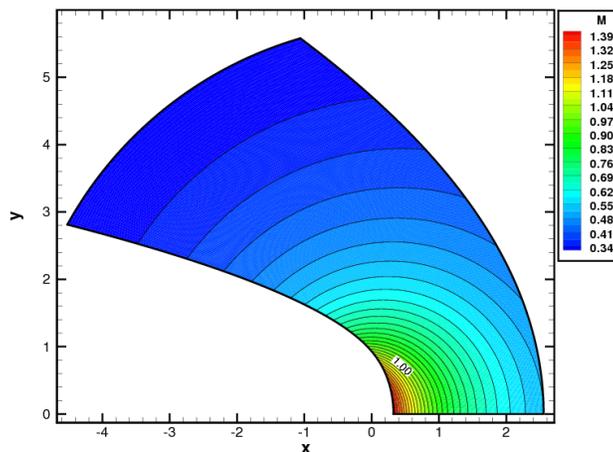
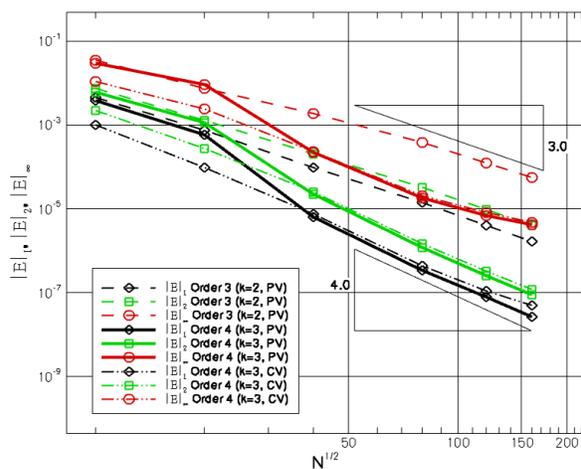
Figure 5.25: (a) Predicted Mach number distribution for subsonic Ringleb's flow obtained using the 4th-order CENO scheme; and (b)  $L_1$ ,  $L_2$ , and  $L_\infty$  error norms in the predicted solution density for this flow obtained using the 3rd- and 4th-order versions of the CENO schemes as a function of mesh density. The error norms for the 4th-order CENO scheme are shown for both reconstruction of primitive variables (PV) and of conserved variables (CV).

boundary conditions based on the exact solution are prescribed. The predicted Mach number distribution for subsonic Ringleb's flow obtained using the 4th-order CENO FV scheme on a  $40 \times 40$  body-fitted mesh is shown in Fig. 5.25(a). The  $L_1$ ,  $L_2$ , and  $L_\infty$  norms of the error in the predicted solution density for 3rd- and 4th-order versions ( $k = 2$  and  $k = 3$ ) of the CENO scheme with reconstruction of primitive variable (PV) obtained on a series of grids ranging in size from  $10 \times 10$  to  $160 \times 160$  is given in Fig. 5.25(b) for this subsonic flow. As the mesh is refined, the slopes of the  $L_1$ - and  $L_2$ -norms for the 3rd- and 4th-order schemes approach -2.97, -3.02, -4.08, and -4.09, respectively, indicating that the expected theoretical accuracy has been achieved in each case. Moreover, the same figure shows that the error norms corresponding to the CENO variant based on reconstruction of conserved variables (CV) are very comparable to those resulting from the CENO reconstruction of primitive variables, the former being characterized by a slightly lower magnitude than the latter but basically by the same order of accuracy.

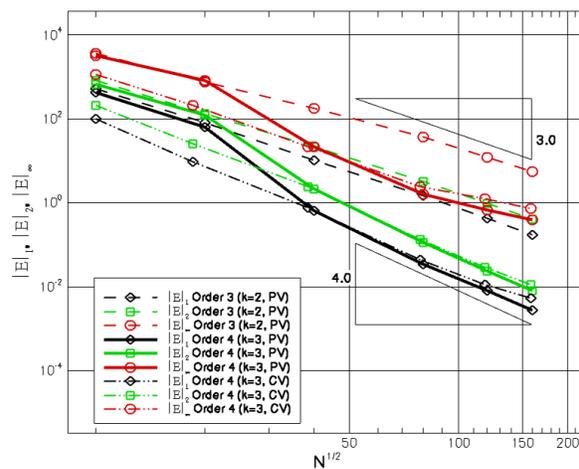
A transonic variant of Ringleb's flow defined by  $k_{min} = 0.5$ ,  $k_{max} = 1.2$ , and  $q = 0.3$  has also been examined here. In this case, reflection boundary conditions were applied along the streamline boundaries by enforcing the inviscid (slip) condition,  $\vec{V} \cdot \vec{n} = 0$ , at all Gauss integration points via the constrained least-squares reconstruction procedure outline in Sect. 3.6.3. The predicted Mach number distribution for this flow obtained using the 4th-order CENO finite-volume scheme on a  $80 \times 80$  body-fitted mesh is given in Fig. 5.26(a) and the  $L_1$ ,  $L_2$ , and  $L_\infty$  norms of the error in predicted density and pressure obtained using the 3rd- and 4th-order versions of the CENO scheme are shown in Fig. 5.26(b) and (c).

In this case, the slopes of the  $L_1$ - and  $L_2$ -norms shown in Fig. 5.26(b) for the 3rd- and 4th-order CENO schemes based on reconstruction of primitive variables approach -3.06, -3.00, -3.93 and -4.02, respectively, illustrating that the accuracy of the scheme can be maintained at curved boundaries by using constrained least-squares reconstruction and accurate boundary description. Similarly to the subsonic case, the error norms of the 4th-order CENO schemes based on reconstruction of primitive and conserved variables, respectively, are compared for this supersonic flow in Fig. 5.26(b). The results show that the convergence rates for the two reconstruction methods based on primitive and conserved variables are again very similar and both recover the expected convergence rate in the asymptotic regime. Additionally, it is easily observed by inspecting Fig. 5.26(b) and (c) that the error norms in predicted solution pressure are merely a translation of those obtained in predicted solution density, thereby

confirming that the same order of accuracy is recovered for all primitive solution variables. Moreover, our numerical experiments demonstrate that the error introduced by the mapping of average conserved to average primitive variables as explained in Sect. 3.6 is relatively small and does not affect noticeably the order of convergence for the scheme.

(a) 4th-order ( $k=3$ ) CENO prediction

(b) Density error norms



(c) Pressure error norms

Figure 5.26: (a) Predicted Mach number distribution for transonic Ringleb's flow obtained using 4th-order CENO scheme; and  $L_1$ ,  $L_2$ , and  $L_\infty$  error norms in the predicted solution density (b) and pressure (c) for transonic Ringleb's flow obtained using the 3rd- and 4th-order CENO schemes as a function of mesh density. The error norms for the 4th-order CENO scheme are shown for both reconstruction of primitive variables (PV) and of conserved variables (CV).

### 5.6.4 Subsonic Flow Past Circular Cylinder

The accuracy and performance of the proposed CENO schemes are now further examined by considering subsonic flow past a circular cylinder with a free-stream Mach number of  $M_\infty = 0.38$ . The predicted Mach number distributions obtained using the 2nd- and 4th-order CENO schemes are given in Fig. 5.27(a) and (b), respectively. It is quite noticeable that,

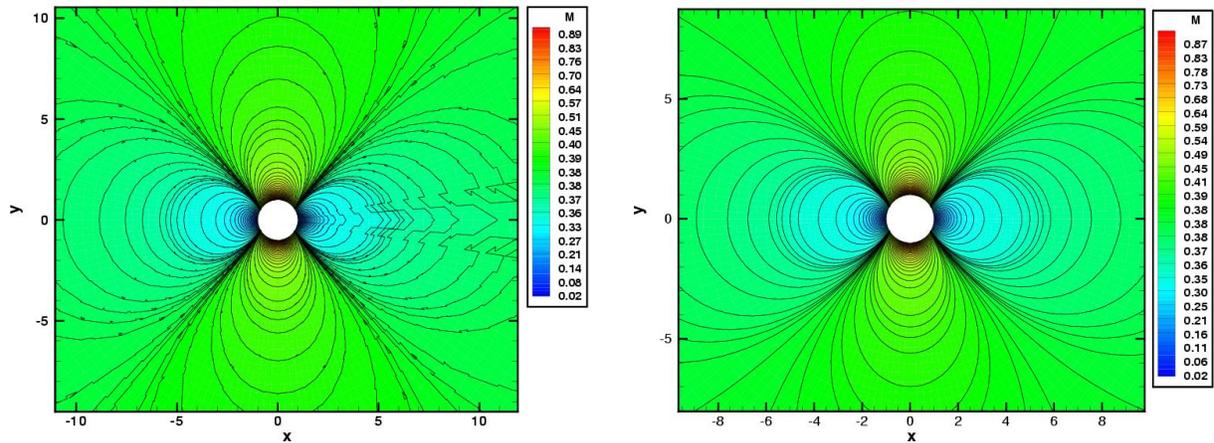
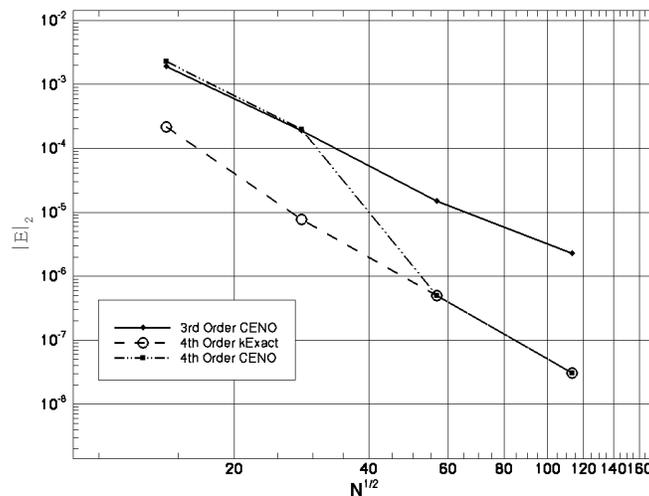
(a) 2nd-order ( $k=1$ ) CENO(b) 4th-order ( $k=3$ ) CENO(c)  $L_2$  error norms

Figure 5.27: Predicted Mach number distributions for  $M=0.38$  subsonic flow past a circular cylinder obtained using (a) the 2nd-order CENO and (b) the 4th-order CENO on a  $80 \times 40$  grid; (c)  $L_2$  norms of the error in entropy for the subsonic flow past a circular cylinder.

for the same mesh resolution, the 4th-order CENO solution is significantly more accurate than the 2nd-order CENO result. To quantify these findings, the  $L_2$  norms of the error in the predicted entropy (the flow is homentropic and the entropy is constant) are determined on a sequence of four structured O-grids ranging in size from  $20 \times 10$  to  $160 \times 80$  and shown in Fig. 5.27(c). As the mesh is refined, the slopes of the  $L_2$ -norms for the 3rd- and 4th-order schemes approach  $-2.74$  and  $-4.02$ , respectively.

### 5.6.5 Supersonic Flow Past Circular Cylinder

To demonstrate the ability of the CENO schemes to handle robustly flows with both smooth and non-smooth regions, the predicted Mach number distribution obtained on a  $160 \times 160$  grid  $M = 1.5$  flow past a circular cylinder is shown in Fig. 5.28(a). By reducing to limited linear reconstruction near the shocks in the flow, a monotone solution is obtained using the 4th-order scheme. The regions of the flow in which the CENO scheme uses the unlimited cubic and limited linear reconstruction procedures are shown in Fig. 5.28(b). As can be easily observed by comparing the two figures, the regions in which the numerical procedure

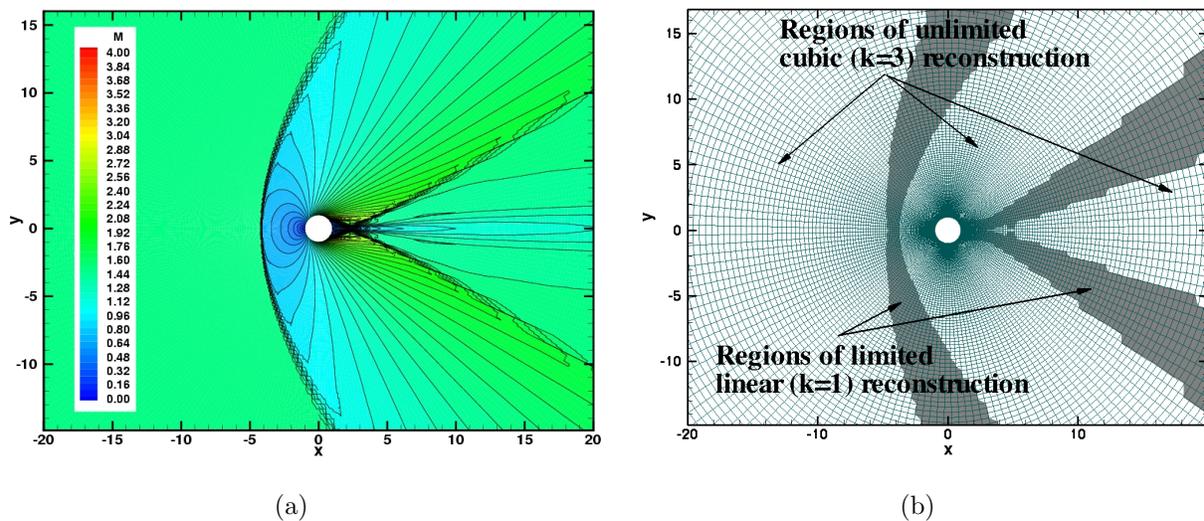


Figure 5.28: (a) Predicted Mach number distribution for  $M = 1.5$  flow past a circular cylinder obtained using the 4th-order CENO scheme on a  $160 \times 160$  grid; and (b) regions in which the CENO scheme uses unlimited cubic and limited linear reconstruction for  $M = 1.5$  flow past a circular cylinder.

uses the limited linear reconstruction correspond to the location of the bow shock and the shocks occurring at the back of the cylinder.

## 5.7 Solution of the Euler Equations with AMR

### 5.7.1 Supersonic Flow Past Circular Cylinder

The application of the proposed high-order AMR CENO finite-volume scheme to the solution of supersonic flow past a circular cylinder is again considered. In this case the free-stream Mach number is  $M_\infty = 2.1$  and the capability of the high-order scheme is examined in conjunction with AMR. The predicted pressure distributions obtained using the 4th-order CENO scheme on a final mesh consisting of 2,150 blocks and 215,000 computational cells is shown in Fig. 5.29(a). Corresponding to this solution, the regions of the flow in which the CENO scheme uses the unlimited cubic and limited linear reconstruction are shown in Fig. 5.29(b). The sequence of adaptively refined meshes leading to the final refined mesh starting with an initial mesh that consists of 8  $10 \times 10$  solution blocks is depicted in Fig. 5.30.

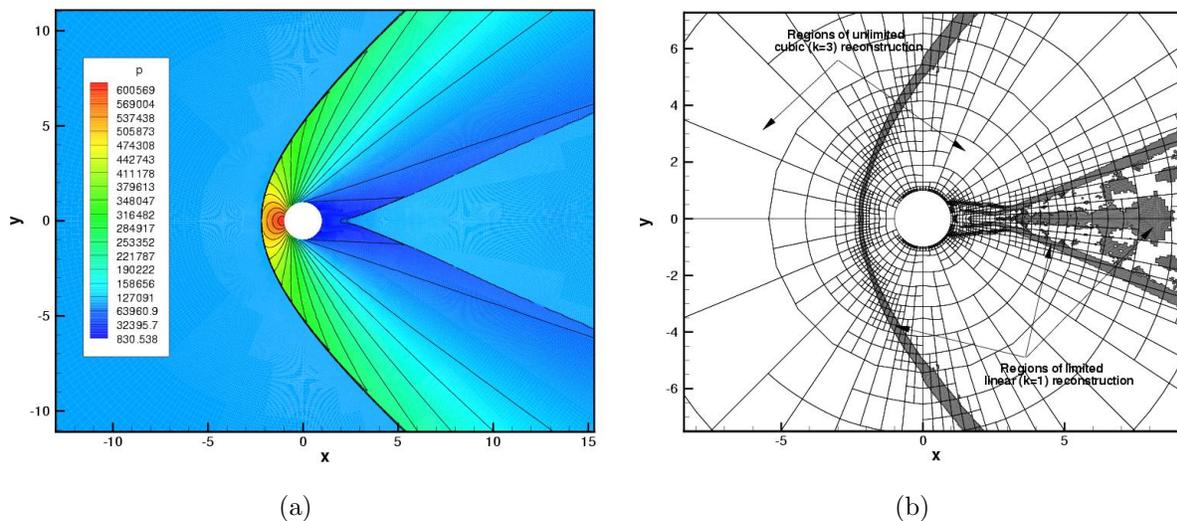


Figure 5.29: (a) Predicted pressure distribution for  $M = 2.1$  flow past a circular cylinder obtained using the 4th-order CENO scheme on the final refined AMR mesh; and (b) corresponding regions in which the CENO scheme uses unlimited cubic and limited linear reconstruction for  $M = 2.1$  flow past a circular cylinder.

The results clearly show that the proposed AMR scheme in conjunction with the  $h$ -refinement criteria based on the smoothness indicator of the hybrid CENO reconstruction technique is capable of refining both under-resolved (in-accurate) and non-smooth regions of the solution and will not unnecessarily refine resolved solution content. The bow shock and shocks arising at the base of the cylinder are well identified by the smoothness indicator and well resolved by the AMR procedure.

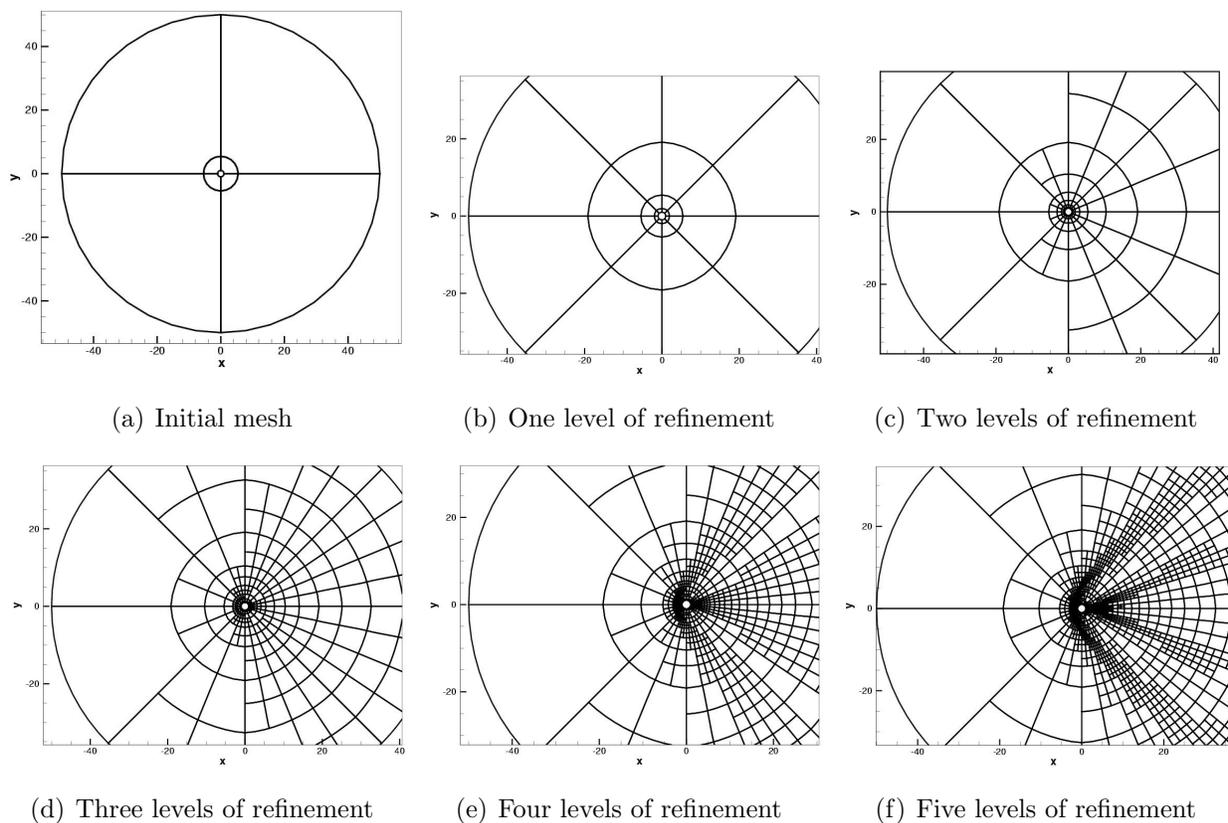


Figure 5.30: *Sequence of adaptively refined AMR mesh for supersonic flow past a circular cylinder: (a) initial mesh with 8  $10 \times 10$  blocks; (b) first refined mesh with 32  $10 \times 10$  blocks; (c) second refined mesh with 110  $10 \times 10$  blocks; (d) third refined mesh with 320  $10 \times 10$  blocks; (e) fourth refined mesh with 908  $10 \times 10$  blocks; (f) final refined mesh with 2150  $10 \times 10$  blocks and refinement efficiency of  $\eta=0.737$ .*

### 5.7.2 Smooth Translation of Density Variation

Finally, in order to estimate the accuracy of the CENO algorithm in conjunction with the AMR procedure, the linear advection of the density variation  $\rho(\mathbf{x}) = 1 - \tanh^2(\mathbf{x} - \mathbf{x}_o)$  from  $\mathbf{x}_o(10, 10)$  along a  $30^\circ$  direction is considered in a rectangular box. The Mach number of the advecting flow is  $M = 0.7$  and the final time of interest is  $t = 100$  ms. Although this problem can also be solved using only the advection equation described in Chapt. 2.1 and does not require the full system of Euler equations, its presentation as an Euler simulation is only for the purpose of exemplifying first different qualitative aspects of the AMR CENO finite-volume algorithm and then to undergo this quantitative accuracy assessment.

The initial condition as well as the starting meshes are shown in Fig. 5.31(a). In Fig. 5.31(b) it is shown the final solution and the meshes at the end of the advection. The error norms based on the error measurements at the end of the advection are shown in Fig. 5.32 for uniform and dynamically refined AMR Cartesian meshes.

The plot of the error norms of this problem shows that for targeted solution accuracies less than  $10^{-3}$ , the number of computational cells required by the 4th-order CENO scheme

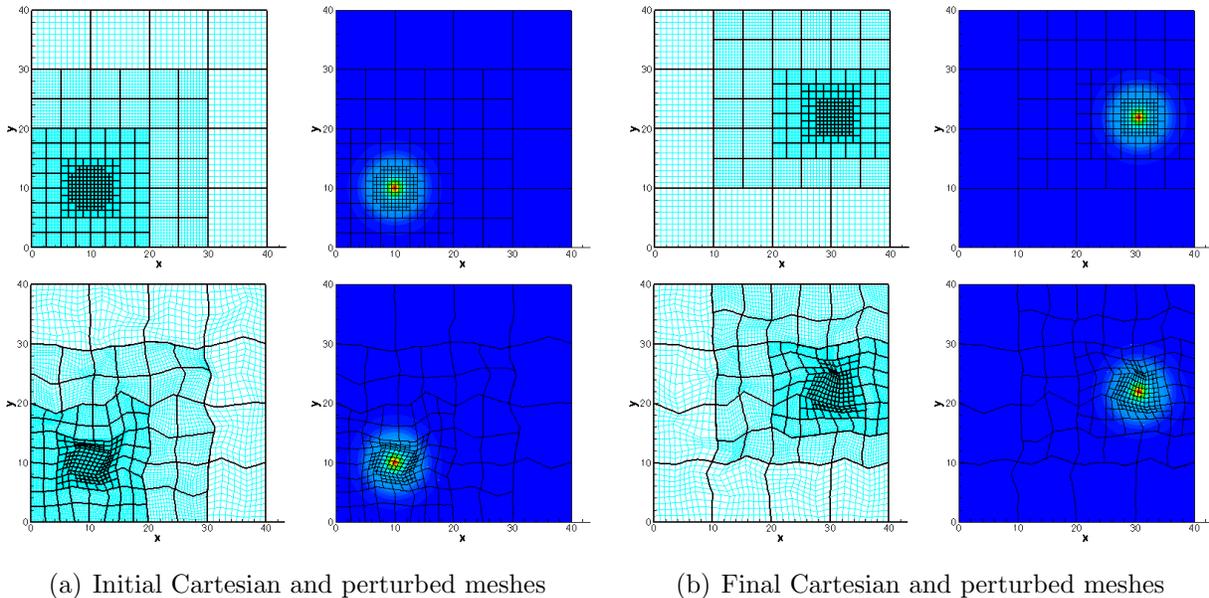


Figure 5.31: (a) Representation of initial density variation  $\rho(\mathbf{x}) = 1 - \tanh^2(\mathbf{x} - \mathbf{x}_o)$ ,  $\mathbf{x}_o(10, 10)$ , on the starting Cartesian and disturbed AMR meshes; (b) Final solution representation on both Cartesian and perturbed AMR meshes at the end of the advection.

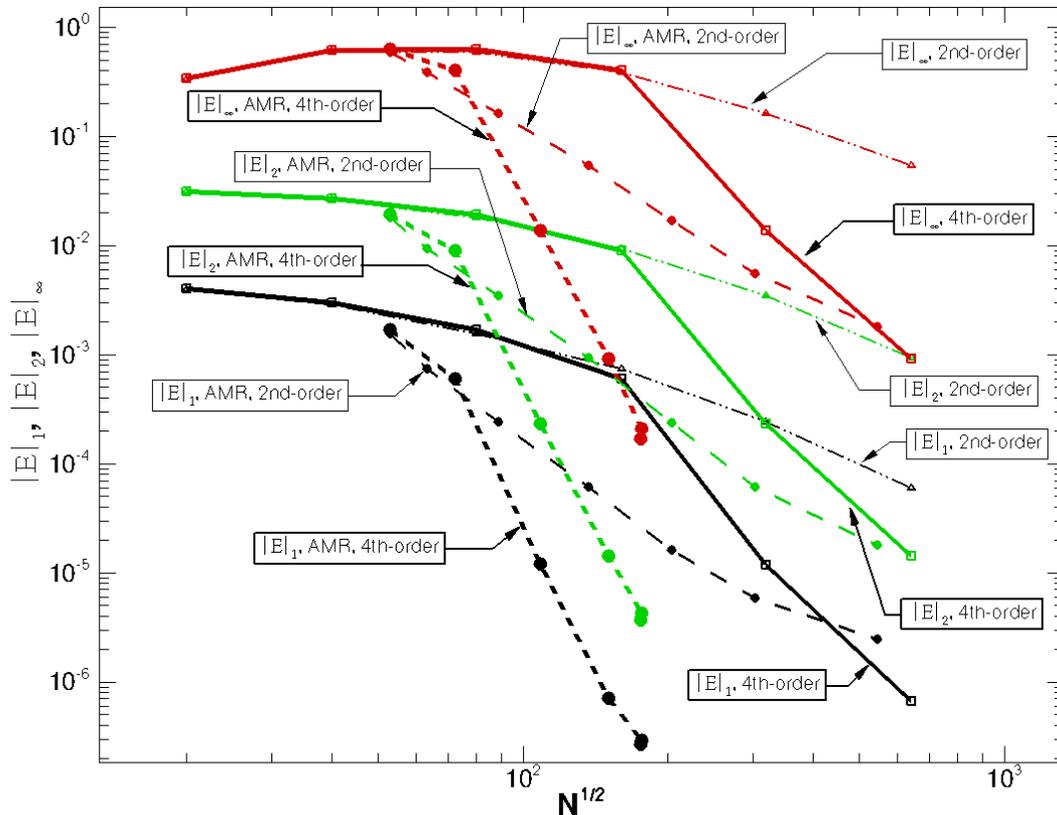


Figure 5.32:  $L_1$ ,  $L_2$  and  $L_\infty$  norms of the error at the end of the advection (see Fig. 5.31(b)) as a function of mesh density obtained using a 2nd-order limited scheme and the 4th-order CENO method on both uniform and dynamically refined AMR Cartesian meshes.

applied in conjunction with AMR is about 10 to 20 times smaller in each coordinate direction than what a limited 2nd-order scheme on uniform mesh would require. This would imply a reduction in mesh requirements by a factor of 100-400 for two-dimensional problems and by a factor of 1,000-8,000 in three dimensions, thus further demonstrating the great potential of the high-order AMR approach. The plot also reveals the fact that if low solution error is required, the high-order scheme on uniform mesh may require fewer computational cells than a 2nd-order method used in conjunction with AMR, which confirms Jameson's findings [12] mentioned earlier in the motivation section of this thesis (see Sect. 1.1). Note that this comparison has not taken into account the computational costs of the AMR and high-order schemes as compared to their uniform mesh and low-order method counterparts. In particular, this was not attempted here due to inequities and inefficiencies in the time marching schemes which may lead to unfair comparisons.

## 5.8 Solution of the Navier-Stokes Equations on Fixed Mesh

The verification test problems and analyzes considered so far have demonstrated the accuracy and robustness of the proposed CENO approach for discretization of equations including both hyperbolic and elliptic operators. Finally, the application of the proposed high-order CENO FV algorithm to numerical simulation of flows governed by the full set of Navier-Stokes equations described in Sect. 2.3 is considered herein to illustrate the predictive capabilities of the high-order method for prediction of laminar viscous flows. Numerical simulations for fixed mesh are presented in this section followed by AMR results in the next one.

### 5.8.1 Laminar Flat Plate

The accuracy of the proposed algorithm for prediction of viscous flows is investigated first by considering the laminar flow over a flat plate at zero incidence. The free-stream Mach number and Reynolds number, based on the length of the plate,  $L$ , are  $M_\infty = 0.1$  and  $Re = 10,000$ , respectively. The domain considered for this problem is the rectangular box of dimensions  $4L \times 2L$ , where the first dimension is in the flow direction. An adiabatic viscous wall boundary condition, which extends over the second half of the lower boundary, has been used to simulate the presence of the flat plate. For the rest of the lower boundary a symmetry plane BC has been used. Note that both of these BC types have been enforced via the constrained reconstruction approach described in Sect. 3.6.3. The BC for the upper boundary of the domain was set to extrapolate data from the interior but to enforce the free-stream pressure. Subsonic inflow and outflow BCs have been used for the left and right boundaries, respectively.

In Fig. 5.33 the solution obtained with the 4th-order CENO algorithm ( $k=3$ ) described above on a  $160 \times 160$  Cartesian mesh stretched towards the plate is compared against the Blasius analytical solution for the incompressible boundary layer equations. The predicted boundary layer non-dimensional velocity profiles,  $u/U_\infty$  and  $\phi = v\sqrt{Re_x}/U_\infty$ , are plotted against the similarity variable,  $\eta = y\sqrt{U_\infty/2\nu x}$ , at  $Re_x = U_\infty x/\nu = 8,000$  in Fig. 5.33(a), where  $\nu$  is the kinematic viscosity. The number of computational cells in the boundary layer at this sampling station, considering  $\eta=5$  as the boundary layer edge, is 65. The predicted

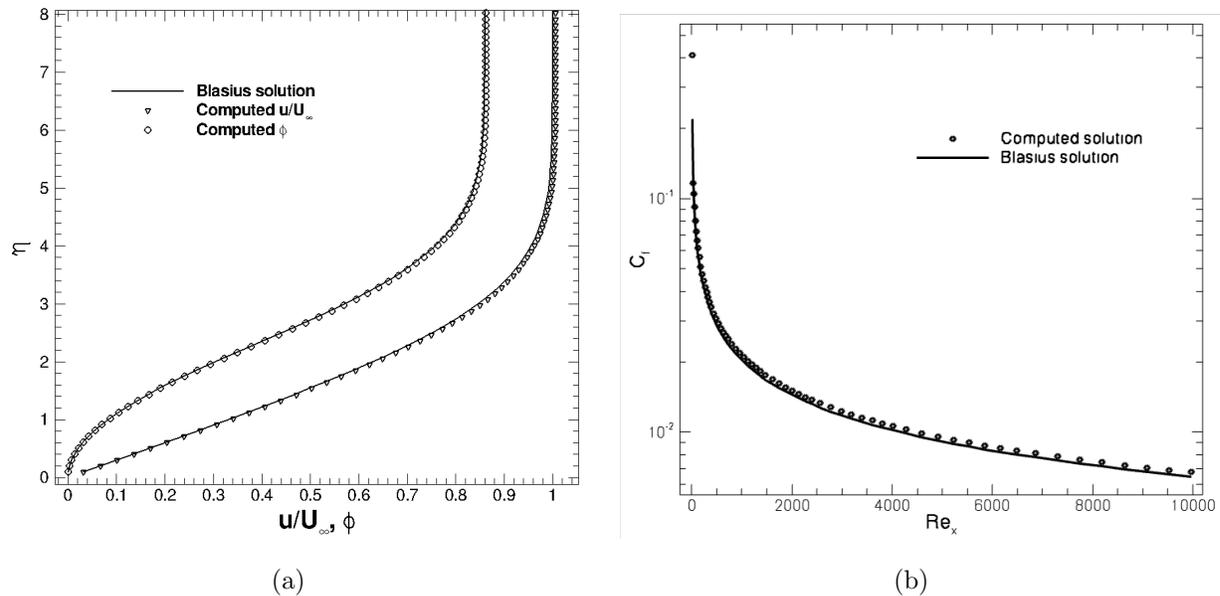


Figure 5.33: (a) Variation of the non-dimensional velocity components at  $Re_x = 8,000$ ; and (b) Distribution of skin friction coefficient along the plate.

skin friction coefficient along the plate is given in Fig. 5.33(b). The total drag coefficient estimated for the plate is  $C_D = 0.01349$  which compares very well with the prediction based on Blasius drag law,  $C_D = 0.01328$ . Note also that the same algorithm predicts  $C_D = 0.01351$  on a mesh with half the resolution of that for which the solution is shown. It is evident that the results are in very good agreement with the Blasius solution thereby providing support for the validity of the spatial discretization procedure for laminar fluid flows.

### 5.8.2 Steady Laminar Subsonic Flow Past Circular Cylinder

The predictive capabilities of the proposed CENO scheme are further examined by considering the laminar subsonic flow past a circular cylinder with a free-stream Mach number of  $M_\infty = 0.1$ . Numerical simulations were carried out for two Reynolds numbers,  $Re = 30$  and  $Re = 110$ , for which no three-dimensional effects are present [169]. For  $Re = 30$ , the solution is a steady wake behind the cylinder as the one shown in Fig. 5.34. Details about the unsteady flow corresponding to  $Re = 110$  are given in the next section.

The geometry considered in these numerical simulations is the domain between two concentric cylinders of which the inner cylinder has a diameter  $d_i = 0.0001$  and the outer cylinder

der has been positioned at 40 inner diameters. Five uniform but stretched towards the inner cylinder O-grid meshes were used during this study with the following resolutions:  $M_1 = 3,200$ ,  $M_2 = 12,800$ ,  $M_3 = 51,200$ ,  $M_4 = 204,800$  and  $M_5 = 3,276,800$  computational cells. The investigation of the steady laminar subsonic flow past circular cylinder has been carried out with two numerical methods: a 4th-order (i.e. quartic reconstruction) CENO scheme and a second-order method which uses a combination of piecewise linear least-squares reconstruction and a diamond path reconstruction for the discretization of the hyperbolic and elliptic operator, respectively, the details of which can be found in [98]. The imposed boundary conditions for the inner cylinder were no slip and adiabatic wall, both being enforced for the CENO scheme with the constrained reconstruction algorithm described in Sect. 3.6.3. The CENO predictions of the drag coefficients were compared against those of the second-order scheme and against those calculated with the curve fits proposed by Henderson [170] which are based on an thorough numerical investigation near the onset of vortex shedding. Note that Henderson's predictions were verified against experimental data in the aforementioned paper.

The predicted Mach number distribution for laminar viscous flow around the cylinder at

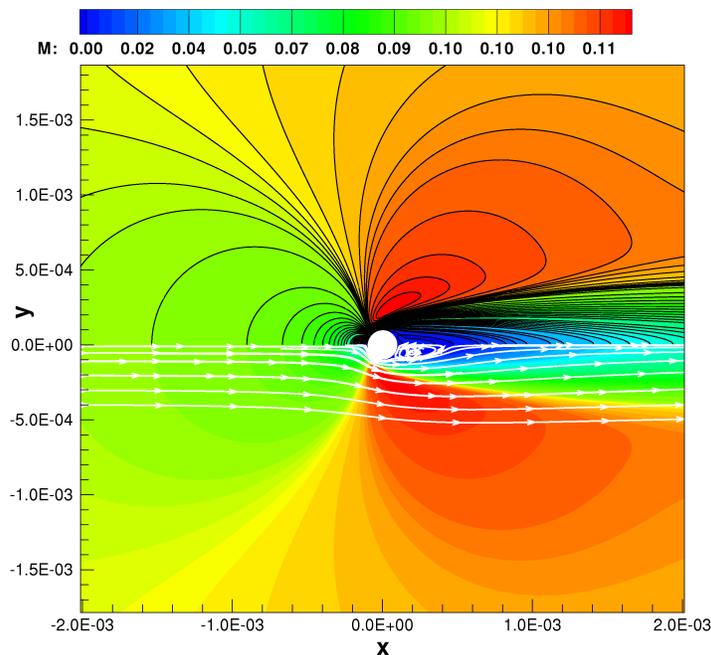


Figure 5.34: *Prediction of Mach number and streamlines for laminar flow over cylinder at  $M_\infty = 0.1$  and  $Re = 30$  on a mesh with 51,200 cells.*

$Re=30$  obtained using the 4th-order ( $k=4$ ) CENO finite-volume scheme on the body-fitted mesh  $M_3$  is shown in Fig. 5.34. The drag coefficients computed based on the high-order numerical solutions for this problem were  $C_D = 1.7498$  with  $M_1$ ,  $C_D = 1.7512$  with  $M_2$ ,  $C_D = 1.7522$  with  $M_3$ ,  $C_D = 1.7528$  with  $M_4$ , and  $C_D = 1.7541$  with  $M_5$ , respectively. These values are in good agreement with Henderson's prediction of  $C_D = 1.737$  and the experimental data available in the literature [120, 171].

The drag coefficients predicted by both 4th- and 2nd-order schemes are plotted in Fig. 5.35. An analysis of these results reveals several observations worth discussing. First, the predictions of both high-order and second-order methods converge to the same drag coefficient value as the mesh is refined, thus providing confidence in the validity of the two approaches. At its minimum, the difference between the predictions of the two schemes becomes only 6.5 drag counts, which is obtained on the finest mesh as can be seen in Fig. 5.35(b). Secondly, the error of the CENO predicted drag coefficient on mesh  $M_1$  relative to the best estimation (i.e., the value obtained on mesh  $M_5$ ), is only 0.245% while the mesh resolution is 1,024 times lower. In contrast, the 2nd-order method exhibits a 1.966% relative error between its drag coefficient predictions on the same meshes, which is about 8 times larger than that

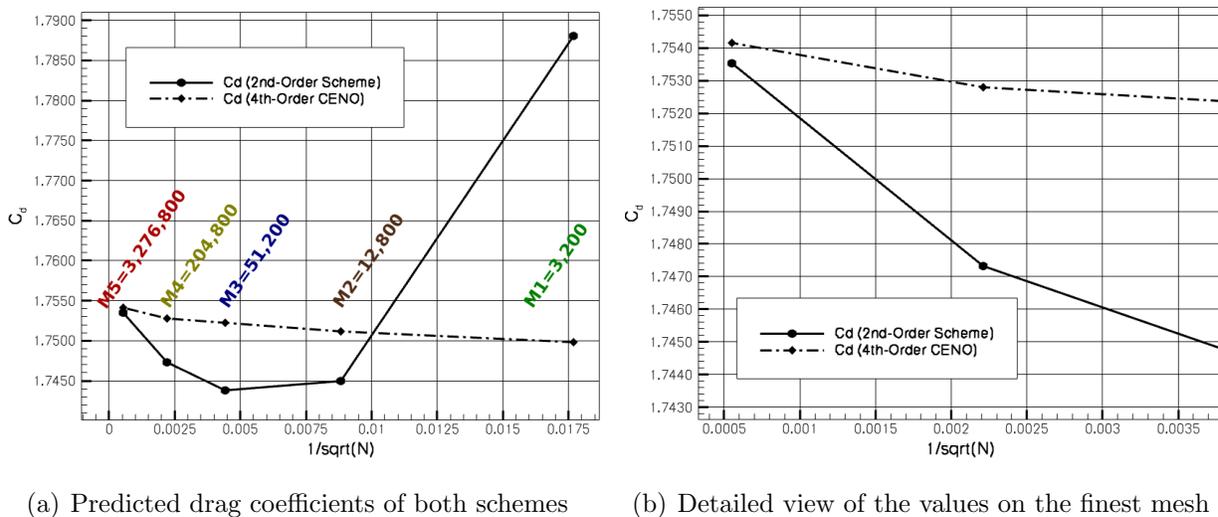


Figure 5.35: Comparison of drag coefficients predicted by the 4th-order CENO method and the 2nd-order scheme described in the text for the laminar flow over cylinder at  $M_\infty = 0.1$  and  $Re=30$ . The x-axis represents the equivalent mesh spacing determined as  $\Delta x = \frac{1}{\sqrt{N}}$ . A zoom-in of the plot in (a) for the finest meshes is shown in (b).

corresponding to the high-order CENO scheme. Furthermore, the data shows that in order to obtain the same drag coefficient error as the CENO scheme on mesh  $M_1$  (i.e., 0.245%), the 2nd-order method requires a computational grid that is about 132 times finer than  $M_1$ .

It should be noted however, that no attempt has been made here to quantify the computational savings generated by the high-order procedure as a result of a reduced mesh resolution due to inefficiencies and inequities in the time-marching schemes which may lead to unfair comparisons between the two spatial discretizations. As the number of spatial residual evaluations required to converge to a steady-state solution greatly depends on the employed time-marching strategy, a better alternative to assess the cost of the two numerical methods is to compare the computational cost of only one spatial residual evaluation for each computational cell. Thus, time measurements on single and multiple (i.e. four) Intel Xeon CPUs, X5460@3.16GHz, with shared memory have been carried out for both high- and low-order schemes with the mesh  $M_3$  which contains 128 blocks. The measurements indicate that the high-order CENO scheme requires about  $0.0524 \pm 0.0024$  ms for one residual evaluation per each computational cell which is about 2.9 to 3.4 times more expensive than the cost of the second-order counterpart. These numbers imply that the proposed high-order CENO scheme has the potential to provide large computational savings when accurate drag predictions are required but they will ultimately depend on how effective the time-marching strategy is in eliminating the transient component of the high-order solution.

The parallel performance of the proposed high-order CENO algorithm has also been assessed based on the steady laminar subsonic flow past circular cylinder by measuring/evaluating the parallel speedup and parallel efficiency, which are defined as

$$S_p = \frac{t_1}{t_p} \quad \text{and} \quad E_p = \frac{S_p}{p}, \quad (5.9)$$

respectively, where  $t_1$  and  $t_p$  are the execution times required to solve the problem by a single processor and by  $p$  processors, respectively. Herein, the parallel speedup, also known as strong scaling, has been measured by considering a fixed problem size of 2,048  $20 \times 20$  blocks and a fixed number of explicit time steps, and performing the simulation on an increasing number of processors  $p$ , which had a maximum value of 1,024 in this study. Note that a perfect speedup corresponds to the ideal situation in which the execution time of the simulation on  $p$  computing cores is  $p$  times smaller than  $t_1$ . In Fig. 5.36 the parallel speedup and efficiency achieved by the CENO method as a function of the number of computing cores

is plotted against the corresponding ideal variations of the parallel-performance parameters. The results show that the 4th-order CENO method has good parallel scalability even for computational blocks of relatively-small number (e.g., 400) of cells and manages to achieve parallel efficiencies of 0.78 and 0.72 on 512 and 1,024 computing cores, respectively. The drop incurred in parallel efficiency with increasing the processor count is an expected one as not only additional inter-block parallel communication is required as more processors are used but also the amount of computational work per computing core decreases significantly.

### 5.8.3 Unsteady Laminar Subsonic Flow Past Circular Cylinder

For  $Re = 110$  vortex shedding occurs for laminar flow past a cylinder, as illustrated in Fig. 5.37, and the unsteady flow around the cylinder is characterized by a periodic variation of lift and drag coefficients, as shown in Fig. 5.38(a)–(b). The results show that the frequency of the lift variation is double the frequency recorded for the drag, which agrees well with

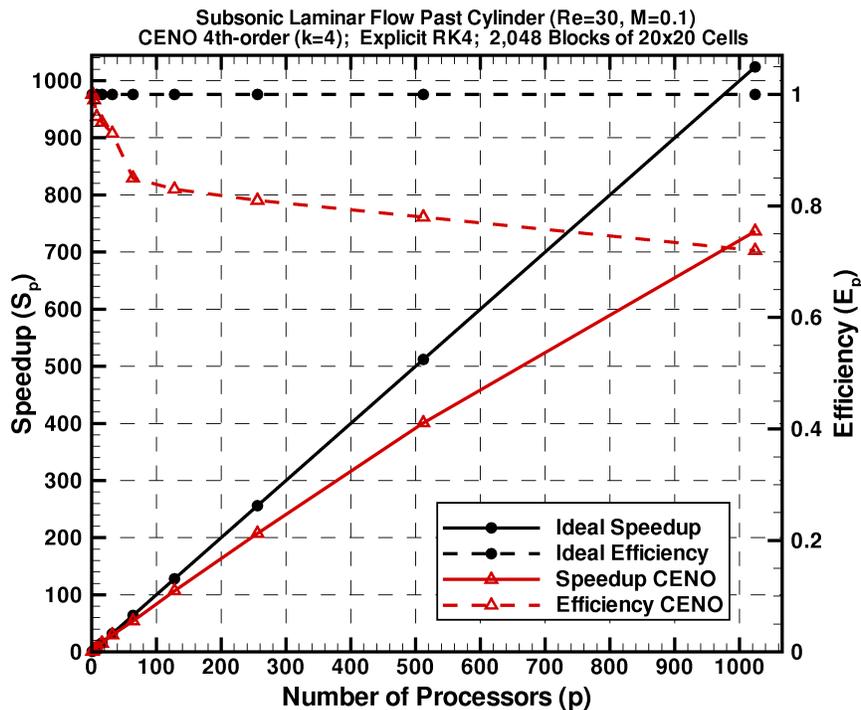


Figure 5.36: The parallel speedup and efficiency achieved for the steady laminar flow past cylinder by the 4th-order CENO algorithm as a function of the number of computing cores.

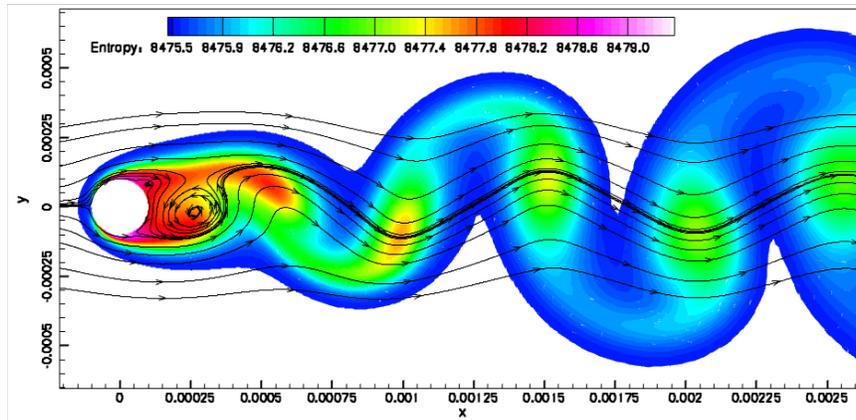


Figure 5.37: Entropy prediction and streamlines for vortex shedding flow past circular cylinder at  $Re=110$  obtained using the 4th-order ( $k=4$ ) CENO scheme on mesh  $M_3$ .

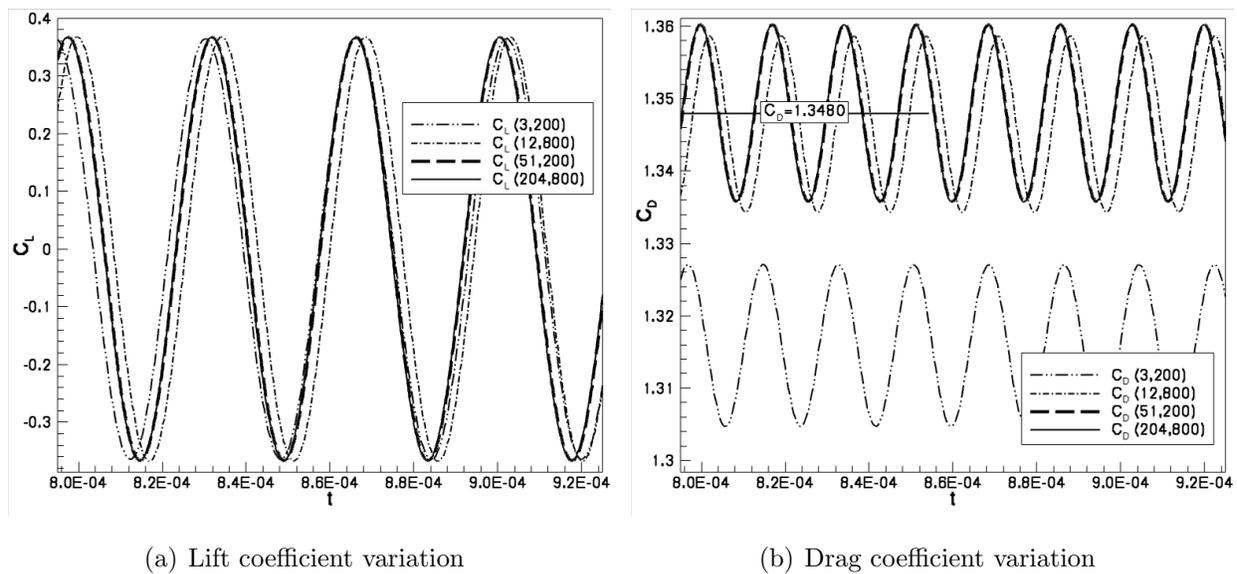


Figure 5.38: Time dependent variation of the (a) lift and (b) drag coefficients for the vortex shedding flow obtained using the 4th-order CENO scheme on  $M_1$ ,  $M_2$ ,  $M_3$  and  $M_4$  meshes. The predictions on  $M_3$  and  $M_4$  basically overlaps. The mean drag on  $M_4$  is indicated in (b).

the observations reported in literature. Note also that a converged solution for both lift and drag coefficient variations requires around 51,200 cells (i.e.  $M_3$ ), at which resolution there is insignificant difference relative to the solution obtained on the finest mesh. The mean drag coefficient determined based on the converged solution is  $C_D = 1.3480$  which is again in good agreement with Henderson's result of  $C_D = 1.34$ .

## 5.9 Solution of the Navier-Stokes Equations with AMR

### 5.9.1 Unsteady Laminar Subsonic Flow Past Circular Cylinder

To demonstrate the predictive capabilities of the CENO scheme in conjunction with the block-based AMR algorithm for solutions of the Navier-Stokes equations governing laminar viscous flows, the unsteady vortex shedding flow over the cylinder described in the previous section is again considered. The outer cylinder or boundary of the computational domain has been positioned in this case at 80 inner diameters  $d_i$  instead of the previously used value of 40 in order to test the capabilities of the adaptive high-order CENO algorithm to resolve accurately flow features occurring over a wide computational domain with a reduced number of grid elements. An instance of the periodic steady-state solution obtained on the initial O-grid mesh of 32 self-similar  $8 \times 8$  solution blocks and 2,048 computational cells is shown in Fig. 5.39 and the corresponding drag coefficient variation with respect to physical time is depicted in the lower left corner of Fig. 5.40. As can be seen in Fig. 5.39 the flow features near the cylinder and in the wake generated behind it are not adequately resolved on this very coarse mesh even by the 4th-order ( $k = 4$ ) scheme and consequently, the mean drag coefficient is well under-predicted.

Automatic mesh adaptation directed by the proposed  $h$ -refinement criterion based on the CENO smoothness indicator of the velocity components has been further applied to this very coarse solution in the way described next to improve the representation of relevant flow features and the prediction of the drag coefficient. The initial strategy employed up to physical time  $t = 0.006225$  was to incrementally improve the solution accuracy of the unsteady flow by generating a sequence of AMR meshes on which periodic steady-state solutions are obtained. The five meshes generated by the block-based refinement algorithm in this simulation step consists of 80, 221, 536, 1,349 and 2,486 blocks  $8 \times 8$  and 5,120, 14,144, 34,304, 86,336 and 159,104 cells. A selected plot of the instantaneous density contour lines and the entropy field is shown for each mesh in Fig. 5.41(a)–(e). As can be clearly observed from the solution plots, the regions of boundary layer detachment and the vortices in the wake behind the cylinder are progressively better resolved as the multi-block grid gets refined and the mesh resolution is increased especially near and behind the inner cylinder. The drag coefficient variation following from the predicted solution on each of the five meshes is

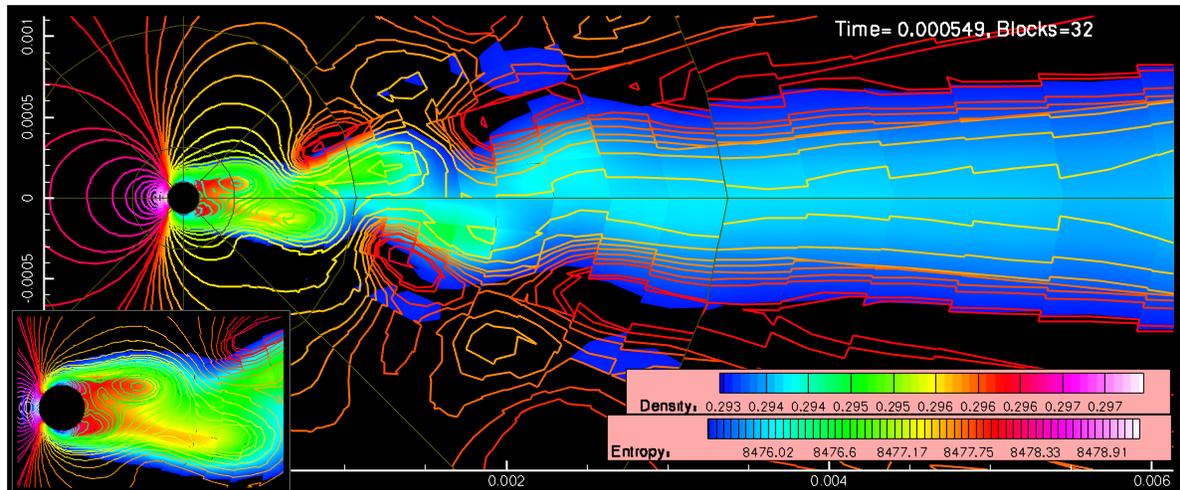


Figure 5.39: *Snapshot of entropy field prediction and density contour lines for vortex shedding flow past circular cylinder at  $Re = 110$  obtained using the 4th-order ( $k = 4$ ) CENO scheme on the initial AMR mesh at the indicated physical time. The number of grid blocks is shown in the top right corner and the block boundaries are depicted with dark green colour. The bottom left figure is a zoom-in of the flow configuration near the interior cylinder.*

depicted in Fig. 5.40, which also shows the mean  $C_D$  value for the mesh with 221 (Detail A), 536 (Detail B) and 2,486 blocks (Detail C).

Following the application of the first AMR strategy, automatic mesh adaptation has been further performed for this problem dynamically at regular intervals of physical time. After an initial constant increase, the first of which being to 4,175 blocks, the AMR algorithm generated and maintained a number of blocks in the range of 10,000 to 11,000 which corresponds to 10-12 levels of refinement and mostly varied their location so as to “follow” the unsteady flow features and shed vortices. This fact can be observed in Fig. 5.43 in which three consecutive solution snapshots are depicted to illustrate the distribution of the AMR blocks in the vicinity of the inner cylinder and the wake behind it. Notice that the domain covered by our plots span over 60 inner cylinder radii and the wake behind the cylinder gets well resolved even at large distances from the trailing edge. Notice also that the smoothness indicator recognizes the regions where interesting flow features occur such as vortices, boundary layers, flow detachment zones, and other areas characterized by sharp solution variations, and the block-based AMR algorithm adequately resolves them so as to achieve

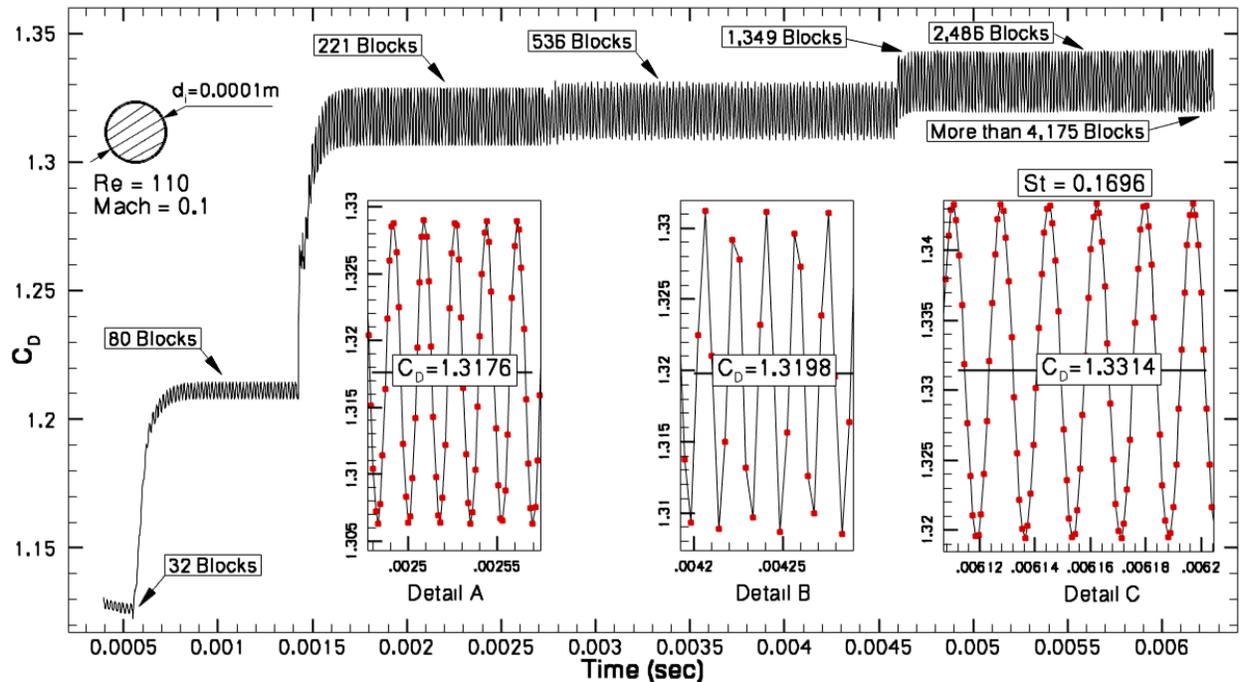
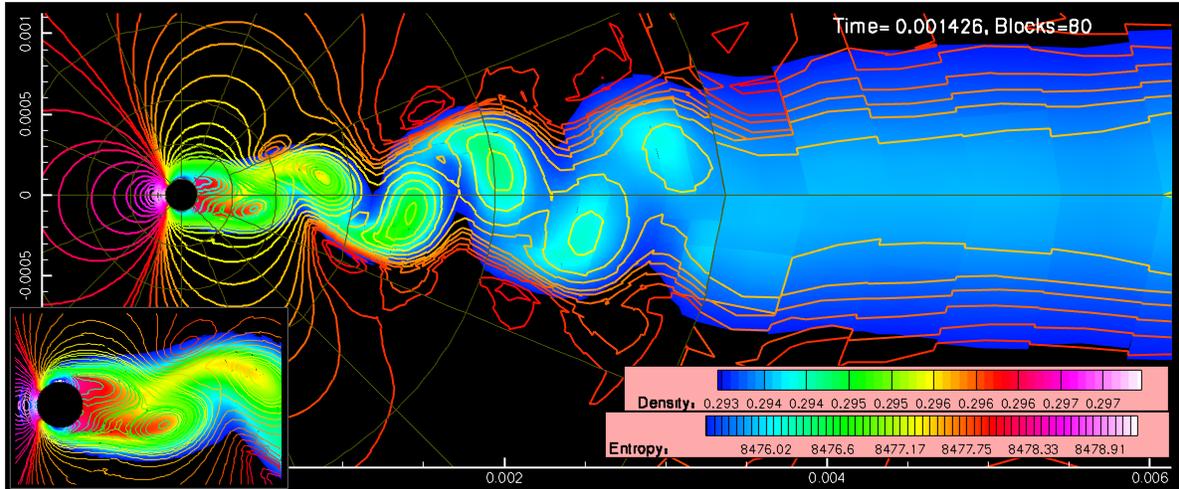


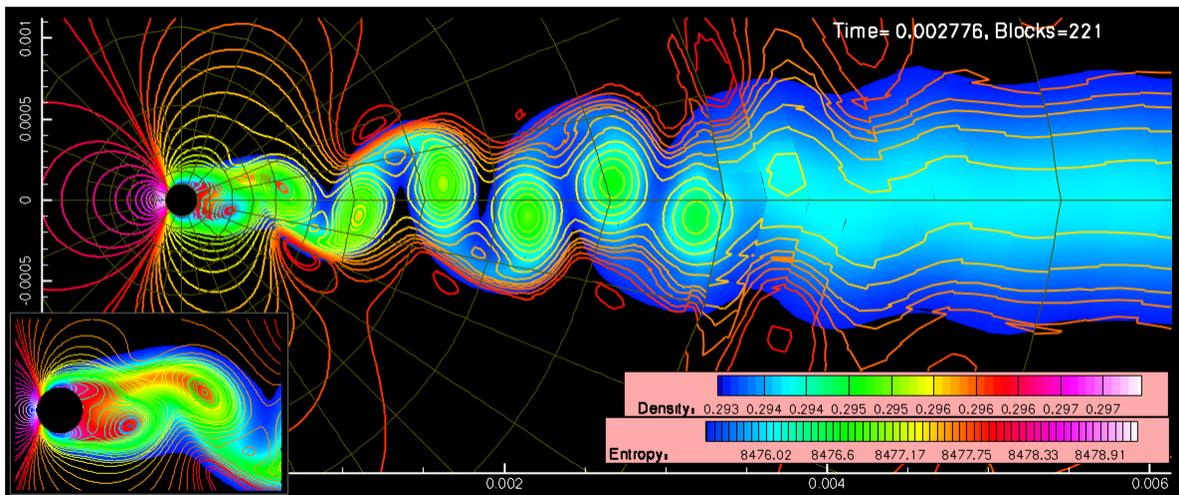
Figure 5.40: Variation of drag coefficient for unsteady vortex shedding flow past the cylinder as a function of physical time determined by the predicted flow field solution obtained using the 4th-order ( $k=4$ ) CENO scheme in conjunction with the block-based AMR algorithm.

the specified level of smoothness. Nevertheless, the level of mesh resolution imposes severe restrictions on the allowable time step in the explicit time-marching scheme and consequently, a tremendous computational effort is required to simulate the many vortex shedding periods necessary to transport the convective flow features resolved near the cylinder into the far wake. Our current dynamic simulation extends over almost one period and consequently, it is anticipated that better representation of the far vortices would have been possible if more shedding periods were achieved.

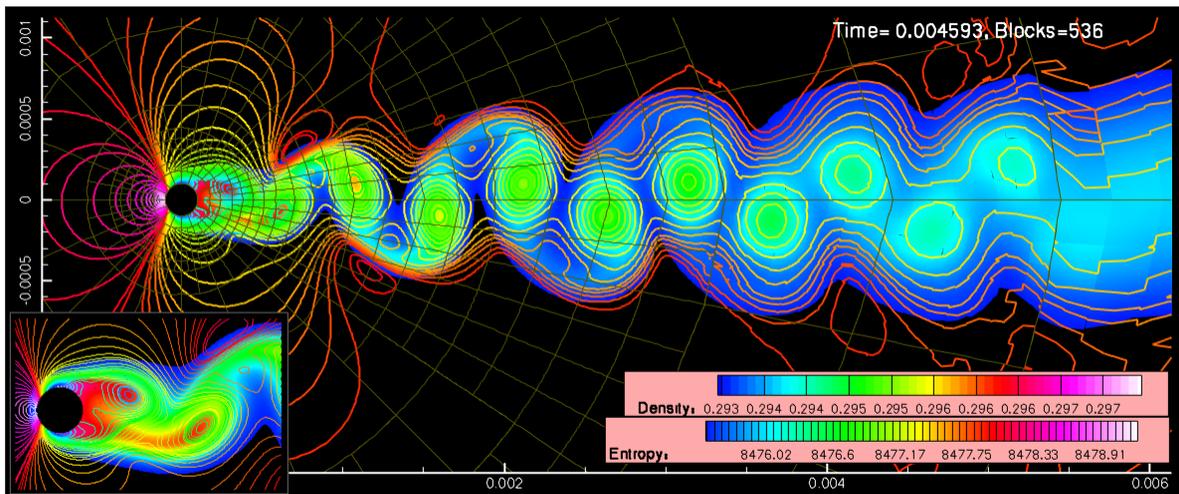
The  $C_D$  curve presented in Fig. 5.40 exhibits convergence to a grid-independent variation as the mesh is refined by the AMR algorithm and shows that the improvements generated by the predictions on the dynamically refined meshes are minor. Our best prediction of the mean drag coefficient and Strouhal number is  $C_D = 1.3314$  and  $St = 0.1696$ , respectively, which agrees well with values from the literature. Thus, the Strouhal number based on the relationship proposed by Roshko [172] to describe the best fit to experimental data is  $St = 0.1711$ . Furthermore, the predicted mean drag coefficient compares well with the value



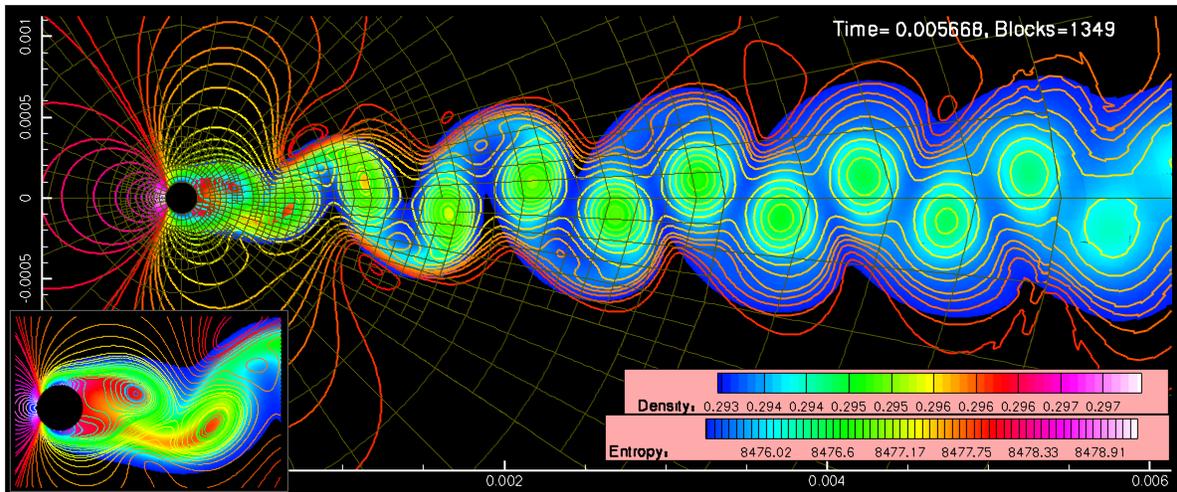
(a) Predicted flow field on the mesh with 80 blocks and 5,120 cell



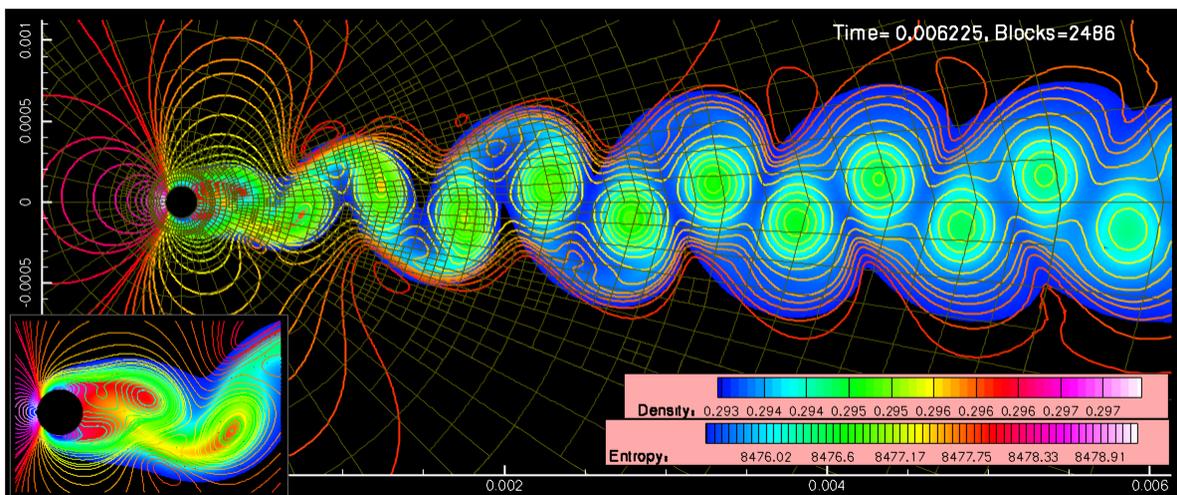
(b) Predicted flow field on the mesh with 221 blocks and 14,144 cell



(c) Predicted flow field on the mesh with 536 blocks and 34,304 cell



(d) Predicted flow field on the mesh with 1,349 blocks and 86,336 cell



(e) Predicted flow field on the mesh with 2,486 blocks and 159,104 cell

Figure 5.41: *Contour plots of the predicted entropy and density fields for unsteady vortex shedding obtained on the first five AMR meshes at various time instances.*

$C_D = 1.34$  reported in the computational study by Henderson [170]. Note that Sheard *et al.* [173] have also reported detailed computations of the drag for the cylinder and have compared their predictions against Henderson's and the experimentally measured drag coefficients from Wieselberger [171]. Thus, Fig. 5.42 shows the drag coefficients obtained for both the steady ( $Re = 30$ ) and unsteady ( $Re = 110$ ) laminar flows past circular cylinder studied in this thesis, plotted against the data from Sheard *et al.* [173]. The plot reveals that both predictions

by CENO algorithm compare very well with the previously reported values. Moreover, for  $Re = 110$ , it seems that CENO prediction is slightly more accurate with respect to Wieselberger's measurements than the curve fit value by Sheard *et al.* [173]. Without overstating this result, as our study has been carried out for only one Reynolds number and not for the wide range covered by the other authors, it is worth concluding that these findings provide strong support for the validity of the present implementation and indicate that the high-order CENO scheme in combination with the block-based AMR algorithm has the potential to provide reliable predictions of viscous laminar flows over complex geometries while significantly reducing the number of required computational elements.

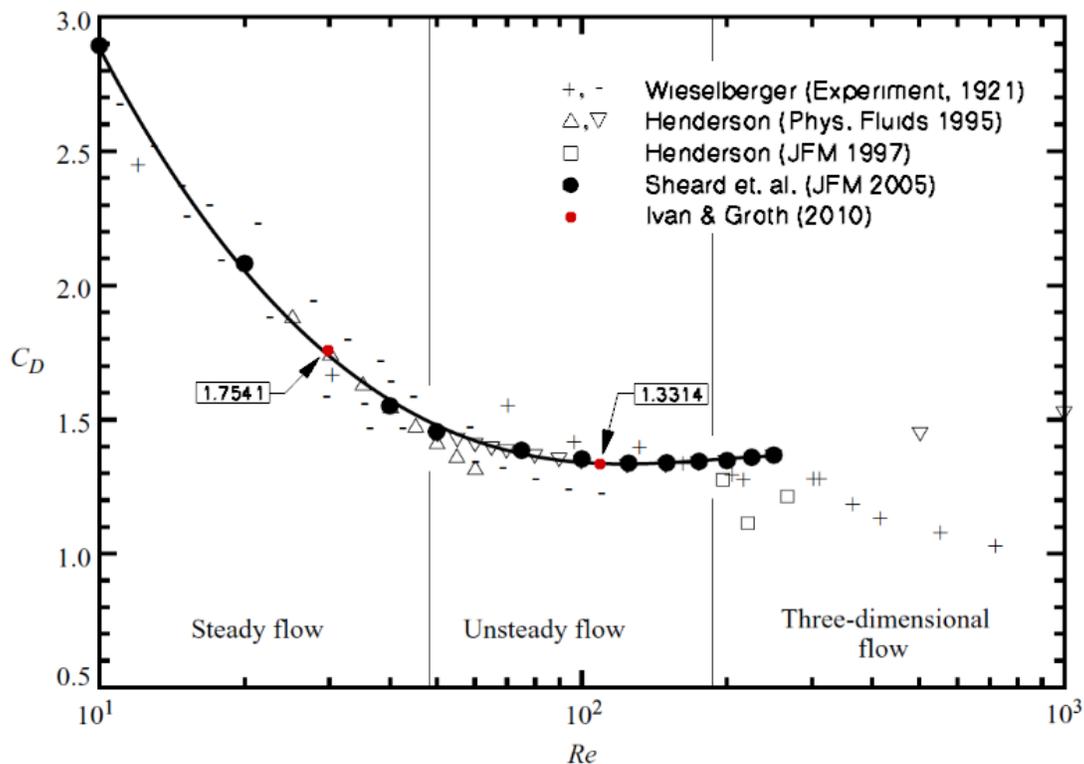
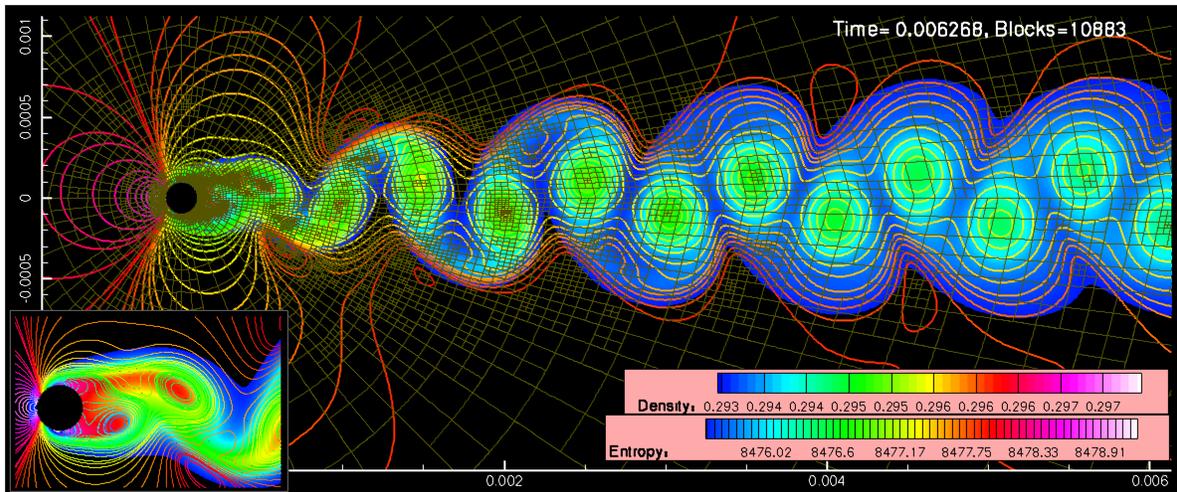
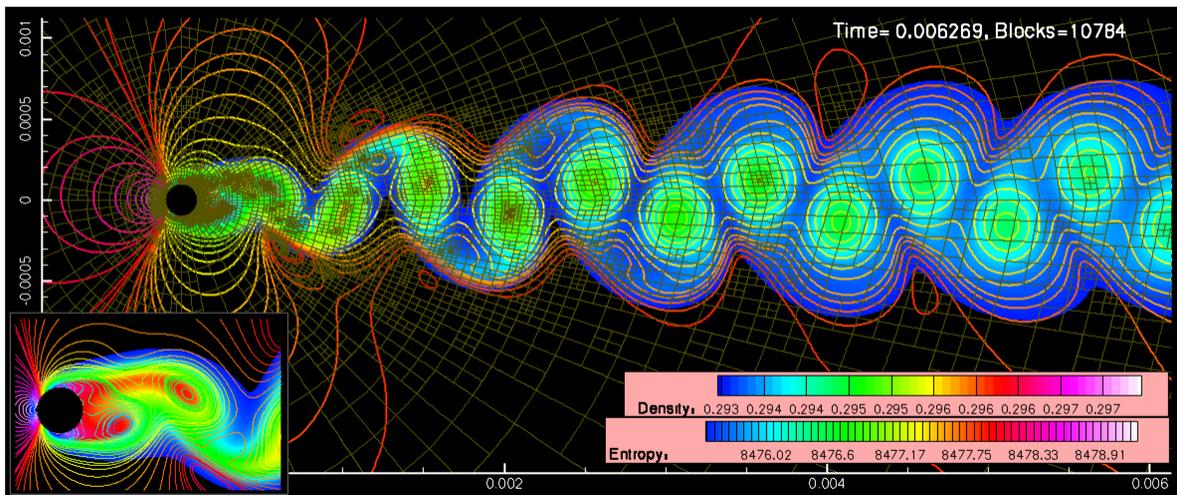


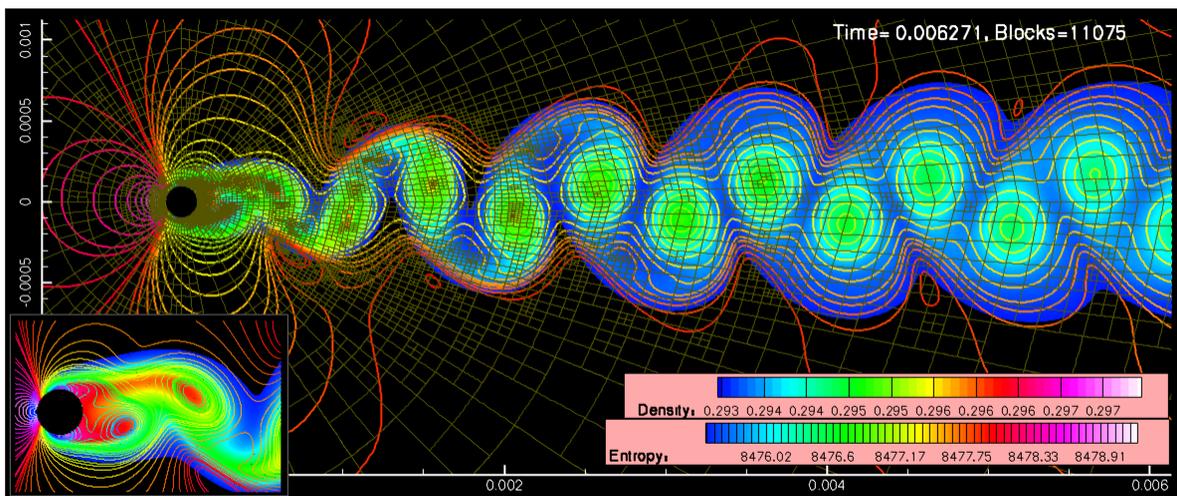
Figure 5.42: Depiction of the steady and mean drag coefficients predicted by the 4th-order CENO scheme for the flow past circular cylinder at  $Re = 30$  and  $Re = 110$  relative to previously reported values. The data obtained from Sheard *et al.* [173] includes, in addition to their computational predictions, the experimental measurements of Wieselberger (1921) and the two- and three-dimensional computations by Henderson (1995 and 1997, respectively). Additionally, the transitional Reynolds numbers from steady to unsteady flow (i.e.,  $Re \approx 47$ ) and from two- to three-dimensional flow (i.e.,  $Re \approx 189$ ) are indicated with vertical lines.



(a) Predicted flow field on a mesh with 10,883 blocks and 696,512 cell



(b) Predicted flow field on the mesh with 10,787 blocks and 690,368 cell



(c) Predicted flow field on the mesh with 11,075 blocks and 708,800 cell

Figure 5.43: Predicted flow fields on a sequence of dynamically refined meshes.

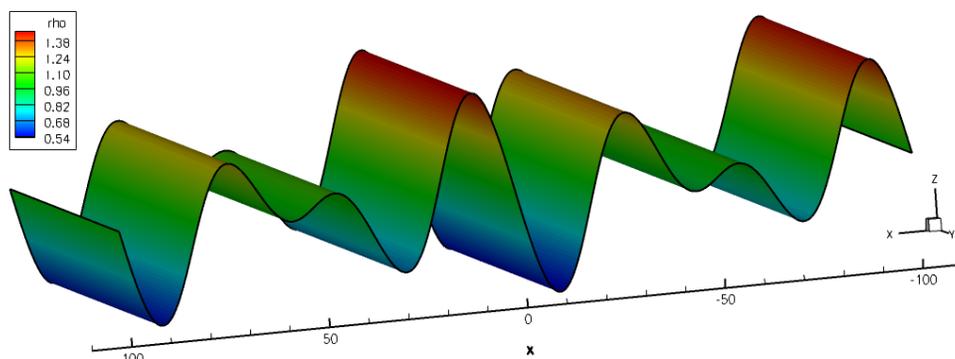
## 5.10 Computational Cost of High-Order Scheme

The final set of results considered herein as part of this thesis pertain to some estimates of the computational cost of the proposed high-order CENO method. They also attempt to shed some light on whether or not high-order schemes are capable of being less expensive than their standard second-order counterparts. It should be clear that from the perspective of performing practical calculations, the numerical properties of a high-order discretization procedure become mostly relevant only insofar as they contribute to achieving the ultimate goal of providing significant computational savings to a numerical simulation of a specified accuracy. Although the computational cost associated with different discretization orders can be influenced by many factors among which the talent and knowledge of the programmer are equally important, it represents an important issue to be addressed.

Therefore, the periodic linear convection of density variation in a rectangular box is considered herein to assess the computational cost of the 4th-order ( $k=3$ ) CENO scheme with a smoothness indicator cutoff of  $\mathcal{S}_c=3,500$  versus that of a limited 2nd-order method which uses the non-linear limiter proposed by Venkatakrishnan [147]. Although this numerical study does not provide a complete picture of the computational cost incurred by the two methods in general, it is still relevant for providing useful insight into the aforementioned issue. Note that this assessment has been carried out with the two-dimensional Euler solver. Since the computational work associated with each of the two schemes in general is definitely problem dependent and also involves other parameters, such as the accuracy of boundary conditions, the time-marching method or the grid topology, a full assessment and comparison of the computational cost associated with each of the two methods would require a more elaborate study than the present one. However, such an assessment has not been carried out in this work as it is beyond the current work scope and it represents a significant effort in its own right.

Figure 5.44(a) shows the exact representation of the selected density variation, which is given by  $\rho(x, y) = 1.0 + 0.5 \cos(\pi \mathcal{J}(x)) \sin(5 \pi \mathcal{J}(x))$ , where  $\mathcal{J}(x)$  represents a linear mapping of the domain  $[-100, 100]$  into the domain  $[-1, 1]$ . The other parameters in the initial condition for this test case are  $v_x = 40$  m/s,  $v_y = 0$  m/s and  $p = 101.325$  kPa. The predicted density solution obtained after one full convection period for this problem using the 4th-order CENO scheme on a Cartesian grid with  $200 \times 20$  computational cells is shown in

Fig. 5.44(b). A visual comparison between the two density variations shown in Fig. 5.44 reveals that the cubic interpolant used by the high-order  $k$ -exact reconstruction captures all the inflection points of this smooth density variation very well and the solution is being recognized everywhere as smooth by the CENO approach. The error norms as well as the simulation CPU time and memory requirements corresponding to the limited 2nd-order and the 4th-order CENO schemes are presented in Table 5.2. Additionally, the same type of data pertaining to the unlimited 2nd-order variant is also shown in the table. Note that the data included in the table has been obtained by using the same CFL = 0.5 number for both second- and fourth-order Runge-Kutta methods used for the second- and fourth-order



(a) Exact solution

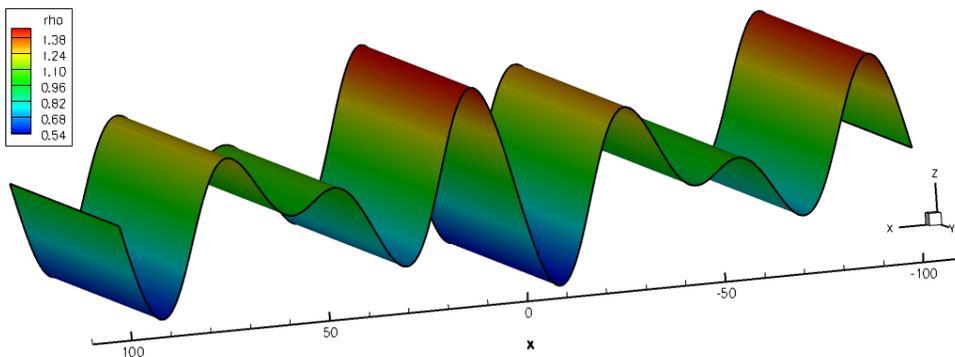
(b) 4th-order ( $k=3$ ) CENO prediction

Figure 5.44: (a) *Exact solution for the periodic linear advection of the density variation  $\rho(x, y) = 1.0 + 0.5 \cos(\pi \mathcal{J}(x)) \sin(5 \pi \mathcal{J}(x))$ ; and (b) Predicted density distribution for the advection of density variation shown in (a) after one full period using the 4th-order CENO scheme on a  $200 \times 20$  Cartesian grid.*

spatial discretizations, respectively. Based on the data shown in the table it can be seen that for the same mesh resolution the current 4th-order CENO scheme requires about 10 times more computational work than the 2nd-order method and about 1.5 times more memory. However, in order to obtain the same accuracy provided by the 4th-order CENO, a limited 2nd-order scheme would require about 20 times more computational cells (i.e., 80,000 instead of 4,000) which, in turn, necessitate approximately 42 times more computational work than the one of the high-order CENO. As a further benefit, this large saving in the number of computational cells is translated into a decrease in the memory requirements by a factor of about 6.5, which demonstrates the potential benefits of the high-order method. Furthermore, the high-order CENO method still remains about 25 times faster than a 2nd-order variant without any limiter applied, although it includes the extra computation associated with the calculation of the smoothness indicator. Note also that these savings are actually for a one-dimensional problem solved using the two-dimensional implementations of the two schemes. Therefore, the computational effort spent by the 4th-order CENO scheme to compute additional high-order derivatives related to the  $y$ -direction is not recovered as an increase in the accuracy of the scheme. Consequently, it is expected that for a *true* two-dimensional similar problem the computational savings offered by the high-order CENO would be even more significant. More importantly to large-scale applications is that, based on the same sort of rationale, it appears that three-dimensional simulations could tremendously benefit from the computational savings that the CENO high-order methods would seem to offer.

Table 5.2: *Error norms, simulation CPU time and memory requirements associated with the unlimited and limited 2nd-order, and the 4th-order CENO schemes obtained for the periodic linear convection of the density variation shown in Fig. 5.44.*

# Cells		$\mathcal{O}(\Delta x^2)$	$\mathcal{O}(\Delta x^2)$	$\mathcal{O}(\Delta x^4)$
		Unlimited	Limited	CENO
4,000 (200x20)	$L_1$ :	$1.82 \times 10^{-2}$	$2.74 \times 10^{-2}$	$1.96 \times 10^{-4}$
	$L_2$ :	$2.06 \times 10^{-2}$	$3.33 \times 10^{-2}$	$2.19 \times 10^{-4}$
	$L_\infty$ :	$3.62 \times 10^{-2}$	$7.37 \times 10^{-2}$	$3.46 \times 10^{-4}$
	$Time(h)$ :	0:01:45	0:01:48	<b>0:18:05</b>
	$Mem(kB)$ :	20,336	20,336	<b>31,232</b>
8,000 (400x20)	$L_1$ :	$4.56 \times 10^{-3}$	$9.60 \times 10^{-3}$	$1.20 \times 10^{-5}$
	$L_2$ :	$5.17 \times 10^{-3}$	$1.19 \times 10^{-2}$	$1.34 \times 10^{-5}$
	$L_\infty$ :	$9.08 \times 10^{-3}$	$2.98 \times 10^{-2}$	$2.12 \times 10^{-5}$
	$Time(h)$ :	0:07:01	0:07:42	<b>01:15:59</b>
	$Mem(kB)$ :	30,000	30,000	<b>50,816</b>
80,000 (4000x20)	$L_1$ :	$4.55 \times 10^{-5}$	$1.12 \times 10^{-4}$	-
	$L_2$ :	$5.16 \times 10^{-5}$	$2.25 \times 10^{-4}$	-
	$L_\infty$ :	$9.07 \times 10^{-5}$	$1.33 \times 10^{-3}$	-
	$Time(h)$ :	12:00:58	<b>12:27:14</b>	-
	$Mem(kB)$ :	203,680	<b>203,680</b>	-

# Chapter 6

## CONCLUSIONS AND RECOMMENDATIONS

*“I think and think for months, for years. Ninety-nine times the conclusion is false. The hundredth time I am right.”*

by Albert Einstein (1879 - 1955), German-born theoretical physicist

### 6.1 Summary and Conclusions

This thesis has presented and analyzed a new high-order finite-volume scheme with adaptive mesh refinement for solving hyperbolic and elliptic systems of conservation laws on body-fitted multi-block mesh. The proposed numerical procedure was applied to the advection-diffusion equation and to the Euler and the Navier-Stokes system of PDEs governing inviscid and laminar viscous flows of thermally and calorically perfect gases. A considerable amount of attention has been given to the verification and validation of each component of the high-order computational framework described in this dissertation. This has been accomplished by comparing predicted solutions to a variety of available analytical results, previously reported computations and experimental data. The ability of the scheme to accurately represent solutions with smooth extrema and yet robustly handle under-resolved and/or non-smooth solution content (i.e., solutions with shocks and discontinuities) is demonstrated. The analyses and the results included herein confirm that the proposed numerical algorithm has many of the desirable features of a large-scale simulation framework for inviscid and viscous flows. Consequently, it can be concluded that the thesis objectives outlined in Sect. 1.2 have been achieved and the current scheme has the potential to realize the overarching objective of the research, that is to reduce the computational cost of LES of turbulent combusting

flows although this shall remain a subject for future follow-on studies.

The original contributions of this thesis can be summarized as follows:

- **Development and evaluation of a new high-order CENO approach.**

The CENO scheme represents a hybrid approach which relies on a robust  $k$ -exact reconstruction to provide the high-order accuracy and on a monotonicity preserving reconstruction to avoid occurrence of  $\mathcal{O}(1)$  numerical oscillations in regions of large solution gradients and/or discontinuities. In particular, a limited piecewise linear interpolant is used herein to preserve monotonicity which is equivalent to reducing the order of the scheme in under-resolved solution regions. Although the global uniform accuracy of the high-order procedure is lost for flows with **non-smooth** solution content, a key advantage of the proposed scheme is that the solution reconstruction can be performed based only on a **fixed** central stencil, which in turn leads to many computational advantages. It allows the definition and use of a unique, symmetric, accurate and non-singular reconstruction stencil (the same stencil for every variable), the efficient extension of the algorithm to multiple dimensions and unstructured mesh, and, in turn, the reduction of the computational costs of the high-order procedure. Moreover, it has been shown that the challenges associated with boundary-condition prescription and parallel implementation arising from the enlarged stencils can be dealt with in an effective manner (see third contribution below). One possible downside of the current numerical method is that it requires two reconstructions in regions flagged as under-resolved, the second one being less expensive than the primary solution reconstruction. Consequently, the CENO scheme presents computational advantages primarily for flows mostly characterized by smooth and complex solution variations in which the increased accuracy stemmed from the high-order interpolant can be efficiently exploited. Finally, head-to-head comparisons between one-dimensional solutions obtained with the original ENO scheme based on the selection of the smoothest stencil and with the proposed CENO algorithm have demonstrated the similarities between the solution accuracy of the two schemes and confirmed the validity of the CENO approach to confer the ENO property to the numerical scheme.

- **Formulation of smoothness indicator for directing AMR.**

The cornerstone of the hybrid CENO approach is the solution smoothness indicator

which determines the switching between the two solution reconstructions. This thesis has directed a considerable body of research towards developing a reliable, simple and flexible indicator. Although a heuristic formulation, the mathematical expression of the smoothness indicator exploits basic polynomial characteristics with respect to the degree of smoothness in the neighbourhood of smooth and non-smooth regions and thus, it is able to identify high-order solution reconstructions prone to compromise the positivity of the numerical model. Additionally, the many numerical analyzes carried out in this work document extensively the properties of the smoothness indicator and its behaviour with varying solution resolution.

- **Development of high-order CENO scheme in conjunction with a parallel block-based AMR algorithm.**

A high-order finite-volume scheme with AMR has been proposed, verified and validated for prediction of inviscid and viscous flows governed by the advection-diffusion, the Euler and the Navier-Stokes equations on body-fitted multi-block quadrilateral mesh. The spatial discretization of the inviscid (hyperbolic) terms is based on the hybrid CENO solution reconstruction procedure and the high-order viscous (elliptic) fluxes are computed based on a  $k$ -order accurate average gradient derived from the same  $(k+1)$ -order accurate reconstruction used in the discretization of the inviscid term. This work has also proposed a unified computational framework for prescribing high-order boundary conditions for all the aforementioned PDEs on body-fitted multi-block quadrilateral mesh. Numerical studies with disparate solution length scales have indicated the high-order procedure in combination with AMR is capable of reducing the number of required computational elements for a given solution accuracy by orders of magnitude as compared to low-order counterparts. The usefulness of an  $h$ -refinement criterion based on the smoothness indicator of the hybrid scheme has also been evaluated and shown to provide a robust and reliable mesh adaptation algorithm that is capable of refining both under-resolved (in-accurate) and non-smooth regions of the solution and will not unnecessarily refine resolved solution content. Numerical experiments indicate that this algorithm has a great potential to reduce the overall computational cost of large-scale numerical simulations thereby reducing the grid resolution required to resolved accurately solutions. For large systems and multi-dimensional problems, the proposed

CENO scheme would seem to provide high-order accuracy at a lower computational cost as compared to other similar approaches.

## 6.2 Recommendations for Future Research

*“The important thing is not to stop questioning. Curiosity has its own reason for existing.”*

by Albert Einstein (1879 - 1955), German-born theoretical physicist

Although the high-order computational framework outlined in this dissertation provides by itself an accurate tool for studying inviscid and laminar viscous flows and represents a well documented methodology for extension to more complicated governing PDE sets, there are a number of areas in which additional capability could be added and the performance of the algorithm could be improved, which were deemed beyond the scope and time frame of this thesis. A brief indication of several directions for future research is given below.

- **Further investigation of the CENO approach.**

The heuristic form of the smoothness indicator lies at the core of the CENO formulation and its properties have been mostly determined numerically by extensive grid convergence studies. A formal analytical analysis of the solution smoothness indicator would provide mathematical proof and a complimentary validation of the properties induced by numerical experiments. Additionally, more compact ways for detecting non-smoothness are very desirable and few possibilities worth exploring have been suggested in Sect. 3.6.2. Furthermore, the application testing of the CENO approach should be extended to more complicated flows than the ones included herein. Within the context of the compressible Navier-Stokes equations, the scheme should be explored for a greater selection of more complex transonic and supersonic/hypersonic flow cases as well as for turbulent flows described with RANS, LES or DNS techniques. Moreover, the CENO method can be applied to other equation sets, such as those governing the turbulent reacting and non-reacting flows.

- **Extension of the CENO approach to other discretizations.**

The smoothness concepts employed in the definition of the CENO approach are generally valid and, as such, they could be applied to other spatial discretizations as well. One avenue of future research is a feasibility study regarding the formulation of a CENO approach for discontinuous Galerkin finite-element and/or residual distribution methods and the resulting potential benefits.

- **Exploration of other approaches to preserve monotonicity of the high-order procedure.**

The current CENO formulation uses a limited linear reconstruction as a mean to enforce solution monotonicity in those regions flagged as non-smooth. In addition to requiring an extra low-order reconstruction, this procedure has the disadvantage of being non-differentiable and consequently it may raise convergence issues for some steady-state solutions. However, a more efficient and differentiable method may be formulated by employing/developing a different way in which the monotonicity is conferred to the scheme. One possibility to avoiding the limited linear reconstruction would be to apply a differentiable limiter to the high-order reconstruction as described in the recent work by Michalak and Ollivier-Gooch [32].

- **Application of CENO discretization in conjunction with effective time integration algorithms.**

The current CENO framework uses standard explicit time-marching schemes which have not been optimized for the high-order method. Several ways to improve the efficiency of the time integration procedure is to consider the use of  $h$ - $p$ -multigrid [174] and implicit algorithms [31, 175]. Additionally, the convergence of non-linear iterative methods for discontinuous solution problems could be enhanced by adding dissipation terms to the numerical flux functions of the non-smooth elements as done in [156].

- **Further improvement of the multi-block communication and the algorithm parallelization.**

Sufficient layers of ghost cells have been used in this work as a way to share information among the grid blocks and to allow the execution of a completely independent high-order solution procedure on each block. This approach is definitely easier to program

than other methods but may not be the most efficient one for the implementation of the high-order CENO. In particular, as several steps of the solution procedure such as the computation of the smoothness indicator require only few pieces of information with respect to the solution variation in the adjacent cells it may be more efficient to exchange only this reduced data set among the neighbour blocks than to duplicate information and carry out high-order reconstructions in overlapping ghost cells. However, it is noted that the complexity of such an implementation would increase considerably, especially for a framework using adaptive mesh refinement, and its efficiency may be very dependent on the particular size of the blocks.

- **Extension to 3D problems and to hybrid structured/unstructured meshes.**

It is definitely worth exploring the computational benefits of an adaptive CENO algorithm for three-dimensional problems and for unstructured meshes. A starting point in this direction is represented by the extension of CENO method to three-dimensional fixed Cartesian multi-block mesh carried out in [176].

- **Development of  $h$ - $p$ -adaptation.**

The adaptive high-order algorithm in this work was based solely on an  $h$ -refinement procedure while maintaining the same polynomial order throughout the computational domain. This strategy is very effective in improving the local accuracy especially near solution discontinuities, but may not efficiently use the degrees of freedom of the numerical scheme. One extreme example of inefficiency is the representation with high-order polynomials of regions of “flat” solution variation. Adding an  $h$ - $p$ -adaptive capability at the block level should allow for better degree of freedom management and optimization in the adaptation process. In smooth flow regions, refinement could be mostly done with  $p$  and near discontinuities with  $h$ , a technique which typically allows for exponential convergence rates.

Considering this brief summary of open research directions for only one of the existing high-order schemes, it should already give an impression of how much is left to be explored in this field. On the bright side, it is noted that the last decade has seen an increased interest in developing high-order methods following the recognition of the potential benefits these procedures can provide to complex flow simulations especially. It is speculated that

the mathematical apparatus of many high-order schemes will soon achieve maturity and that it is becoming more imperative to perform the head-to-head comparison of the existing formulations and to clearly recognize the trade-offs in each case. Although most likely future algorithms will find more elegant solutions to the current issues of high-order methods, it is anticipated they will inevitably face their own challenges because, as Charles Kettering said:

*“There will always be a Frontier where there is an open mind and a willing hand.”*

# REFERENCES

- [1] Thom, A., “The flow past circular cylinder at low speeds, procedures,” *Proceedings of the Royal Society of London A*, Vol. 141, 1933, pp. 651–666.
- [2] Longwell, H. L., “The Future of the Oil and Gas Industry: Past Approaches, New Challenges,” *World Energy*, Vol. 5, No. 3, 2002, pp. 100–104.
- [3] Mavriplis, D. J., “Unstructured Mesh Discretizations and Solvers for Computational Aerodynamics,” Paper 2007-3955, AIAA, June 2007.
- [4] Boudier, G., Gicquel, L. Y. M., and Poinso, T. J., “Effects of mesh resolution on large eddy simulation of reacting flows in complex geometry combustors,” *Combustion and Flame*, Vol. 155, 2008, pp. 196–214.
- [5] Tanahashi, M., Fajimura, M., and Miyauchi, T., “Coherent Fine-Scale Eddies in Turbulent Premixed Flames,” *Proceedings of Combustion Institute*, Vol. 28, 2000, pp. 529–535.
- [6] Tanahashi, M., Nada, Y., Ito, Y., and Miyauchi, T., “Local Flame Structure in the Well-Stirred Reactor Regime,” *Proceedings of Combustion Institute*, Vol. 2, 2002, pp. 2041–2049.
- [7] Bell, J., Day, M., and Grcar, J., “Numerical Simulation of Premixed Turbulent Methane Combustion,” *Proceedings of Combustion Institute*, Vol. 29, 2002, pp. 1987–1993.
- [8] Gullbrand, J., “Explicit filtering and subgrid-scale models in turbulent channel flow,” *Annual Research Briefs*, Center for Turbulence Research, 2001, pp. 31–42.

- [9] Georges, L., Winckelmans, G., and Geuzaine, P., “Improving shock-free compressible RANS solvers for LES on unstructured meshes,” *Journal of Computational and Applied Mathematics*, Vol. 215, 2008, pp. 419–428.
- [10] Selle, L., Lartigue, G., Poinso, T., Koch, R., Schildmacher, K., Krebs, W., Prade, B., Kaufmann, P., and Veynante, D., “Compressible large eddy simulation of turbulent combustion in complex geometry on unstructured meshes,” *Combustion and Flame*, Vol. 137, 2004, pp. 489–505.
- [11] Moin, P. and Mahesh, K., “Direct numerical simulation: a tool in turbulence research,” *Annual Review of Fluid Mechanics*, Vol. 30, 1998, pp. 539–578.
- [12] Jameson, L., “AMR vs. High Order Schemes,” *Journal of Scientific Computing*, Vol. 18, 2003, pp. 1–24.
- [13] Lax, P. D. and Richtmyer, R. D., “Survey of the stability of linear finite difference equations,” *Communications on Pure and Applied Mathematics*, Vol. 9, 1956, pp. 267–293.
- [14] Lomax, H., Pulliam, T., and Zingg, D., *Fundamentals of Computational Fluid Dynamics*, Springer-Verlag, New York, 2001.
- [15] Hirsch, C., *Numerical Computation of Internal and External Flows, Volume 2, Computational Methods for Inviscid and Viscous Flows*, John Wiley & Sons, Toronto, 1990.
- [16] Barth, T. J., “Recent Developments in High Order K-Exact Reconstruction on Unstructured Meshes,” Paper 93-0668, AIAA, January 1993.
- [17] Runge, C., “Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten,” *Zeitschrift für Mathematik und Physik*, Vol. 46, 1901, pp. 224–243.
- [18] Harten, A., Enquist, B., Osher, S., and Chakravarthy, S. R., “Uniformly High Order Accurate Essentially Non-Oscillatory Schemes, III,” *Journal of Computational Physics*, Vol. 71, 1987, pp. 231–303.
- [19] Harten, A. and Chakravarthy, S. R., “Multi-dimensional ENO schemes for general geometries,” Report 91-76, ICASE, September 1991.

- [20] Abgrall, R., “On Essentially Non-Oscillatory Schemes on Unstructured Meshes: Analysis and Implementation,” *Journal of Computational Physics*, Vol. 114, 1994, pp. 45–58.
- [21] Sonar, T., “On the construction of essentially non-oscillatory finite volume approximations to hyperbolic conservation laws on general triangulations: polynomial recovery, accuracy and stencil selection,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 140, 1997, pp. 157.
- [22] Jiang, G.-S. and Shu, C.-W., “Efficient Implementation of Weighted ENO Schemes,” *Journal of Computational Physics*, Vol. 126, 1996, pp. 202–228.
- [23] Friedrich, O., “Weighted Essentially Non-Oscillatory Schemes for the Interpolation of Mean Values on Unstructured Grids,” *Journal of Computational Physics*, Vol. 144, 1998, pp. 194–212.
- [24] Hu, C. and Shu, C.-W., “Weighted Essentially Non-Oscillatory Schemes on Triangular Meshes,” *Journal of Computational Physics*, Vol. 150, 1999, pp. 97–127.
- [25] Stanescu, D. and Habashi, W., “Essentially Nonoscillatory Euler solutions on unstructured meshes using extrapolation,” *AIAA Journal*, Vol. 36, 1998, pp. 1413–1416.
- [26] Haselbacher, A., “A WENO Reconstruction Algorithm for Unstructured Grids Based on Explicit Stencil Construction,” Paper 2005-0879, AIAA, January 2005.
- [27] Wolf, W. R. and Azevedo, J., “High-Order Unstructured Essentially Nonoscillatory and Weighted Essentially Nonoscillatory Schemes for Aerodynamic Flows,” *AIAAJ*, Vol. 44, No. 10, October 2006, pp. 2295–2310.
- [28] Barth, T. J. and Fredrickson, P. O., “Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction,” Paper 90-0013, AIAA, January 1990.
- [29] Ollivier-Gooch, C. F., “High-Order ENO Schemes for Unstructured Meshes Based on Least-Squares Reconstruction,” Paper 97-0540, AIAA, January 1997.
- [30] Ollivier-Gooch, C. F., “Quasi-ENO Schemes for Unstructured Meshes Based on Unlimited Data-Dependent Least-Squares Reconstruction,” *Journal of Computational Physics*, Vol. 133, 1997, pp. 6–17.

- [31] Nejat, A. and Ollivier-Gooch, C. F., “A high-order accurate unstructured finite volume Newton-Krylov algorithm for inviscid compressible flows,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 2582–2609.
- [32] Michalak, K. and Ollivier-Gooch, C. F., “Limiters for Unstructured Higher-Order Accurate Solutions of the Euler Equations,” Paper 2008-776, AIAA, January 2008.
- [33] Capdeville, G., “Towards a compact high-order method for non-linear hyperbolic systems. I: The Hermite Least-Square Monotone (HLSM) reconstruction,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 3762–3788.
- [34] Colella, P., Dorr, M. R., Hittinger, J. A. F., and Martin, D. F., “High-Order Finite-Volume Methods in Mapped Coordinates,” submitted to the *Journal of Computational Physics*, 2010.
- [35] McCorquodale, P. and Colella, P., “A high-order finite-volume method for hyperbolic conservation laws on locally-refined grids,” submitted to the *Communications in Applied Mathematics and Computational Science*, June 2010.
- [36] Colella, P. and Sekora, M. D., “A limiter for PPM that preserves accuracy at smooth extrema,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 7069–7076.
- [37] Pirozzoli, S., “On the spectral properties of shock-capturing schemes,” *Journal of Computational Physics*, Vol. 219, 2006, pp. 489–497.
- [38] Pirozzoli, S., “High-order shock-capturing schemes: ENO, WENO, and beyond,” Private communication, *High Order Non-Oscillatory Methods for Wave Propagation: Algorithms and Applications III*, March 2009.
- [39] Cockburn, B. and Shu, C.-W., “Nonlinearly stable compact schemes for shock calculations,” *SIAM Journal on Numerical Analysis*, Vol. 31, 1994, pp. 607–627.
- [40] Cockburn, B. and Shu, C.-W., “TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite-Element Method for Conservation Laws II: General Framework,” *Mathematics of Computation*, Vol. 52, 1989, pp. 411.

- [41] Cockburn, B., Lin, S.-Y., and Shu, C.-W., “TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite-Element Method for Conservation Laws III: One-Dimensional Systems,” *Journal of Computational Physics*, Vol. 84, 1989, pp. 90.
- [42] Cockburn, B., Hou, S., and Shu, C.-W., “TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite-Element Method for Conservation Laws IV: The Multidimensional Case,” *Journal of Computational Physics*, Vol. 54, 1990, pp. 545.
- [43] Qui, J. and Shu, C.-W., “Runge-Kutta Discontinuous Galerkin Method Using WENO Limiters,” *SIAM Journal on Scientific Computing*, Vol. 26, No. 3, 2005, pp. 907–929.
- [44] Barter, G. E. and Darmofal, D. L., “Shock Capturing with Higher-Order, PDE-Based Artificial Viscosity,” Paper 2007-3823, AIAA, June 2007.
- [45] Xu, Z., Liu, Y., and Shu, C.-W., “Hierarchical reconstruction for discontinuous Galerkin methods on unstructured grids with a WENO-type linear reconstruction and partial neighboring cells,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 2194–2212.
- [46] Kubatko, E. J., Dawson, C., and Westerink, J. J., “Time step restrictions for RungeKutta discontinuous Galerkin methods on triangular grids,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 9697–9710.
- [47] Dumbser, M., Balsara, D. S., Toro, E. F., and Munz, C.-D., “A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 8209–8253.
- [48] Wang, Z. J., “Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids,” *Journal of Computational Physics*, Vol. 178, 2002, pp. 210–251.
- [49] Wang, Z. J., Zhang, L., and Liu, Y., “High-Order Spectral Volume Method for 2D Euler Equations,” Paper 2003–3534, AIAA, June 2003.
- [50] Liu, Y., Vinokur, M., and Wang, Z., “Discontinuous spectral difference method for conservation laws on unstructured grids,” *Proceedings of the 3rd International Conference on Computational Fluid Dynamics, Toronto, Canada, July 12–16, 2004*.

- [51] Huynh, H., “A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods,” Paper 2007-4079, AIAA, June 2007.
- [52] Yang, M. and Wang, Z. J., “A Parameter-Free Generalized Moment Limiter for High-Order Methods on Unstructured Grids,” Tech. Rep. 2009-605, AIAA, 2009.
- [53] Yang, M. and Wang, Z. J., “A Parameter-Free Generalized Moment Limiter for High-Order Methods on Unstructured Grids,” *Advances in Applied Mathematics and Mechanics*, Vol. 1, No. 4, 2009, pp. 451–480.
- [54] Wang, Z. J. and Gao, H., “A Unifying Lifting Collocation Penalty Formulation for the Euler Equations on Mixed Grids,” Paper 2009-401, AIAA, 2009.
- [55] Wang, Z. J. and Gao, H., “A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 8161–8186.
- [56] Abgrall, R., “Toward the Ultimate Conservative Scheme: Following the Quest,” *Journal of Computational Physics*, Vol. 167, 2001, pp. 277–315.
- [57] Csík, A., Ricchiuto, M., and Deconinck, H., “A Conservative Formulation of the Multidimensional Upwind Residual Distribution Schemes for General Nonlinear Conservation Laws,” *Journal of Computational Physics*, Vol. 179, 2002, pp. 286–312.
- [58] Abgrall, R. and Roe, P. L., “High Order Fluctuation Schemes on Triangular Mesh,” *Journal of Computational Physics*, Vol. 19, No. 1–3, 2003, pp. 3–36.
- [59] Abgrall, R., “Essentially non-oscillatory residual distribution schemes for hyperbolic problems,” *Journal of Computational Physics*, Vol. 214, 2006, pp. 773–808.
- [60] Guzik, S. and Groth, C. P. T., “Comparison of Solution Accuracy of Multidimensional Residual Distribution and Godunov-Type Finite-Volume Methods,” *International Journal of Computational Fluid Dynamics*, Vol. 22, No. 1, 2008, pp. 61–68.
- [61] Guzik, S., *Accurate Residual-Distribution Schemes for Accelerated Parallel Architectures*, Ph.D. thesis, University of Toronto Institute for Aerospace Studies, February 2010.

- [62] Dias, S. C. and Zingg, D. W., “A High-Order Parallel Newton-Krylov flow solver for the Euler Equations,” Paper 2009-3657, AIAA, June 2009.
- [63] Nordström, J. and Carpenter, M. H., “Boundary and interface conditions for high-order finite-difference methods applied to the Euler and Navier-Stokes equations,” *Journal of Computational Physics*, Vol. 148, No. 2, 1999, pp. 621–645.
- [64] Coirier, W. J., *An Adaptively-Refined, Cartesian, Cell-Based Scheme for the Euler and Navier-Stokes Equations*, Ph.D. thesis, University of Michigan, 1994.
- [65] Delanaye, M., Aftomis, M. J., Berger, M. J., Liu, Y., and Pulliam, T. H., “Automatic Hybrid-Cartesian Grid Generation for High-Reynolds Number Flows Around Complex Geometries,” Paper 99-0777, AIAA, January 1999.
- [66] Sun, Y., Wang, Z. J., and Liu, Y., “Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow,” *Journal of Computational Physics*, Vol. 215, No. 1, 2006, pp. 41–58.
- [67] May, G. and Jameson, A., “High-Order Accurate Methods for High-Speed Flow,” Paper 2005-5251, AIAA, 2005.
- [68] May, G. and Jameson, A., “A Spectral Difference Method for the Euler and Navier-Stokes Equations on Unstructured Meshes,” Paper 2006-304, AIAA, January 2006.
- [69] Gao, H. and Wang, Z. J., “A High-Order Lifting Collocation Penalty Formulation for the Navier-Stokes Equations on 2-D Mixed Grids,” Paper 2009-3784, AIAA, 2009.
- [70] Haga, T., Gao, H., and Wang, Z. J., “A High-Order Unifying Discontinuous Formulation for 3D Mixed Grids,” Paper 2010-540, AIAA, January 2010.
- [71] Barad, M. and Colella, P., “A fourth-order accurate local refinement method for Poisson’s equation,” *Journal of Computational Physics*, Vol. 209, 2005, pp. 1–18.
- [72] Ollivier-Gooch, C. F. and Van Altena, M., “A High-order Accurate Unstructured Mesh Finite-Volume Scheme for the Advection-Diffusion Equation,” *Journal of Computational Physics*, Vol. 181, No. 2, 2002, pp. 729–752.

- [73] van Leer, B., Lo, M., and van Raalte, M., “A discontinuous Galerkin method for diffusion based on recovery,” Paper 2007-4083, AIAA, June 2007.
- [74] van Raalte, M. and van Leer, B., “Bilinear Forms for the Recovery-Based Discontinuous Galerkin Method for Diffusion,” *Communications in Computational Physics*, Vol. 5, No. 2–4, 2009, pp. 683–693.
- [75] Oliver, T. A. and Darmofal, D. L., “An Unsteady Adaptation Algorithm for Discontinuous Galerkin Discretizations of the RANS Equations,” Paper 2007-3940, AIAA, June 2007.
- [76] De Rango, S. and Zingg, D., “Higher-order spatial discretization for turbulent aerodynamic computations,” *AIAAJ*, Vol. 39, 2001, pp. 1296–1304.
- [77] Gui, W. and Babuška, I., “The  $h$ ,  $p$  and  $h$ - $p$  Versions of the Finite Element Method in 1 Dimension,” *Numerische Mathematik*, Vol. 49, 1986, pp. 577–612.
- [78] Farhat, C., Degand, C., Koobus, B., and Lesoinne, M., “Torsional Springs for Two-Dimensional Dynamic Unstructured Fluid Meshes,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 163, 1998, pp. 231–245.
- [79] Bar-Yoseph, P. Z., Mereu, S., Chippada, S., and Kalro, V. K., “Automatic Monitoring of Element Shape Quality in 2-D and 3-D Computational Mesh Dynamics,” *Computational Mechanics*, Vol. 27, No. 5, 2001, pp. 378–395.
- [80] Stein, K., Tezduyar, T., and Benney, R., “Mesh Moving Techniques for Fluid-Structure Interactions with Large Displacements,” *Journal of Applied Mechanics*, Vol. 70, No. 1, 2003, pp. 58–63.
- [81] Habashi, W. G., Dompierre, J., Bourgault, Y., Ait-Ali-Yahia, D., Fortin, M., and Vallet, M. G., “Anisotropic Mesh Adaption: Towards User-Independent, Mesh-Independent and Solver-Independent CFD. Part I: General Principles,” *International Journal for Numerical Methods in Fluids*, Vol. 32, No. 6, 2000, pp. 725–744.
- [82] Béal, P., Koko, J., and Touzani, R., “Mesh  $r$ -Adaptation for Unilateral Contact Problems,” *International Journal of Applied Mathematics and Computer Science*, Vol. 12, No. 1, 2002, pp. 9–16.

- [83] Truong, A., Oldfield, C., and Zingg, D., “Mesh Movement for a Discrete-Adjoint Newton-Krylov Algorithm for Aerodynamic Optimization,” *AIAA Journal*, Vol. 46, No. 7, 2008, pp. 1695–1704.
- [84] Feng, H. and Mavriplis, C., “Adaptive Spectral Element Simulations of Thin Premixed Flame Sheet Deformations,” *Journal of Scientific Computing*, Vol. 17, No. 1-4, 2002, pp. 385–395.
- [85] Demkowics, L., Rachowicz, W., and Devloo, P., “A Fully Automatic *hp*-Adaptivity,” *Journal of Scientific Computing*, Vol. 17, No. 1-4, 2002, pp. 117–142.
- [86] Berger, M. J. and Olinger, J., “Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations,” *Journal of Computational Physics*, Vol. 53, 1984, pp. 484–512.
- [87] Berger, M. J. and Colella, P., “Local Adaptive Mesh Refinement for Shock Hydrodynamics,” *Journal of Computational Physics*, Vol. 82, 1989, pp. 67–84.
- [88] Berger, M. J. and LeVeque, R. J., “An Adaptive Cartesian Mesh Algorithm for the Euler Equations in Arbitrary Geometries,” Paper 89-1930, AIAA, June 1989.
- [89] Coirier, W. J. and Powell, K. G., “Solution-Adaptive Cartesian Cell Approach for Viscous and Inviscid Flows,” *AIAA Journal*, Vol. 34, No. 5, May 1996, pp. 938–945.
- [90] De Zeeuw, D. and Powell, K. G., “An Adaptively Refined Cartesian Mesh Solver for the Euler Equations,” *Journal of Computational Physics*, Vol. 104, 1993, pp. 56–68.
- [91] De Zeeuw, D. L., Groth, C. P. T., Marshall, H. G., Gombosi, T. I., Powell, K. G., and Stout, Q. F., “A Parallel Adaptive High-Resolution Scheme for MHD with Applications in Space Plasma Physics,” submitted to the *SIAM Journal on Scientific Computing*, September 1999.
- [92] Gao, X. and Groth, C. P. T., “A parallel adaptive mesh refinement algorithm for predicting turbulent non-premixed combustng flows,” *International Journal of Computational Fluid Dynamics*, Vol. 20, No. 5, 2006, pp. 349–357.
- [93] Groth, C. P. T., De Zeeuw, D. L., Gombosi, T. I., and Powell, K. G., “A Parallel Adaptive 3D MHD Scheme for Modeling Coronal and Solar Wind Plasma Flows,” *Space Science Reviews*, Vol. 87, 1999, pp. 193–198.

- [94] Groth, C. P. T. and Northrup, S. A., “Parallel Implicit Adaptive Mesh Refinement Scheme for Body-Fitted Multi-Block Mesh,” Paper 2005-5333, AIAA, June 2005.
- [95] Henshaw, W. D. and Schwendeman, D. W., “An adaptive numerical scheme for high-speed reacting flow on overlapping grids,” *Journal of Computational Physics*, Vol. 191, 2003, pp. 420–447.
- [96] Northrup, S. A. and Groth, C. P. T., “Solution of Laminar Diffusion Flames Using a Parallel Adaptive Mesh Refinement Algorithm,” Paper 2005-0547, AIAA, January 2005.
- [97] Powell, K. G., Roe, P. L., and Quirk, J., “Adaptive-Mesh Algorithms for Computational Fluid Dynamics,” *Algorithmic Trends in Computational Fluid Dynamics*, edited by M. Y. Hussaini, A. Kumar, and M. D. Salas, Springer-Verlag, New York, 1993, pp. 303–337.
- [98] Sachdev, J. S., Groth, C. P. T., and Gottlieb, J. J., “A Parallel Solution-Adaptive Scheme for Predicting Multi-Phase Core Flows in Solid Propellant Rocket Motors,” *International Journal of Computational Fluid Dynamics*, Vol. 19, No. 2, 2005.
- [99] Bell, J., Day, M., Almgren, A., Lijewski, M. J., and Rendleman, C. A., “Adaptive numerical simulation of turbulent premixed combustion,” *Proceedings of the First MIT Conference on Computational Fluid and Solid Mechanics*, June 2001.
- [100] Gao, X., *A Parallel Solution-Adaptive Method for Turbulent Non-Premixed Combusting Flows*, Ph.D. thesis, University of Toronto, 2008.
- [101] Dervieux, A., Leservoisier, D., George, P.-L., and Coudiere, Y., “About theoretical and practical impact of mesh adaptation on approximation of functions and PDE solutions,” *International Journal for Numerical Methods in Fluids*, Vol. 43, 2003, pp. 507–516.
- [102] Kamkar, S. J., Jameson, A., Wissink, A. M., and Sankaran, V., “Feature-Driven Cartesian Adaptive Mesh Refinement in the Helios Code,” Paper 2010–171, AIAA, January 2010.

- [103] Moore, P. K., “An implicit interpolation error-based error estimation strategy for  $hp$ -adaptivity in one space dimension,” *Journal of Numerical Mathematics*, Vol. 12, No. 2, 2004, pp. 143–167.
- [104] Cao, W., “An Interpolation Error Estimate in  $\mathcal{R}^2$  Based on the Anisotropic Measures of Higher Order Derivatives,” *Mathematics of Computation*, Vol. 77, No. 261, 2008, pp. 265–286.
- [105] Verfürth, R., “*A posteriori* error estimation and adaptive mesh-refinement techniques,” *Journal of Computational and Applied Mathematics*, Vol. 50, 1994, pp. 67–83.
- [106] Machiels, L., Peraire, J., and Patera, A. T., “A Posteriori Finite-Element Output Bounds for the Incompressible NavierStokes Equations: Application to a Natural Convection Problem,” *Journal of Computational Physics*, Vol. 91, No. A7, 1986, pp. 8057–8064.
- [107] Venditti, D. A. and Darmofal, D. L., “Anisotropic Adaptation for Functional Outputs of Viscous Flow Simulations,” Paper 2003-3845, AIAA, June 2003.
- [108] Venditti, D. A. and Darmofal, D. L., “Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows,” *Journal of Computational Physics*, Vol. 187, 2003, pp. 22–46.
- [109] Nemec, M. and Aftosmis, M. J., “Adjoint Error Estimation and Adaptive Refinement for Embedded-Boundary Cartesian Meshes,” Paper 2007-4187, AIAA, June 2007.
- [110] Nemec, M., Aftosmis, M. J., and Wintzer, M., “Adjoint-Based Adaptive Mesh Refinement for Complex Geometries,” Paper 2008-0725, AIAA, January 2008.
- [111] Rumpfkeil, M. P. and Zingg, D. W., “A hybrid algorithm for far-field noise minimization,” *Computers & Fluids*, Vol. 39, No. 9, 2010, pp. 1516–1528.
- [112] Gao, X. and Groth, C. P. T., “A parallel solution-adaptive method for three-dimensional turbulent non-premixed combustions flows,” *Journal of Computational Physics*, Vol. 229, 2010, pp. 3250–3275.

- [113] McDonald, J. and Groth, C. P. T., “Numerical Modeling of Micron-Scale Flows Using the Gaussian Moment Closure,” Paper 2005-5035, AIAA, June 2005.
- [114] Roache, P. J., *Verification and Validation in Computational Science and Engineering*, Hermosa Publisher, New Mexico, 1998.
- [115] Toro, E. F., *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer-Verlag, New York, 1999.
- [116] Petrovsky, I. G., *Lectures on Partial Differential Equations*, Interscience, New York, 1954.
- [117] Bers, L., John, F., and Schechter, M., *Partial Differential Equations*, Interscience, New York, 1966.
- [118] Munson, B. R., Young, D. F., and Okiishi, T. H., *Fundamentals of Fluid Mechanics*, John Wiley & Sons, New York, 2002.
- [119] Anderson, J. D., *Computational Fluid Dynamics: The Basics with Applications*, McGraw Hill, New York, 1995.
- [120] Schlichting, H. and Gersten, K., *Boundary-Layer Theory*, Springer, New York, 8th ed., 2000.
- [121] Temam, R., *Navier-Stokes Equations*, North-Holland, New York, 1977.
- [122] Doering, C. R., “The 3D Navier-Stokes Problem,” *Annual Review of Fluid Mechanics*, Vol. 41, 2009, pp. 109–128.
- [123] LeVeque, R. J., *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, Cambridge, 2002.
- [124] Hirsch, C., *Numerical Computation of Internal and External Flows, Volume 1, Fundamentals of Numerical Discretization*, John Wiley & Sons, Toronto, 1989.
- [125] Kleb, W. L., Nielsen, E. J., Gnoffo, P. A., Parky, M. A., and Wood, W. A., “Collaborative Software Development in Support of Fast Adaptive AeroSpace Tools (FAAST),” Paper 2003-3978, AIAA, 2003.

- [126] “Fully Unstructured Navier-Stokes (FUN3D),” Retrieved from <http://fun3d.larc.nasa.gov/> (Accessed 4 July 2010).
- [127] ANSYS, Inc., “ANSYS FLUENT Flow Modeling Software,” Retrieved from <http://www.ansys.com/products/fluid-dynamics/fluent/> (Accessed 4 July 2010).
- [128] van Leer, B., Tai, C. H., and Powell, K. G., “Design of Optimally-Smoothing Multi-Stage Schemes for the Euler Equations,” Paper 89-1933-CP, AIAA, June 1989.
- [129] van Leer, B., Lee, W.-T., Roe, P. L., Powell, K. G., and Tai, C.-H., “Design of optimally smoothing multistage schemes for the Euler equations,” *Communications in Applied Numerical Methods*, Vol. 8, No. 10, 1992, pp. 761–769.
- [130] Godunov, S. K., “Finite-Difference Method for Numerical Computations of Discontinuous Solutions of the Equations of Fluid Dynamics,” *Matematicheskii Sbornik*, Vol. 47, 1959, pp. 271–306.
- [131] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical Recipes. The Art of Scientific Computing 3rd-Edition*, Cambridge University Press, New York, 2007.
- [132] Shi, J., Hu, C., and Shu, C., “A technique of treating negative weights in WENO schemes,” *Journal of Computational Physics*, Vol. 175, 2002, pp. 108–127.
- [133] Dumbser, M. and Käser, M., “Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems,” *Journal of Computational Physics*, Vol. 221, 2007, pp. 693–723.
- [134] Capdeville, G., “A central WENO scheme for solving hyperbolic conservation laws on non-uniform meshes,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 2977–3014.
- [135] Harten, A., Enquist, B., Osher, S., and Chakravarthy, S. R., Report 86-22, ICASE, April 1986.
- [136] Harten, A., “High Resolution Schemes for Hyperbolic Conservation Laws,” *Journal of Computational Physics*, Vol. 49, 1983, pp. 357–393.

- [137] Barth, T. J., “Numerical Methods for Gasdynamics Systems on Unstructured Meshes,” *Lecture Notes in Computational Science and Engineering*, Vol. 5, Springer-Verlag, Heidelberg, 1998, pp. 195–285.
- [138] Mavriplis, D. J., “Revisiting the Least-Squares Procedure for Gradient Reconstruction on Unstructured Meshes,” Paper 2003-3986, AIAA, June 2003.
- [139] Lawson, C. and Hanson, R., *Solving least squares problems*, Prentice-Hall, INC, 1974.
- [140] Penrose, R., “A generalized inverse for matrices,” *Proceedings of the Cambridge Philosophical Society*, Vol. 51, 1955, pp. 406–413.
- [141] Neumaier, A., “Solving Ill-Conditioned and Singular Linear Systems: A Tutorial on Regularization,” *SIAM Review*, Vol. 40, No. 3, 1998, pp. 636–666.
- [142] van Leer, B., “Towards the Ultimate Conservative Difference Scheme. III. Upstream-Centered Finite-Difference Schemes for Ideal Compressible Flow,” *Journal of Computational Physics*, Vol. 23, 1977, pp. 263–275.
- [143] van Leer, B., “Towards the Ultimate Conservative Difference Scheme. IV. A New Approach to Numerical Convection,” *Journal of Computational Physics*, Vol. 23, 1977, pp. 276–299.
- [144] Sweby, P. K., “High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws,” *SIAM Journal on Numerical Analysis*, Vol. 21, 1984, pp. 995–1011.
- [145] Berger, M. J., Aftosmis, M. J., and Murman, S. M., “Analysis of Slope Limiters on Irregular Grids,” Paper 2005-0490, AIAA, January 2005.
- [146] Shu, C.-W., “TVB Uniformly High-Order Schemes for Conservation Laws,” *Mathematics of Computation*, Vol. 49, No. 179, 1987, pp. 105–121.
- [147] Venkatakrishnan, V., “On the Accuracy of Limiters and Convergence to Steady State Solutions,” Paper 93-0880, AIAA, January 1993.
- [148] Gustafsson, B., “The Convergence Rate for Difference Approximations to Mixed Initial Boundary Value Problems,” *Mathematics of Computation*, Vol. 29, No. 130, 1975, pp. 396–406.

- [149] Balakrishnan, N. and Fernandez, G., “Wall Boundary Conditions for Inviscid Compressible Flows on Unstructured Meshes,” *International Journal for Numerical Methods in Fluids*, , No. 28, 1998, pp. 1481–1501.
- [150] Gottlieb, J. J. and Groth, C. P. T., “Assessment of Riemann Solvers for Unsteady One-Dimensional Inviscid Flows of Perfect Gases,” *Journal of Computational Physics*, Vol. 78, 1988, pp. 437–458.
- [151] Roe, P. L., “Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes,” *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.
- [152] Einfeldt, B., “On Godunov-Type Methods for Gas Dynamics,” *SIAM Journal on Numerical Analysis*, Vol. 25, 1988, pp. 294–318.
- [153] Linde, T. J., *A Three-Dimensional Adaptive Multifluid MHD Model of the Heliosphere*, Ph.D. thesis, University of Michigan, May 1998.
- [154] Linde, T., “A practical, general-purpose, two-state HLL Riemann solver for hyperbolic conservation laws,” *International Journal for Numerical Methods in Fluids*, Vol. 40, 2002, pp. 391–402.
- [155] Gao, X., Northrup, S., and Groth, C. P. T., “Parallel solution-adaptive method for two-dimensional non-premixed combusting flows,” in press in *Progress in Computational Fluid Dynamics*, July 2010.
- [156] Barth, T. J. and Larson, M. G., “A-Posteriori Error Estimation for Higher Order Godunov Finite Volume Methods on Unstructured Meshes,” *Finite Volumes for Complex Applications III*, edited by R. Herbin and D. Kroner, HERMES Science Publishing, London, 2002, pp. 41–63.
- [157] Stroustrup, B., *C++ Programming Language, The 3rd Edition*, Addison-Wesley Professional, 1997.
- [158] Gropp, W., Lusk, E., and Skjellum, A., *Using MPI*, MIT Press, Cambridge, Massachusetts, 1999.

- [159] AIAA, “Guide for the Verification and Validation of Computational Fluid Dynamics Simulations,” AIAA G-077-1998, 1998.
- [160] Zingg, D. W., “Grid Studies for Thin-Layer Navier-Stokes Computations of Airfoil Flowfield,” *AIAA Journal*, Vol. 30, No. 10, October 1992, pp. 2561–2564.
- [161] Kirby, R. M. and Silva, C., “The Need For Verifiable Visualization,” *IEEE Computer Graphics and Applications*, Vol. 28, No. 5, 2008, pp. 78–83.
- [162] Meyer, M., Nelson, B., Kirby, R. M., and Whitaker, R., “Particle Systems for Efficient and Accurate Finite Element Visualization,” *IEEE Transactions on Visualization and Computer Graphics*, Vol. 13, No. 5, 2007, pp. 1015–1026.
- [163] Thompson, J. F., Warsi, Z. U. A., and Mastin, C. W., *Numerical Grid Generation—Foundations and Applications*, North-Holland, New York, 1985.
- [164] AGARD Subcommittee C, “Test Cases for Inviscid Flow Field Methods,” AGARD Advisory Report 211, 1986.
- [165] Abgrall, R. and Harten, A., “Multiresolution Representation in Unstructured Meshes,” *SIAM Journal on Numerical Analysis*, Vol. 35, No. 6, 1998, pp. 2128–2146.
- [166] Woodward, P. and Colella, P., “The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks,” *Journal of Computational Physics*, Vol. 54, 1984, pp. 115–173.
- [167] Sod, G. A., “A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws,” *Journal of Computational Physics*, Vol. 27, 1978, pp. 1–31.
- [168] Coirier, W. J. and Powell, K. G., “An Accuracy Assessment of Cartesian-Mesh Approaches for the Euler Equations,” *Journal of Computational Physics*, Vol. 117, 1995, pp. 121–131.
- [169] Williamson, C., “Vortex Dynamics in the Cylinder Wake,” *Annual Review of Fluid Mechanics*, Vol. 28, 1996, pp. 477–539.

- [170] Henderson, R. D., “Details of the drag curve near the onset of vortex shedding,” *Physics of Fluids*, Vol. 7, No. 9, 1995, pp. 2102–2104.
- [171] Wieselsberger, C., “New data on the laws of fluid resistance,” Technical Note 84, NACA, March 1922.
- [172] Roshko, A., “On the Development of Turbulent Wakes from Vortex Streets,” Technical Note 1191, NACA, 1954.
- [173] Sheard, G. J., Hourigan, K., and Thompson, M. C., “Computational of the drag coefficients for low-Reynolds-number flow past rings,” *Journal of Fluid Mechanics*, Vol. 526, 2005, pp. 257–275.
- [174] Nastase, C. R. and Mavriplis, D. J., “High-order discontinuous Galerkin methods using an *hp*-multigrid approach,” *Journal of Computational Physics*, Vol. 213, 2006, pp. 330–357.
- [175] Northrup, S. A. and Groth, C. P. T., “Solution of Laminar Combusting Flows Using a Parallel Implicit Adaptive Mesh Refinement Algorithm,” *Proceedings of the 4th International Conference on Computational Fluid Dynamics, Ghent, Belgium, July 10–14, 2006*, 2006.
- [176] Rashad, R., *Development of a High-Order Finite-Volume Method for the Navier-Stokes Equations in Three Dimensions*, Master’s thesis, University of Toronto, 2009.
- [177] Zingg, D. W. and Yarrow, M., “A Method of Smooth Bivariate Interpolation for Data Given on a Generalized Curvilinear Grid,” *SIAM Journal for Scientific and Statistical Computing*, Vol. 13, 1992, pp. 687–693.
- [178] Gander, W. and Gautschi, W., “Adaptive Quadrature – Revisited,” *BIT Numerical Mathematics*, Vol. 40, 2000, pp. 84–101.
- [179] Bashein, G. and Detmer, P. R., *Graphics Gems IV*, chap. I.1, San Diego: Academic Press, 1994.
- [180] Maplesoft<sup>TM</sup>, “Maple,” Retrieved from <<http://www.maplesoft.com/>> (Accessed 4 July 2010).

- [181] Liu, Y. and Vinokur, M., “Exact Integrations of Polynomials and Symmetric Quadrature Formulas Over Arbitrary Polyhedral Grids,” *Journal of Computational Physics*, Vol. 140, 1998, pp. 122–147.

# APPENDICES

# Appendix A

## GEOMETRIC PROPERTIES OF QUADRILATERAL ELEMENTS AND INTEGRATION OF FUNCTIONS OVER SUCH DOMAINS

A key element for obtaining the desired high-order accuracy with the proposed CENO finite-volume algorithm is the calculation of geometric properties of quadrilateral mesh elements with an order of accuracy that is consistent with that of the scheme. Since many of the required properties are in fact defined as integrals of geometric functions (e.g. element centroid, area, geometric moments about the centroid etc.), the capability to perform accurate integration over convex quadrilateral domains becomes critical to the overall accuracy of the numerical procedure. Additionally, the integration algorithms employed for these calculations should be as efficient as possible. Although the efficiency condition may seem obviously necessary in a computational framework, that may not be the case with computation of geometric properties since many of these quantities (e.g. centroid, cell area etc.) are so often used in the solution procedure that it is usually more efficient to have them computed and stored as part of a pre-processing step. Consequently, for numerical computations requiring only fixed meshes any incurred inefficiencies in the geometry calculation can have a reduced impact to the overall performance of the numerical procedure, although they can significantly affect the computational cost of the pre-processing step. However, in the context of unsteady mesh adaptation these inefficiencies could become an important part of the overall cost of the numerical algorithm and thus, it is highly desirable to be avoided. Therefore, special

attention has been paid in this work to the efficiency with which the geometric properties of both straight- and curved-edge quadrilaterals are computed in all stages of the numerical scheme. Moreover, the capability to integrate accurately arbitrary functions over quadrilateral finite-volumes represents an important and desirable feature for procedures such as solution initialization with analytic functions and/or accuracy estimation. Thus, the numerical procedures employed for computing the necessary integrals must be accurate, robust and efficient, and it is highly desirable to offer flexibility in the integrand specification.

To address the aforementioned requirements, this research has considered the derivation of a collection of analytical results for situations in which they could be determined and the development of several numerical integration techniques, the selection of which is based on the function to be integrated and on the shape of the definition domain. Since the methods used for straight- and curved-edge quadrilaterals are not identical in general, the techniques used in each case are presented separately in the remainder of this section.

### Straight-edge quadrilaterals

A general numerical procedure can be developed for integrating arbitrary functions over these types of domain by employing a bilinear transformation [177] which maps the quadrilateral domain into a unit square. Thus, the integral  $\mathcal{J}$  of the arbitrary function  $f(x, y)$  over the quadrilateral domain  $\mathcal{D}$  can be expressed as an integration over the image domain  $\mathcal{D}'$  as

$$\mathcal{J} = \iint_{\mathcal{D}} f(x, y) dx dy = \iint_{\mathcal{D}'} f(x(p, q), y(p, q)) J dp dq, \quad (\text{A.1})$$

where  $J$  is the Jacobian of the bilinear transformation,  $J = \left| \frac{D(g, h)}{D(p, q)} \right|$ , and the functions  $g$  and  $h$  must exist and map the two spaces as follows:  $x = g(p, q)$  and  $y = h(p, q)$ . Note that the transformation Jacobian can be written as a linear function in  $p$  and  $q$  as  $J(p, q) = J_0 + J_1 p + J_2 q$ , where  $J_0$ ,  $J_1$  and  $J_2$  are completely defined by the coordinates  $(x, y)$  of the quadrilateral vertices and the coordinates  $p$  and  $q$  are chosen to vary in the interval  $[0, 1]$ . To perform the integration in the image domain, the adaptive Lobatto quadrature algorithm of degree 19 proposed by Gander and Gautschi [178] is used in each Cartesian space direction. Note that this adaptive integration algorithm has been already included and recommended for usage in the recent volume of Numerical Recipes [131]. Hence, based on this algorithm, the accurate integration of all arbitrary functions considered in this work was possible to be

performed with a reduced computational cost.

However, there are several geometric quantities required as part of the solution procedure that are defined as surface integrals of polynomials and thus, more efficient expressions than the general integration procedure outlined above can be used for their computation. For example, the centroid and area of the quadrilateral element is computed very efficiently with the algorithm for arbitrary polygonal shapes described in [179]. Furthermore, the calculation of geometric moments  $(\overline{x^{p_1}y^{p_2}})_I$  defined by Eq. (3.22) have been computed here based on analytical expressions derived for all powers  $(p_1, p_2)$  of interest in this work as function of the coordinates of the quadrilateral vertices. The derivation of the analytical expressions is based on the same bilinear transformation approach outline above to which symbolic integration with the Maple package [180] has been applied. As an illustration, the analytical expression for the cubic geometric moment of powers  $(2, 1)$  is shown in Table A.1, where the coefficients  $a_0, a_1, a_2, a_3, b_0, b_1, b_2, b_3, J_0, J_1$  and  $J_2$  are all calculated with the coordinates of the quadrilateral vertices as defined by the aforementioned bilinear transformation. The rest of the moments have similar expressions, the complexity of which decreases for lower powers and increases for higher ones, respectively. For that reason, analytical expressions for moments of relatively high orders can become more expensive to be evaluated than directly applying a numerical integration procedure, in which case the latter would be more appropriate for the computation of those geometric moments. Nevertheless, a more efficient implementation can be obtained with analytical expressions for not so high orders, such as the ones targeted in this work, by caching the constant coefficients that appear in the expressions and re-use them for a range of powers.

Other ways in which exact integration of the moments defined by Eq. (3.22) can be obtained have also been presented in the literature. A first possibility is to use Gauss' theorem to convert the surface integrals to contour integrals and then use Gaussian quadrature of appropriate order along the boundary of the control volume. A second choice is to split the integral over a quadrilateral cell into two integrals over triangular domains, and apply a closed form for the integration of polynomial expressions over the domain of these elementary shapes (i.e. triangles). Liu and Vinokur [181] used a coordinate free formulation and derived closed expressions in terms of the position vectors of vertices for integrals of polynomial products up to the fifth order. Their relations are unified for all the three fundamental shapes (i.e. straight-line segment, triangle and tetrahedron), and thus, their procedure can be ap-

Table A.1: Analytical expression for computing the geometric moment of powers  $(2, 1)$ ,  $(\overline{x^2 y^1})_I$ , function of the coordinates of the quadrilateral vertices.

$$\begin{aligned}
(\overline{x^2 y^1})_I = & \left[ 360 \left( J_0 a_0^2 b_2 + J_0 a_0^2 b_1 + a_0^2 b_0 J_1 + J_1 a_0 a_1 b_1 + a_0^2 b_0 J_2 + J_1 a_0 a_2 b_0 + \right. \right. \\
& \left. \left. + J_2 a_0 a_1 b_0 + J_0 a_0 a_2 b_1 + J_0 a_0 a_1 b_2 + J_0 b_0 a_0 a_3 + J_0 b_0 a_2 a_1 + J_2 a_0 a_2 b_2 \right) + \right. \\
& + 60 \left( J_0 a_3^2 b_2 + J_2 a_3^2 b_0 + J_1 a_2^2 b_3 + J_1 a_3^2 b_0 + J_0 a_3^2 b_1 + J_2 a_1^2 b_3 \right) + \\
& + 80 \left( J_0 a_3^2 b_0 + J_2 a_1^2 b_2 + J_1 a_2^2 b_1 \right) + 180 \left( a_2^2 b_2 J_0 + J_2 a_0^2 b_1 + J_0 a_0^2 b_3 + \right. \\
& \left. + a_1^2 b_1 J_0 + J_1 a_0^2 b_2 + J_1 a_1^2 b_0 + J_2 a_2^2 b_0 + J_1 a_0 a_1 b_3 + J_1 b_1 a_0 a_3 + \right. \\
& \left. + J_1 b_1 a_2 a_1 + J_1 a_1 a_3 b_0 + J_0 a_1 a_3 b_1 + J_2 a_0 a_2 b_3 + J_2 b_2 a_0 a_3 + J_2 b_2 a_2 a_1 + \right. \\
& \left. + J_2 a_2 a_3 b_0 + J_0 a_2 a_3 b_2 \right) + 120 \left( J_1 a_2^2 b_0 + J_2 a_1^2 b_0 + J_0 a_2^2 b_1 + J_1 a_0^2 b_3 + \right. \\
& \left. + J_0 a_1^2 b_2 + J_2 a_0^2 b_3 + J_1 b_3 a_0 a_3 + J_1 b_3 a_2 a_1 + J_1 a_2 a_3 b_1 + J_1 a_1 a_3 b_2 + \right. \\
& \left. + J_2 a_1 a_3 b_1 + J_0 a_1 a_3 b_3 + J_1 a_2 a_3 b_2 + J_2 b_3 a_0 a_3 + J_2 b_3 a_2 a_1 + J_2 a_2 a_3 b_1 + \right. \\
& \left. + J_2 a_1 a_3 b_2 + J_0 a_2 a_3 b_3 \right) + 240 \left( J_0 a_1^2 b_0 + J_2 a_0^2 b_2 + J_1 a_0^2 b_1 + J_0 a_2^2 b_0 + \right. \\
& \left. + J_1 a_0 a_1 b_2 + J_1 b_0 a_0 a_3 + J_1 b_0 a_2 a_1 + J_2 a_0 a_1 b_1 + J_0 a_0 a_1 b_3 + J_0 b_1 a_0 a_3 + \right. \\
& \left. + J_0 b_1 a_2 a_1 + J_0 a_1 a_3 b_0 + J_1 a_0 a_2 b_2 + J_2 a_0 a_2 b_1 + J_2 a_0 a_1 b_2 + J_2 b_0 a_0 a_3 + \right. \\
& \left. + J_2 b_0 a_2 a_1 + J_0 a_0 a_2 b_3 + J_0 b_2 a_0 a_3 + J_0 b_2 a_2 a_1 + J_0 a_2 a_3 b_0 \right) + \\
& + 72 \left( J_2 a_2^2 b_3 + J_1 a_1^2 b_3 \right) + 36 \left( a_3^2 b_3 J_1 + a_3^2 b_3 J_2 \right) + 48 \left( J_2 a_3^2 b_2 + J_1 a_3^2 b_1 \right) + \\
& + 144 \left( a_2^2 b_2 J_2 + a_1^2 b_1 J_1 + J_1 a_1 a_3 b_1 + J_2 a_2 a_3 b_2 \right) + 45 \left( J_1 a_3^2 b_2 + J_2 a_3^2 b_1 + \right. \\
& \left. + a_3^2 b_3 J_0 \right) + 90 \left( a_1^2 b_1 J_2 + J_2 a_2^2 b_1 + J_0 a_2^2 b_3 + J_1 a_2 a_3 b_3 + a_2^2 b_2 J_1 + \right. \\
& \left. + J_1 a_1^2 b_2 + J_0 a_1^2 b_3 + J_2 a_1 a_3 b_3 \right) + 720 \left( a_0^2 b_0 J_0 + J_0 a_0 a_2 b_0 + J_0 a_0 a_1 b_0 \right) + \\
& + 96 \left( J_1 a_1 a_3 b_3 + J_2 a_2 a_3 b_3 \right) + 160 \left( J_1 a_0 a_2 b_3 + J_1 b_2 a_0 a_3 + J_1 b_2 a_2 a_1 + \right. \\
& \left. + J_1 a_2 a_3 b_0 + J_2 a_0 a_1 b_3 + J_2 b_1 a_0 a_3 + J_2 b_1 a_2 a_1 + J_2 a_1 a_3 b_0 + J_0 b_3 a_0 a_3 + \right. \\
& \left. + J_0 b_3 a_2 a_1 + J_0 a_2 a_3 b_1 + J_0 a_1 a_3 b_2 \right) + 480 \left( J_1 a_0 a_1 b_0 + J_0 a_0 a_1 b_1 + \right. \\
& \left. + J_2 a_0 a_2 b_0 + J_0 a_0 a_2 b_2 \right) \left. \right] \frac{1}{720 A_I}
\end{aligned}$$

plied regardless of the dimensionality of the problem (i.e. one-, two- or three-dimensions). As previously described, this research uses analytical results obtained for polynomials up to fourth order for convex quadrilaterals with straight edges, which have been considered more efficient than the other techniques. However, the Gauss' theorem approach represents an important technique heavily employed for integration over quadrilaterals which have at least one of the faces curved. Equally important is the analytical integration of polynomials along a straight-line segment which represents a basic ingredient in curvilinear integration over arbitrary geometric splines.

The computation of other geometric properties required by the solution procedure, such as the lengths of the edges and the normal unit vectors at the Gauss-quadrature point locations, is carried out with relationships that are widely available in many textbooks about analytical geometry.

### **Curved-edge quadrilaterals**

Undoubtedly, computation of geometric properties and integration of arbitrary functions over curved-edge quadrilaterals is far more complicated than on straight-edge counterpart domains. Moreover, general available techniques such as Monte-Carlo integration may be either converging too slowly or require too many function evaluations to be always used. As anticipated in the previous section, a more efficient mean than Monte-Carlo integration approach [131] for computing integrals of functions for which one dependency can be integrated analytically is to use Green-Gauss' theorem for converting surface integrals to contour integrals, which represents a two-dimensional special case of the more general Stokes' theorem. The reader is reminded that the curved geometric boundaries of the quadrilateral element are segments of the block boundary splines, for which a representation (e.g., Lagrange-type interpolation) of an appropriate order to that of the interior scheme is considered to be known. Consequently, the path  $\mathcal{S}$  along which contour integration would be required is precisely known everywhere between the end vertices.

To illustrate how the integral over the surface of the quadrilateral element,  $\mathcal{A}_I$ , is converted to integration along the element contour,  $\partial\mathcal{A}_I$ , the computation of the geometric

moment defined by Eq. (3.22) is considered next. Thus, the moment  $(\overline{x^{p_1}y^{p_2}})_I$  is obtained as

$$(\overline{x^{p_1}y^{p_2}})_I = \frac{1}{A_I} \iint_{A_I} (x - \bar{x}_{i,j})^{p_1} (y - \bar{y}_{i,j})^{p_2} dx dy \quad (\text{A.2})$$

$$= \frac{1}{(p_1 + 1) A_I} \oint_{\partial A_I} (x - \bar{x}_{i,j})^{p_1+1} (y - \bar{y}_{i,j})^{p_2} dy \quad (\text{A.3})$$

$$= \frac{1}{(p_1 + 1) A_I} \sum_{l=1}^{N_f} \oint_{\mathcal{S}_l} (x - \bar{x}_{i,j})^{p_1+1} (y - \bar{y}_{i,j})^{p_2} dy, \quad (\text{A.4})$$

in which a curvilinear integral must be computing along each face  $\mathcal{S}_l$  of the quadrilateral. As mentioned earlier, a maximum of two curved faces can occur in a given control volume.

Two approaches have been considered in this work for computing the curvilinear integrals with respect to the  $y$ -Cartesian coordinate along curved paths in Eq. (A.4). The first technique is to divide the path  $\mathcal{S}_l$  into a series of straight-line segments which recover the “true” curvilinear geometry with an accuracy consistent with the order of the numerical scheme. Thus, the integral with respect to  $y$  is computed as the sum of the integrals for each segment. For polynomial functions each piecewise integral can be obtained analytically. Needless to say that this is the method used for computing integrals along the straight edges of the quadrilateral. A direct application of the integration technique described above is illustrated herein with the calculation of the element area, which can be expressed as

$$A = \oint_{\partial A} x dy = \sum_{l=1}^{N_f} \oint_{\mathcal{S}_l} x dy = \sum_{l=1}^{N_f} \sum_{n=0}^{N_s} \left( \int_{y_1}^{y_2} x dy \right)_{n,l} = \sum_{l=1}^{N_f} \sum_{n=0}^{N_s} \left( \Delta y \left[ x_1 + \frac{1}{2} \Delta x \right] \right)_{n,l}, \quad (\text{A.5})$$

where  $N_s$  is the number of straight-line segments that make up the contour  $\mathcal{S}_l$  and  $(x_1, y_1)$  and  $(x_2, y_2)$  are the Cartesian coordinates defining the vertices of each line segment. Although this approach is capable of providing accurate results which obviously depend on the number of segments,  $N_s$ , it is quite expensive to be used for computing all geometric properties requiring integration (e.g. area, centroid and geometric moments up to the polynomial order) and therefore, an alternative approach has also been developed in this work.

By using Gauss quadrature integration of appropriate order along those integration paths  $\mathcal{S}_l$  that are curved, a more efficient algorithm to compute the curvilinear integrals in Eq. (A.4) can be obtained. Thus, considering that the functions  $x = x(s)$  and  $y = y(s)$  exist and provide the Cartesian coordinates of the points along the curved path  $\mathcal{S}_l$ , which in turn is part of the

integration contour  $\partial\mathcal{A}_I$ , the curvilinear integral in Eq. (A.4) is calculated with Gaussian quadratures as follows

$$\mathcal{J} = \oint_{s_i} (x(s) - \bar{x}_{i,j})^{p_1+1} (y(s) - \bar{y}_{i,j})^{p_2} \frac{dy(s)}{ds} ds = \oint_{s_i} F(s) ds = L_l \sum_{m=1}^{N_G} (\omega F)_m, \quad (\text{A.6})$$

where the function  $F(s) = (x(s) - \bar{x}_{i,j})^{p_1+1} (y(s) - \bar{y}_{i,j})^{p_2} \frac{dy(s)}{ds}$  is evaluated at each of the  $N_G$ -Gauss quadrature points. In Eq. (A.6)  $\omega$  is the quadrature weighting coefficient,  $L_l$  is the total arc length and the locations of the Gauss integration points are computed by taking the corresponding abscissas along the arc length of the integration path. For functions other than polynomials, the monomial expression in  $F(s)$  is replaced accordingly. The derivative of the  $y$ -coordinate with respect to the path length at each integration point can be obtained either numerically by finite-differences or analytically by differentiating the expression  $y = y(s)$  if it is known explicitly. Note that the same technique can also be applied to compute the normal unit vector at the flux calculation quadrature points, which gives  $\vec{n}(s) = \left( \frac{dy(s)}{ds}, -\frac{dx(s)}{ds} \right)$ .

The number of Gauss integration points used to compute the curvilinear integrals vary with the order of the numerical scheme. For cubic and quartic reconstructions the approach considered herein is to use a five-point integration rule which provides exact results for polynomials up to nine order. Note that determining the location of the integration points is typically an expensive operation and therefore, it is more efficient to store them after their calculation. Note also that the order of the monomial to be integrated along the contour is one order higher than the one which originally needs to be integrated over the surface, and therefore, higher-order quadrature rules are required. Moreover, any errors in determining the coordinates of the centroid in Eq. (A.4) tend to amplify the errors in the monomial integration, especially for higher orders, and that can deteriorate significantly the accuracy of the high-order moments. Consequently, the preferred approach in the current work is to compute the cell area and centroid with the first technique outlined above (i.e., summation of integrals on straight-line segments) and the geometric moments with the Gauss quadrature integration algorithm previously mentioned. Another advantage of having the Gauss quadrature points readily available is that computation of functionals such as lift and drag forces over curved shaped bodies can be carried out very efficiently. Thus, a good balance between accuracy and computational efficiency can be obtained.

# Appendix B

## CALCULATION OF GEOMETRIC MOMENTS

As shown in Chapt. 3, two types of geometric moments arise in the computation of the  $k$ -exact reconstruction and they can be categorized based on the relationship between the integration domain and the centroid with respect to which the moment is taken. Thus, the geometric moments defined by Eq. (3.22) are local to each control volume because the centroid is part of the integration domain. In contrast, the geometric moments given by Eq. (3.23) describe a relationship between neighbouring cells because the integration is performed on one element and the centroid belongs to a different cell. We shall see shortly that the latter geometric moments can be derived using knowledge about the local ones and the distance between the centroids of the two control volumes.

Since the geometric moments depend only on the mesh geometry, in theory they could all be computed as part of a pre-processing step and stored for later re-use, requiring only to be updated when mesh refinements occur. However, it turns out that their number is quite large (e.g. for  $k=3$  in two-dimensions there are 250 geometric moments for each cell of the domain) and therefore, more judgement is required before committing memory storage to these parameters, especially for large simulation problems. Note that these geometric moments represent the “building blocks” for the entries in the LHS matrix of the least-squares linear system given by Eq. (3.19). Consequently, instead of storing these moments if the memory storage is available, a more efficient alternative is to store directly the pseudo-inverse,  $\mathbf{A}^{-1}$ , of the LHS term, as described in Sect. 3.6.1. On the contrary, when not sufficient memory is available, it should be obvious that recalculating *all* geometric moments for each cell reconstruction at each time step would make the high-order reconstruction approach discouragingly expensive. Besides, even the computation of the pseudo-inverse

matrix in the pre-processing step could become an expensive operation.

To address these issues, a solution is to store only part of the geometric moments whereas the others to be computed by geometric translations, as previously mentioned. Therefore, the local control volume moments given by Eq. (3.22) could be computed and stored for each cell and the moments requiring integration over the domain of a neighbour cell are obtained based on Eq. (3.23) as follows:

$$\begin{aligned}
\left(\widetilde{x^{p_1}y^{p_2}}\right)_{IJ} &= \frac{1}{\mathcal{A}_J} \iint_{A_J} [(x - x_{\gamma,\delta}) + (x_{\gamma,\delta} - x_{i,j})]^{p_1} [(y - y_{\gamma,\delta}) + (y_{\gamma,\delta} - y_{i,j})]^{p_2} dx dy - (\overline{x^{p_1}y^{p_2}})_I \\
&= \sum_{l=0}^{p_2} \sum_{q=0}^{p_1} \left[ C_{p_1}^q C_{p_2}^l (x_{\gamma,\delta} - x_{i,j})^q (y_{\gamma,\delta} - y_{i,j})^l \left(\overline{x^{p_1-q}y^{p_2-l}}\right)_J \right] - (\overline{x^{p_1}y^{p_2}})_I \\
&= \sum_{l=0}^{p_2} \sum_{q=0}^{p_1} \left[ C_{p_1}^q C_{p_2}^l \Delta x_{IJ}^q \Delta y_{IJ}^l \left(\overline{x^{p_1-q}y^{p_2-l}}\right)_J \right] - (\overline{x^{p_1}y^{p_2}})_I, \tag{B.1}
\end{aligned}$$

where the binomial coefficients  $C_\alpha^\beta$  can be computed efficiently by recursion as

$$C_\alpha^\beta = \frac{\alpha - \beta + 1}{\beta} C_\alpha^{\beta-1}; \quad C_\alpha^0 = 1, \tag{B.2}$$

and the  $x$ - and  $y$ -distance between the centroids of the two neighbour cells,  $\vec{X}_I(x_{i,j}, y_{i,j})$  and  $\vec{X}_J(x_{\gamma,\delta}, y_{\gamma,\delta})$ , are denoted by  $\Delta x_{IJ}$  and  $\Delta y_{IJ}$ , respectively. In this approach, the memory requirements for the aforementioned cubic reconstruction are reduced to only 10 moments that need to be stored in each cell of the domain. Additionally, some reasonable extra computational cost is incurred due to geometric translations.

Finally, the reader is reminded that a complete presentation of the methods considered in this work for computing the surface integral present in the definition of the local geometric moments,  $(\overline{x^{p_1}y^{p_2}})_I$ , is provided in Appendix A.