

A Novel Approach for Temporal Discretizations with High-order Finite Volume Schemes

Pedro Miguel Pereira Costa

Thesis to obtain the Master of Science Degree in

Mechanical Engineering

Supervisors: Dr. Duarte Manuel Salvador Freire Silva de Albuquerque
Prof. José Carlos Fernandes Pereira

Examination Committee

Chairperson: Prof. Carlos Frederico Neves Bettencourt da Silva
Supervisor: Dr. Duarte Manuel Salvador Freire Silva de Albuquerque
Member of the Committee: Dr. Gonçalo Nuno Guerreiro de Jesus Silva

November 2019

To my three grandmothers,

Acknowledgments

I would like to express my appreciation to my supervisors Dr. Duarte Manuel Salvador Freire Silva de Albuquerque for his patient guidance and his generous availability throughout every stage of this Thesis and Prof. José Carlos Pereira for introducing me to the CFD field in his MFC course.

My special thanks is extended to FCT, who financially supported this work through a grant from the research project: High-order immersed boundary for moving body problems - HIBforMBP - with the reference PTDC/EME-EME/32315/2017.

To my friends Rui, Bárbara and Francisco for all the encouragement during this work.

Finally I would like to thank my parents and sister for all the support throughout my degree.

Resumo

No método de Volume Finito, o termo temporal da equação tem de ser integrado no espaço com a mesma precisão que os esquemas convectivo e difusivo de alta ordem. Este problema é ultrapassado através de valores computacionais como médias do volume de controlo. Outra formulação é considerar um novo operador que converte valores computacionais pontuais em médios, o que permite resolver problemas transientes usando valores pontuais. Uma vantagem destes esquemas é o número baixo de contribuições quando comparado com os esquemas baseados em valores médios.

No espaço bidimensional, este operador usa o método de mínimos-quadrados para calcular polinómios locais e quadratura de Gauss para integrar os vários momentos de inércia das células. O código foi verificado para quarta, sexta e oitava ordens em malhas cartesianas e não-estruturadas, quando o esquema temporal Crank-Nicolson é usado. Ambas condições de fronteira Dirichlet e Neumann foram verificadas com sucesso. Como métrica de eficiência dos esquemas de alta ordem implementados, foi estudada a evolução do erro numérico espacial com o tempo de execução e com a memória. Os esquemas de alta ordem permitiram obter resultados mais rápidos e com uma maior precisão do que os esquemas de segunda ordem. Para reduzir o número de avanços temporais, os esquemas BDF foram verificados e estes revelaram uma redução do tempo de execução quando combinados com esquemas espaciais de alta ordem.

Palavras-Chave: Esquemas de alta ordem, Simulações transientes, Método de volume finito, mínimos-quadrados ponderados, malhas poliédricas não-estruturadas, Diferenças finitas regressivas.

Abstract

With the finite volume method (FVM), the temporal term of the equation must be integrated in space with the same accuracy order as the convection and diffusion schemes. One classic way to solve this issue is to consider average computational values in FVM. Another approach is to consider a new operator that converts pointwise values to mean ones, which enables high-order convection and diffusion schemes of the pointwise framework in unsteady problems. One advantage of these schemes is the low number of contributions to the coefficient matrix when compared to the mean-value counterpart.

In the 2D space, this mean operator uses the weighted least-squares method to compute local polynomials and the Gauss quadrature to integrate cell inertial momentums (up to the desired order). The code was verified for fourth, sixth and eighth orders in both Cartesian and unstructured meshes when using the Crank-Nicolson time discretization. Dirichlet and Neumann boundary conditions were also successfully verified. Additionally, the numerical spatial error evolution with the solver runtime and the required memory are studied as efficiency metrics of the implemented high-order schemes. High-order spatial schemes provided faster and more accurate results than the second-order counterpart. To reduce the number of required time steps, high-order backward differentiation formula (BDF) schemes were verified and lead to considerable time savings when using high-order spatial schemes.

Keywords: High-order schemes, Transient simulations, Finite volume method, Weighted least-squares, Polyhedral unstructured grids, Backward differentiation formulas.

Contents

Acknowledgments	v
Resumo	vii
Abstract	ix
List of Tables	xiii
List of Figures	xv
Nomenclature	xvii
Glossary	xix
1 Introduction	1
1.1 Motivation	2
1.2 Background	3
1.3 SOL Code	4
1.4 Stationary Convection-Diffusion Scheme in SOL code	4
1.5 Objectives	5
1.6 Present contributions	6
1.7 Thesis Outline	6
2 High order transient schemes in One Dimensional Space	7
2.1 Discretization of the unsteady convection-diffusion equation using Finite Volume Method	7
2.1.1 General Equation	7
2.2 Proposed Transient Schemes	8
2.2.1 Proposed Mean Operator	8
2.2.2 Transient Transport Equation Discretization	11
2.3 Results and Discussion	11
2.3.1 Error Assessment	12
2.3.2 Peclet Number	13
2.3.3 Pointwise to Mean Value Operator	13
2.3.4 Transient Results	14
2.3.5 Further Considerations	17
2.4 High Order Temporal Schemes	21
2.4.1 Results and Discussion	22

2.5	Conclusion	24
3	High order transient schemes in Two Dimensional Space	25
3.1	Unstructured Grids	25
3.2	Proposed Schemes	26
3.2.1	Stencil	26
3.2.2	Local Polynomial Reconstructions	27
3.2.3	Boundary Treatment	28
3.2.4	Inertial Momentums	29
3.2.5	Transport Equation Discretization	30
3.2.6	Unstructured Grids in 2D	31
3.3	Results and Discussion for Cartesian grids using Crank-Nicolson time discretization . . .	32
3.3.1	Mean Operator	32
3.3.2	Temporal Discretization	34
3.3.3	Spatial Discretization	34
3.3.4	Further Considerations	36
3.4	High Order temporal schemes using Cartesian grids	38
3.4.1	Temporal Orders	39
3.4.2	Solver Runtime	40
3.5	Results and Discussion for Unstructured Grids	41
3.5.1	Spatial Error Orders	41
3.5.2	Solver Runtime	43
3.5.3	Memory Storage	43
3.6	Conclusion	45
4	Conclusion	47
4.1	Contributions	47
4.2	Future Work	48
	Bibliography	49

List of Tables

2.1	Coordinates and Gauss weights for a $[-1,1]$ interval used in Gauss quadrature integration	10
2.2	The $\ e\ _1$ estimated for BDF2,BDF4, BDF6 and CN schemes and the respective ratios, using as reference the value from BDF2.	23
3.1	Number of interpolation terms demanded by each n-th spatial order for the 2D case. . . .	26
3.2	Simplex coordinates and Gaussian weights necessary to perform Gauss Integration in a triangle, data taken from [59] and [60]	30
3.3	SRT estimated for BDF2,BDF4, BDF6 and CN schemes for sixth-order scheme and the respective ratios, using as reference the value from BDF2.	41
3.4	SRT estimated for a spatial error of E-5 with irregular polyhedral, triangular, quadrilateral and hybrid grids for sixth-order scheme and the respective ratios, using as reference the value from hybrid meshes.	45
3.5	NNZ estimated for spatial error of E-5 with irregular polyhedral, triangular, quadrilateral and hybrid grids for sixth-order scheme and the respective ratios, using as reference the value from hybrid meshes.	45

List of Figures

2.1	Mean Operator's spatial accuracy order for all schemes for the full Dirichlet and mixed BCs cases	14
2.2	$\ e\ _1$ as a function of timesteps for eighth-order using full Dirichlet (blue lines) and mixed BC (red lines) and the Crank-Nicolson time discretization	15
2.3	$\ e_{sp}\ _1$ as a function of grid size for all schemes using full Dirichlet and Mixed BCs and its spatial orders	16
2.4	$\ e\ _1$ as a function of number of timesteps for Dampening Exponential Solution using implicit Euler time discretization ($\alpha = 1$) with full Dirichlet (blue lines) and mixed BC (red lines)	17
2.5	Spatial error for all schemes for Dampening Exponential analytical case using Non-Uniform grids with full Dirichlet (blue lines) and mixed BCs (red lines)	18
2.6	Spatial error decay with grid size for the Travelling Wave analytical case for all schemes setting $Pe = 100$ and $Pe = 10000$ using full Dirichlet and mixed BCs	19
2.7	$\ e_{sp}\ _1$ as function of SRT for all schemes using CN	20
2.8	$\ e_{sp}\ _1$ as function of NNZ for all schemes using CN	21
2.9	$\ e\ _1$ as function of timestep size for BDF2, BDF4 and BDF6 for Travelling Wave analytical solution using eighth spatial order scheme in uniform 1D mesh with 35 cells and using full Dirichlet and mixed BCs	22
2.10	$\ e_{sp}\ _1$ as function of SRT for BDF2, BDF4 and BDF6 for Travelling Wave solution in uniform 1D mesh using full Dirichlet BC	23
3.1	Face neighbours of cell C that are used to compute pointwise to mean value conversion operator	27
3.2	Examples of points used in the 2D Gauss quadrature integration [36]	29
3.3	Samples of irregular grid categories studied in this Thesis and which were also previously studied by Diogo [37] for steady cases.	31
3.4	Mean operator error and respective accuracy order with Cartesian grids for all high-order schemes using both full Dirichlet and mixed BC cases	33
3.5	$\ e\ _1$ as a function of timestep for the eight-order spatial scheme with Cartesian grids while using CN temporal scheme and for both full Dirichlet and mixed boundary conditions. . .	34

3.6	$\ e_{sp}\ _1$ as a function of H with Cartesian grids for all spatial schemes using full Dirichlet and mixed BCs and with the computed order between each mesh refinement.	35
3.7	Spatial error for Travelling Wave analytical case for all schemes setting different values of convection velocities, U_x and U_y for the cases with full Dirichlet and mixed BCs	36
3.8	$\ e_{sp}\ _1$ as function of SRT for all schemes using CN and full Dirichlet BC	37
3.9	$\ e_{sp}\ _1$ as function of NNZ for all schemes using the CN time discretization and only Dirichlet BC.	38
3.10	$\ e\ _1$ as function of the timestep for BDF2, BDF4 and BDF6 time schemes for Travelling Wave analytical solution using eighth-order scheme in 2D Cartesian mesh with 30 per 30 cells	39
3.11	Spatial error evolution with SRT for all temporal and spatial schemes at study for Travelling Wave analytical case	40
3.12	$\ e_{sp}\ _1$ as function of H for the Dampening Cosine analytical case using irregular unstructured grids, the CN time scheme and full Dirichlet BC.	42
3.13	$\ e_{sp}\ _1$ as function of SRT for Dampening Cosine analytical case for the case with only Dirichlet BC.	43
3.14	$\ e_{sp}\ _1$ as function of NNZ for Dampening Cosine analytical case using CN time scheme and full Dirichlet BC.	44
3.15	$\ e_{sp}\ _1$ as function of SRT and NNZ for sixth-order spatial scheme in the Dampening Cosine analytical case.	44

Nomenclature

Roman Symbols

x_C, y_C	Cell Centroid.
V_C	Cell Volume.
V_p	Cell Volume Matrix.
U	Convection Velocity.
A	Convection-Diffusion Matrix.
x_f, y_f	Coordinates of the Face Centroid.
w_G	Gauss Weight.
M_{cells}	Global Mean Operator.
$\overline{x^i y^j}$	Inertial Momentum.
M_C	Local Mean Operator vector.
P	Local Polynomial Coefficients Vector.
C	Local Polynomial Interpolation Coefficients.
D_C	Local Polynomial Interpolation Matrix.
\max	Maximum of a Function.
e_{mv}	Mean Operator Error.
e_1	Norm-1 of Global Error.
n_s	Number of Stencil Points.
n_G	Number of Gauss Points.
O	Order.
Pe	Peclet Number.
P	Polynomial Interpolation Matrix.
r	Ratio Between Two Quantities.
M_{faces}	Rectangular Matrix with Boundary Faces Contributions to the Mean Operator.
H	Reference Length, Cell Size.
e_{sp}	Spatial Error.
n	Spatial Order.
p	Temporal order.
S	Transient Source Term Vector.
\vec{n}	Unit Normal Vector.
W_C	Weight Function Matrix.
W_{LS}	Weighted-Least Square Matrix.

Greek Symbols

Δx	Cell's Length.
α	Crank-Nicolson Parameter.
Γ	Diffusion Coefficient.
ζ	Gauss Simplex Coordinates.
ϕ	Pointwise Solution.
ϕ_C	Pointwise Solution in Cell Centroid.
$\overline{\phi_C}$	Mean-value Solution in Cell Centroid.
Δt	Timestep.
φ_ϕ	Transient Source Term.

Operators

$ \bullet $	Absolute Value.
$\nabla \cdot$	Divergence Operator.
∇	Gradient Operator.
∂	Partial derivative.
\sum	Sum.

Subscripts

C	Cell.
f	Face.
LS	Least Squares.
s	Stencil.

Superscripts

-1	Inverse.
R	Reconstruction.
$n + 1, n, \dots$	Time Level.
T	Transpose.

Acronyms

<i>AIAA</i>	American Institute of Aeronautics and Astronautics.
<i>BDF, BDF_n</i>	Backward Differentiation Formula (n-th order).
<i>BC</i>	Boundary Condition.
<i>CFD</i>	Computational Fluid Dynamics.
<i>CV</i>	Control Volume.
<i>CN</i>	Crank-Nicolson [second-order time scheme].
<i>FV</i>	Finite Volume.
<i>LES</i>	Large Eddy Simulation.
<i>NNZ</i>	Number of Non-Zeros.
<i>PDE</i>	Partial Differential Equation.
<i>RANS</i>	Reynolds Average Numerical Simulation.
<i>SRT</i>	Solver Runtime.
<i>TVD</i>	Total Variation Diminishing.
<i>WLS</i>	Weighted Least-Squares.

Chapter 1

Introduction

CFD was first attempted by Richardson *et al.* [1] for weather predictions, who left us an unconditionally unstable time-centred diffusion scheme. Stability was later addressed by Courant, Friedrichs and Lewy [2]. Their discoveries would just prove fruitful with the appearance of the first programmable computer. In the beginning progress was achieved in US laboratories for weapons research, namely the violent flow caused by an atomic bomb. In these early times codes were always modelled after Von Neumann-Richtmeyer method which had a first-order accuracy [3].

During the sixties, second- and higher-order methods were proposed, such as Lax-Wendroff scheme [4], but they were not suitable for strong shocks or even the advection of water vapour. Subsequently, in the early seventies Boris *et al.* [5] and Van Leer [6] introduced the non-oscillatory higher-order methods. During the seventies, computational aerodynamics lead to potential-flow models. In the eighties, with vector computers it was possible to compute solutions of Euler equations in two and three dimensions. Huge advances were achieved in shock capturing simulations. Three-dimensional Euler solutions were computed using second-order Finite Volume methods, such as by Jameson, Schmidt and Turkel [7]. In the end of the same decade aerospace sciences were able to solve Euler equations in unstructured meshes.

CFD has always been strongly steered by the evolution of modern computers when it comes to faster computing, larger memory or in the way numerical algorithms are designed, for example, its possibility to be parallelized [8]. Therefore, in the nineties with the increase in computers' speed and memory capacity, it was possible to perform simulations using Reynolds averaged Navier-Stokes (RANS) equations. Until today, RANS is essentially limited to steady attached flows. NASA CFD Vision 2030 Study (2014) [9] also states that the current methods are still unable to predict turbulent-separated flows. RANS cannot predict reliably separated and unsteady flow phenomena such as buffet. So wind tunnel testing is still cheaper than a very large number of RANS calculations [10].

Vector computers changed the paradigm of CFD in the eighties as the number of flops per second increased drastically and kept gradually improving. However, in the early 2000s the bandwidth did not keep up its rate and it is seen as a major issue nowadays to overcome [11]. LES simulations require exascale computing to design a whole aircraft's fuselage [12]. Although some hybrid RANS-LES methods

are commonly used in simulations where swirls and intentionally detached flows are present, methods combining near wall RANS and far field LES are still in its infancy [9].

Especially as most computation time is spent solving the equations near-wall, designing new numerical methods such as higher-order schemes may be a way to reduce the number of flops in a simulation, increasing its speed.

1.1 Motivation

CFD is based on numerical analysis and provides insight into fluid dynamics. Therefore one of its main goals is to achieve the most accurate results and in the fastest way possible. This thesis concerns with the numerical solution of 2D transient convection-diffusion equation in unstructured grids. Its applicability might range from the medical field to the aerospace engineering one.

In the biomedical field different approaches to kill the tumour through hyperthermia are still being considered such as Magnetic Fluid Hyperthermia [13]. In this treatment patients are subjected to local heating leading to tumour cells' destruction. Simulations are important to minimize the unnecessary normal cells' death and better predict the needed time exposure to radiation. The play of nanoparticles in hyperthermic cancer treatment is largely modelled with Pennes' bio-heat equation [14], which describes the capability of a tissue to remove heat by conduction or to its surrounding blood. According to Sawicki *et al.* (2014) [15] temperature rise may be much faster than expected and a high-order temporal discretization alongside high spatial order method may provide faster and more accurate results.

Fractional differential equations describe accurately the anomalous diffusion phenomena present in glasses, polymers, proteins or organisms. These systems are not described by the exponential time function and are therefore considered sub-diffusion [16]. It also helps predicting the spread of diseases, especially useful to depict diseases in which the patient's recovery rate is dependent on time since infection [17]. The present thesis may be extended to solve fractional derivatives using fractional power series as basis functions in the local spatial polynomial reconstruction [18].

According to Ekaterinaris, second-order FV schemes do not compute accurately vorticity in aerodynamic simulations [19]. He suggests that applying higher-order and low-diffusive numerical schemes counteract this present deficiency. High-order schemes also have the potential to provide more accurate results for eddy simulations of separated flows which are highly unsteady.

1.2 Background

The development of high-order methods in CFD is essentially motivated by the need to reduce the computational cost and memory usage to achieve a required accuracy [20]. Unstructured grids are highly flexible at discretizing 2D and 3D complex domains [21]. These methods impose new considerations and justify the ongoing research. High-order FV methods are a natural extension to current commercial software, such as Fluent and Star-CCM++, which uses second-order FV schemes, and opensoftware like OpenFOAM.

Solving convection-diffusion equations using FV methods was introduced in the sixties by Tikhonov and Samarskii [22] and revived in the eighties by Patankar [23]. Barth and Frederikson (1990) [24] introduced a quadratic reconstruction to solve Euler equations and more recent efforts led to schemes up to fourth-order by Ollivier-Gooch *et al.* [25, 26, 27] and sixth-order schemes on unstructured grids by Clain *et al.* [28, 29, 30]. A comparison study by Nogueira *et al.* [31] showed that, for the same accuracy level, FV schemes require four times less degrees of freedom than its FEM counterpart when solving the Navier-Stokes equations. To avoid oscillations near discontinuities more recent techniques such as essentially non-oscillatory (ENO) [32] and weighted essentially non-oscillatory (WENO) [33] were developed for high-order schemes. Ollivier-Gooch *et al.* [34] used a fourth-order FV scheme to solve a flow around a NACA 0012 airfoil, which showed improvements in accuracy and Jalali *et al.* [35] presented another one for highly-anisotropic grids, which can be used in simulations for turbulent boundary layers. The present Thesis expands on the work of Vasconcelos [36] and Diogo [37], who developed convection-diffusion schemes (up to eighth-order) for steady cases on unstructured grids.

In the case of high-order temporal schemes, three popular families are linear multistep methods, Runge-Kutta method and Lax-Wendroff method. In the case of multistep methods the Adams-Bashforth [38], Adams-Moulton [39] and BDF schemes can be considered sub-families.

BDF schemes are general linear methods, which were first mentioned by Curtis and Hirschfelder (1952) [40]. These schemes were dismissed by Farnell (1962) [41] as they were less accurate than Adams-Moulton, which is true for non-stiff equations. BDF schemes up to sixth-order are stable for stiff problems and allow larger timesteps than explicit schemes. BDF2 is L-stable and very popular, it is even used in software such as COMSOL. BDF3 is not A-stable and it performs inconsistently [42]. Therefore extended BDF (EBDF) [43] and modified EBDF (MEBDF) [44] were developed and they are A-stable up to fourth-order. The two implicit advanced step-point (TIAS) method is A-stable up to sixth-order [45].

Implicit Runge-Kutta methods are stage-based. While fully-implicit schemes have been focus of research [46], diagonally-implicit Runge-Kutta schemes demand more evaluations to reach the same accuracy but stages equations are not solved simultaneously. Singly-diagonally implicit Runge-Kutta methods (SDIRKs) are very popular since its creation during the seventies [47] and SIMPLE by Nørsett *et al.* (1984) [48]. Unfortunately implicit SSP schemes' properties are only assured under a finite timestep and it is not much larger than its explicit scheme counterpart.

Lax-Wendroff schemes [4] are an alternative to the ones presented before. They use the Taylor series expansion in order to increase its order of accuracy while replacing the time derivatives with the usage of the PDE and its differentiated versions. Contrary to SSP Runge-Kutta, it only needs one upwind phase per timestep and allows for more compact stencils [49]. The original work was a second-order scheme [4] and recently it has been extended to fourth-order by Li *et al.* [50] with WENO spatial scheme.

1.3 SOL Code

Sol code was first developed in 2003 by Magalhães during his PhD work [51]. It is programmed in C language and the goal was to create an in-house code. Several authors' contributions may lead to an ever increasing and common code which solves complex cases in the CFD field.

Magalhães developed and implemented numerical solutions on adaptative grids. The current coordinator, Albuquerque [52], created a system to import grids, reviewed aspects which were already implemented and extended the second-order schemes to unstructured and adaptative grids. The code was extended to solve immersed boundary problems by Pereira [53] and Martins [54], and to turbulent jet flows with LES modelling by Leite [55].

This thesis directly extends the work of Vasconcelos [36] who created a high-order diffusion scheme and Diogo [37] who followed it with the addition of a high-order convection scheme and a new stencil algorithm. Both works were verified up to eighth order of accuracy in 2D unstructured grids.

Currently it is possible to import unstructured grids both from STAR CCM+ and the open-source OpenFOAM through ASCII files. Both options allow to test new grids in different numerical schemes as well as new implemented approaches.

1.4 Stationary Convection-Diffusion Scheme in SOL code

For the convection-diffusion schemes, Gooch (2002) [25] uses a Taylor series that respects the mean value. In this framework the local reconstruction around point i for each cell point j of the stencil is given by:

$$\begin{aligned} \bar{\phi}_j = \frac{1}{V_j} \int_{V_i} \phi_i^R dV = \phi_i + \frac{\partial \phi_i}{\partial x} (\bar{x}_j + (x_j - x_i)) + \frac{\partial \phi_i}{\partial y} (\bar{y}_j + (y_j - y_i)) + \frac{1}{2} \frac{\partial^2 \phi_i}{\partial x^2} (\bar{x}_j^2 + 2\bar{x}_j(x_j - x_i) \\ + (x_j - x_i)^2) + \frac{\partial^2 \phi_i}{\partial xy} (\bar{x}_j \bar{y}_j + \bar{x}_j(y_j - y_i) + \bar{y}_j(x_j - x_i) + (x_j - x_i)(y_j - y_i)) + \dots \end{aligned} \quad (1.1)$$

In SOL code it was implemented a pointwise framework, which can be represented by:

$$\phi_j = \phi_i + \frac{\partial \phi_i}{\partial x} (x_j - x_i) + \frac{\partial \phi_i}{\partial y} (y_j - y_i) + \frac{1}{2} \frac{\partial^2 \phi_i}{\partial x^2} (x_j - x_i)^2 + \frac{\partial^2 \phi_i}{\partial xy} (x_j - x_i)(y_j - y_i) + \frac{\partial^2 \phi_i}{\partial y^2} y_i^2 + \dots \quad (1.2)$$

By comparing both Taylor series, it is noticeable that the number of required terms for each derivative starts to increase in the mean framework, while it does not happen in the pointwise counterpart.

For example for the mean Taylor series the seventh derivative with respect to x requires the computation of eight terms while in the pointwise requires only one. The following equation from Gooch [25] shows why the increasing number of terms happens for the successive derivatives:

$$\frac{\partial \phi_i}{\partial x^n \partial y^m} \widehat{x^n y^m}_{ij} = \frac{\partial \phi_i}{\partial x^n \partial y^m} \sum_{l=0}^m \sum_{k=0}^n \binom{m}{l} \binom{n}{k} (x_j - x_i)^k (y_j - y_i)^l \overline{x^{n-k} y^{m-l}} \quad (1.3)$$

To advance in time the mean values must be computed up to the desired spatial accuracy order. Therefore a conversion of the pointwise to mean values must be performed according to:

$$\bar{\phi}_i = \frac{1}{V_i} \int_{V_i} \phi_i dV = \phi_i + \frac{\partial \phi_i}{\partial x} \bar{x}_i + \frac{\partial \phi_i}{\partial y} \bar{y}_i + \frac{1}{2} \frac{\partial^2 \phi_i}{\partial x^2} \bar{x}_i^2 + \frac{\partial^2 \phi_i}{\partial xy} \bar{x}_i \bar{y}_i + \dots \quad (1.4)$$

This type of mean operator is only centred in the cell centroid. So, it only demands one inertia-related term per each spatial derivative, which allows the round-off error to be less significant, due to the lower number of algebraic calculations. Also purely-odd cell-centred inertial momentums are null. Subsequently, the previous equation can be further shortened:

$$\bar{\phi}_i = \frac{1}{V_i} \int_{V_i} \phi_i dV = \phi_i + \frac{1}{2} \frac{\partial^2 \phi_i}{\partial x^2} \bar{x}_i^2 + \frac{\partial^2 \phi_i}{\partial xy} \bar{x}_i \bar{y}_i + \frac{\partial^2 \phi_i}{\partial y^2} \bar{y}_i^2 \quad (1.5)$$

which is the formula used for the fourth-order schemes.

The proposed schemes in the pointwise framework imply computing an additional sparse matrix but requires a much smaller number of algebraic computations and it is easily scalable, which leads ultimately to a better accuracy.

1.5 Objectives

This thesis has the following purposes:

1. The development and implementation of a time operator based on the weighted least-squares, capable of maintaining accuracy orders up to eighth, for the 1D and 2D unsteady convection-diffusion equation, both with Cartesian and unstructured meshes.
2. Implementation of high-order temporal schemes (BDF) up to sixth-order accurate for 1D and 2D spaces.

To achieve these goals the following steps were followed:

1. Creation of a high-order cell-centred scheme based on the WLS method to develop an operator which can convert cell pointwise values to mean cell values.
2. Develop a time discretization using the Crank-Nicolson scheme to compute 1D unsteady convection-

diffusion equation and verify its overall order of accuracy.

3. Verification of the different spatial accuracy orders in the proposed conversion for both Dirichlet and Neumann boundary conditions.
4. Extending the mean operator and overall discretization to 2D unstructured grids.
5. Comparison among all spatial schemes and grid types regarding time and memory costs for different error levels.
6. Implementation of high-order temporal schemes and verification of its temporal accuracy order.
7. Comparison among all temporal schemes efficiency by analysing the Solver Runtime.

1.6 Present contributions

Vasconcelos (2017) [36] and Diogo (2019) [37] developed a 1D Matlab code, which was capable of solving the high-order steady convection-diffusion equation.

To solve unsteady cases, the Author created a 1D high-order mean operator, which converts pointwise values to mean ones, and built the overall unsteady global system which is solved in each timestep using Crank-Nicolson or the implicit time discretization. To obtain high-order temporal order, BDF schemes were developed up to sixth-order for uniform grids using Dirichlet BCs.

Analogously to 1D, the 2D Matlab code was adapted to unsteady cases by creating of a 2D high-order mean operator and implementing the Crank-Nicolson time scheme. Temporal and spatial orders were verified using Cartesian grids with both Dirichlet and Neumann BCs. The global system was adapted to high temporal order using BDF schemes.

In the SOL code, the 2D high-order mean operator was designed to unstructured grids and the global system of equations using CN time discretization was developed.

1.7 Thesis Outline

This Thesis is divided by four chapters. The first one includes the Thesis' motivation, a literature review of the state-of-art, the Author's objectives and present contributions.

The second chapter describes the proposed schemes of Finite Volume discretization to the unsteady convection-diffusion equations in the one dimensional space and explains all implementation details. It also shows the obtained results with 1D grids and the verification of its implementation by computing the accuracy order of the schemes.

The third chapter extends the previous schemes to the two dimensional space. By using the weighted least-squares to minimize local interpolation error when building the mean operator. The implementation on Matlab for Cartesian grids is presented as well as a comparison among different grid types on SOL code that used the unstructured grids framework.

Finally, in the fourth chapter conclusions regarding the work's results and future steps are stated and explained.

Chapter 2

High order transient schemes in One Dimensional Space

This chapter describes the proposed high-order schemes for unsteady transport equation in 1D space. First the construction of the temporal term operator will be explained, followed by the solution of the unsteady transport equation. Three types of time discretization will be compared: implicit Euler, Crank-Nicolson (CN) and Backward Differentiation Formulas (BDF). Afterwards error norms are explained and the obtained results are presented. Lastly a comparative study is shown concerning Solver Runtime (SRT) and Memory Storage (RAM) when combining high order spatial schemes with high-order temporal schemes.

2.1 Discretization of the unsteady convection-diffusion equation using Finite Volume Method

2.1.1 General Equation

The general transport equation considered is

$$\frac{\partial \phi}{\partial t} = \nabla \cdot (\Gamma \nabla \phi) - \nabla \cdot (U \phi) + \varphi_\phi \quad (2.1)$$

where ϕ is the transported unknown, φ_ϕ is the source term, U is the convection velocity and Γ is the diffusion coefficient.

Integrating both sides in time and space using an arbitrary Control Volume:

$$\underbrace{\int_t \int_{CV} \frac{\partial \phi}{\partial t} dV dt}_{\text{Transient Term}} = \int_t \int_{CV} \left(\underbrace{\nabla \cdot (\Gamma \nabla \phi)}_{\text{Diffusive Term}} - \underbrace{\nabla \cdot (U \phi)}_{\text{Convective Term}} + \underbrace{\varphi_\phi}_{\text{Source Term}} \right) dV dt \quad (2.2)$$

Since the convection and diffusion terms were explained in the work of Vasconcelos [36] and Diogo

[37], high-order convection-diffusion schemes are here presented as the matrix A and source term is indicated as S . Assuming the mean temporal derivative is available the following equation is obtained:

$$\int_t V_p \frac{\partial \bar{\phi}}{\partial t} dt = \int_t (A\phi + S) dt \quad (2.3)$$

where V_p is the cell's volume.

Integrating over time analytically on the left side and using generalized Crank-Nicholson on the right side, it leads to:

$$V_p(\bar{\phi}^{n+1} - \bar{\phi}^n) = \Delta t((1 - \alpha)(A\phi^n + S^n) + \alpha A(\phi^{n+1} + S^{n+1})) \quad (2.4)$$

where $\alpha = 0$ leads to an explicit Euler time discretization, $\alpha = 0.5$ leads to the Crank-Nicolson method and $\alpha = 1$ leads to an implicit Euler time discretization. Concerning 1D space CV will be reduce to cell's length and the transported values will be stored at the cells' centroids.

If pointwise to mean conversion in the transient term (left side of equation 2.4) is not performed the global discretization will revert to second order. Since the right side of the equation provides high order spatial results, the left side must not limit the spatial accuracy order. Therefore it is mandatory to create a mean operator as accurate as the order being studied (from fourth up to eighth order). This issue is the main focus of the next section.

2.2 Proposed Transient Schemes

2.2.1 Proposed Mean Operator

A cell-centred stencil should be as balanced and compact as possible because, in PDE, perturbations propagate over all directions.

Using the Taylor series formulation, a cell-centred reconstruction can provide a n-th order accurate (or n-1th exact) results:

$$\phi_C^R(x) = \phi_C + \frac{\partial \phi_C}{\partial x}(x - x_C) + \frac{1}{2!} \frac{\partial^2 \phi_C}{\partial x^2}(x - x_C)^2 + \dots + \frac{1}{(n-1)!} \frac{\partial^{n-1} \phi_C}{\partial x^{n-1}}(x - x_C)^{n-1} \quad (2.5)$$

The mean value at the cell's centroid can be computed by:

$$\bar{\phi}_C = \frac{1}{V_C} \int_{CV} \phi_C^R(x) dV \quad (2.6)$$

and using the integral and addition commutativity property, it leads to:

$$\bar{\phi}_C(x) = \frac{1}{V_C} (\phi_C V_C + \frac{\partial \phi_C}{\partial x} \bar{x} + \frac{\partial^2 \phi_C}{\partial x^2} \overline{x^2} + \dots + \frac{\partial^{n-1} \phi_C}{\partial x^{n-1}} \overline{x^{n-1}}) \quad (2.7)$$

where $\overline{x^n} = \frac{1}{n!} \int_{CV} (x - x_c)^n$

Therefore to build a point-to-mean value operator, it is important to compute the $n-1$ first order derivatives at the cell's centre and its corresponding inertial momentums. In 1D odd-number powered integrals can be reduced to zero as reconstruction is done around the cell's centre.

For each cell it is possible to obtain the local mean operator M_C , as the internal product of derivative's array P with inertial momentums array.

$$M_C = \frac{1}{V_C} P_{ij} \overline{x^j} \quad (2.8)$$

Local Polynomial Reconstructions

The locally polynomial reconstruction is done once for each cell of the domain (at initialization step). The local cell-centred polynomial reconstruction is given by 2.5 and the stencil is made up of its closest neighbour nodes. In 1D the total number of nodes is equal to the order of accuracy. To compute the required derivatives the following $D_C x = b$ system needs to be solved:

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & (x_{i-1} - x_C) & \frac{1}{2!}(x_{i-1} - x_C)^2 & \dots & \frac{1}{(n-1)!}(x_{i-1} - x_C)^{n-1} \\ 1 & (x_{i+1} - x_C) & \frac{1}{2!}(x_{i+1} - x_C)^2 & \dots & \frac{1}{(n-1)!}(x_{i+1} - x_C)^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & (x_{n_s} - x_C) & \frac{1}{2!}(x_{n_s} - x_C)^2 & \dots & \frac{1}{(n-1)!}(x_{n_s} - x_C)^{n-1} \end{bmatrix} \begin{bmatrix} \phi_C \\ \frac{\partial \phi_C}{\partial x} \\ \vdots \\ \frac{\partial^{n-1} \phi_C}{\partial x^{n-1}} \end{bmatrix} = \begin{bmatrix} \phi_C \\ \phi_{n-1} \\ \phi_{n+1} \\ \vdots \\ \phi_{n_s} \end{bmatrix} \quad (2.9)$$

Derivates can be computed in order to the solution inverting matrix D_C such as

$$P = D_C^{-1} \quad (2.10)$$

Since D_C is square and full its inverse can be computed through Gauss-Jordan Elimination without increasing memory storage. Matrix inversion was performed with Matlab function Inv.

Boundary Conditions

Stencils for cells close to the boundaries are extended towards inside the domain avoiding the creation of ghost points. To fully set the problem, solutions will be tested using full Dirichlet and mixed boundary conditions.

Full Dirichlet BC imposes the analytical solution values on both boundaries. When a boundary node belongs to the stencil a line must be added to matrix D_C such as:

$$\begin{bmatrix} 1 & (x_f - x_C) & \frac{1}{2!}(x_f - x_C)^2 & \dots & \frac{1}{(n-1)!}(x_f - x_C)^{n-1} \end{bmatrix} \begin{bmatrix} \phi_C \\ \frac{\partial \phi_C}{\partial x} \\ \vdots \\ \frac{\partial^{n-1} \phi_C}{\partial x^{n-1}} \end{bmatrix} = [\phi_f] \quad (2.11)$$

Mixed BC imposes Dirichlet BC on the right boundary node (as presented above in equation 2.11) and Neumann BC on the left side one i.e. imposes the analytical gradient value. Neumann BC implies adding a row to matrix D_C such as:

$$\begin{bmatrix} 0 & 1 & (x_f - x_C) & \dots & \frac{1}{(n-2)!}(x_f - x_C)^{n-2} \end{bmatrix} \begin{bmatrix} \phi_C \\ \frac{\partial \phi_C}{\partial x} \\ \vdots \\ \frac{\partial^{n-1} \phi_C}{\partial x^{n-1}} \end{bmatrix} = [\nabla \phi_f] \quad (2.12)$$

To fully set the problem Dirichlet and mixed conditions on boundaries will be used. The first sets $\phi = \phi_f$ and the second one sets the left boundary as $\phi = \phi_f$ and the boundary on the right to $\nabla \phi = \nabla \phi_f$.

Inertial Momentums

Inertial momentums up to $n - 1$ can be obtained computing the following integral:

$$\overline{x^n} = \frac{1}{n!} \int_{CV} (x - x_c)^n \quad (2.13)$$

and its discrete form by the Gauss quadrature is given by:

$$\overline{x^n} = \frac{1}{n!} \int_a^b (x - x_c)^n = \frac{1}{n!} \frac{b-a}{2} \sum_{k=1}^{n_G} w_k (x_k - x_c)^n \quad (2.14)$$

where n_G is the number of nodes within the cell and w_k its corresponding gauss weight.

The nodes coordinates x_k are computed by

$$x_k = a + \frac{(b-a)}{2} (1 + \zeta_k) \quad (2.15)$$

and the values of ζ and w used by the Gauss quadrature are in table 2.1. The values were taken from [56].

Accuracy Order	n_G	ζ_i	w_{G_i}
4th	2	$\pm \sqrt{\frac{1}{3}}$	1
6th	3	0 $\pm \sqrt{\frac{3}{5}}$	$\frac{8}{9}$ $\frac{5}{9}$
8th	4	$\pm \frac{\sqrt{525-70\sqrt{30}}}{35}$ $\pm \frac{\sqrt{525+70\sqrt{30}}}{35}$	$\frac{18+\sqrt{30}}{36}$ $\frac{18-\sqrt{30}}{36}$

Table 2.1: Coordinates and Gauss weights for a [-1,1] interval used in Gauss quadrature integration

Building Global Mean Operator

The values of each local M_C are then gathered in a global mean operator matrix, M . Values which multiply with cell centroids' values are stored in matrix M_{cells} . When a local cell-centred stencil includes a face boundary, the value which multiplies by the face boundary's value is stored in a Matrix M_f :

$$\bar{\phi}_{cells} = M_{cells} \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_{n_{cells}} \end{bmatrix} + M_f \begin{bmatrix} \phi_{f1} \\ \phi_{f2} \end{bmatrix} \quad (2.16)$$

2.2.2 Transient Transport Equation Discretization

Once the mean operator has been computed it can be implemented in the Transport Equation discretization (equation 2.4). This leads to the following global matrix equation:

$$V_p M_{cells} (\phi_{cells}^{n+1} - \phi_{cells}^n) + V_p M_f (\phi_f^{n+1} - \phi_f^n) = \Delta t \alpha (A_{cells} \phi_{cells}^{n+1} + A_f \phi_f^{n+1} + S^{n+1}) + \Delta t (1 - \alpha) (A_{cells} \phi_{cells}^n + A_f \phi_f^n + S^n) \quad (2.17)$$

Rearranging the last equation:

$$\underbrace{(V_p M_{cells} - \Delta t \alpha A_{cells})}_{\text{Global System Matrix}} \phi_{cells}^{n+1} = \underbrace{(V_p M_{cells} + \Delta t (1 - \alpha) A_{cells})}_{\text{Global Matrix for Explicit Multiplication}} \phi_{cells}^n + \Delta t (\alpha (A_f \phi_f^{n+1} + S^{n+1}) + (1 - \alpha) (A_f \phi_f^n + S^n)) - V_p M_f (\phi_f^{n+1} - \phi_f^n) \quad (2.18)$$

For each local mean operator the considered stencil is the union of the set of stencils used in the diffusion and convection schemes.

Therefore the convection-diffusion and mean operator matrices have the same bandwidth, which means that the Global System Matrix in equation 2.18 does not have a larger bandwidth than the previous matrices. If timestep is constant, only global matrices need to be stored.

2.3 Results and Discussion

In this section the accuracy orders of the mean operator and transient results will be verified as well as the temporal orders. Other features like implicit temporal discretization, non-uniform grids, Peclet number, runtime and memory costs will be analysed and discussed.

Two examples will be considered as solutions to the following transient transport equation:

$$\frac{\partial \phi_C}{\partial t} + U \frac{\partial \phi_C}{\partial x} = \Gamma \frac{\partial^2 \phi_C}{\partial x^2} + \varphi_\phi \quad U, \Gamma \in \mathbb{R} \quad (2.19)$$

Using as solutions:

Example 1 - the exponential dampening case (with artificial source term):

$$\phi(x, t) = e^{2x-t}, \text{ with source term } \varphi_\phi = (-1 + 2U - 4\Gamma)e^{2x-t} \quad (2.20)$$

Example 2 - the classical dampening Travelling Wave (without artificial source term):

$$\phi(x, t) = e^{\beta x + (\Gamma(\beta^2 - \gamma^2) - \beta U)t} \sin(\gamma(x + (\beta\Gamma - U)t)), \beta, \gamma \in \mathbb{R} \quad (2.21)$$

The present results were concluded by setting setting $Pe_L = 1$ ($U/\Gamma = 1$), $\gamma = \pi$ and $\beta = 2$. Two types of BC were studied: full Dirichlet BC (blue lines) and Mixed BC (red lines).

In order to effectively verify the code, the order of pointwise to mean conversion is computed, followed by temporal scheme order and finally the spatial scheme order.

2.3.1 Error Assessment

In order to evaluate the pointwise to mean value conversion using the mean operator, errors are computed comparing the numerical mean value with the analytical mean value such that:

$$e_{mv} = \bar{\phi}_{numerical} - \bar{\phi}_{exact} \quad (2.22)$$

The numerical error is a sum of discretization scheme error, round-off error and iterative error.

Round-off error comes from computer limitations at the binary conversion of values in bytes. In this present Thesis numerical values are rounded to double precision.

Discretization error, is proportional to scheme's order of accuracy such that

$$e = \phi_{numerical} - \phi_{exact} = O(\Delta x^n, \Delta t^p) \quad (2.23)$$

where ϕ is the cell value, Δx is the cell's length, Δt is the timestep size, n is the spatial accuracy order and p is the temporal accuracy order.

Pointwise to mean value conversion and discretization scheme error are estimated using the mean relative error or the norm 1 of the relative error,

$$\|e\|_1 = \frac{(\sum_{k=1}^{n_{cells}} |e_k|) / n_{cells}}{\max(|\phi_{exact}|)} \quad (2.24)$$

Global scheme discretization error is composed by temporal error and spatial error. Temporal error is dominant if the size of the timestep is not small enough. Temporal accuracy order can be calculated as the slope of the graph curves such that

$$O_p = \frac{\log_{10}(\|e\|_{k+1}) - \log_{10}(\|e\|_k)}{\log_{10}(\Delta t_{k+1}) - \log_{10}(\Delta t_k)} \quad (2.25)$$

where k is referred to the considered mesh.

In this Thesis the global error is considered to be dominated by the spatial error, $\|e_{sp}\|_1$, when a timestep refinement leads to less than one percent decrease in global discretization error i.e. saturation is achieved. Due to this, the spatial error can be obtained by the formula:

$$\|e_{sp}\|_1 = \lim_{\Delta t \rightarrow 0} \|e\|_1 \quad (2.26)$$

and the spatial accuracy order is given by:

$$O_n = \frac{\log_{10}(\|e_{sp}\|_{k+1}) - \log_{10}(\|e_{sp}\|_k)}{\log_{10}(\Delta x_{k+1}) - \log_{10}(\Delta x_k)} \quad (2.27)$$

The schemes' performances are also evaluated comparing Solver Runtime (SRT) and Number of Non-Zeros (NNZ). SRT concerns the time needed for the algorithm to obtain spatial error and NNZ concerns the number of non-zeros of the sparse global matrices i.e. information regarding memory storage.

2.3.2 Peclet Number

Peclet number is a dimensionless number. It is defined by the ratio between advection rate and diffusion rate such as:

$$Pe = \frac{|U|\delta x}{\Gamma} \quad (2.28)$$

where U is the convection velocity, δx is a characteristic length and Γ is the diffusion coefficient. The domain considered in all case studies is one meter (setting $\delta x = 1$) so it will be considered and adapted to the analytical solutions:

$$Pe_L = \frac{U}{\Gamma} \quad (2.29)$$

2.3.3 Pointwise to Mean Value Operator

As stated previously the pointwise on mean operator depends of local derivatives which can be affected by the spatial error. So they have to be computed with the same order as the convection and diffusion terms.

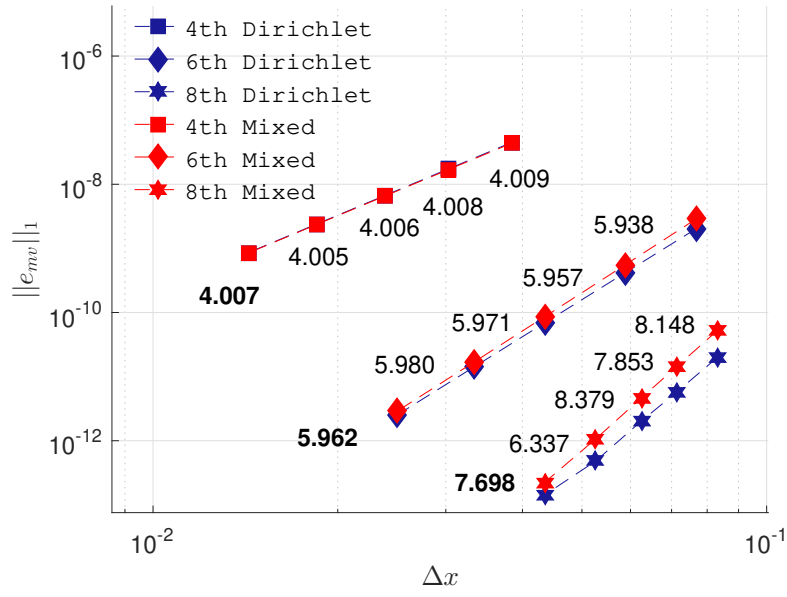
Therefore the following graphs 2.1 (a) and (b) show the mean operator's error (or pointwise to mean conversion error) decay over increasingly refined uniform meshes for the full Dirichlet and mixed BCs for both Dampening Exponential and Travelling Wave case studies. Spatial accuracy orders are presented between each mesh refinement.

Figures 2.1(a) and (b) verify the orders of the pointwise to mean operator for all spatial schemes. For the Dampening Exponential solution $\|e_{mv}\|_1$ decays with the expected orders in both full Dirichlet and mixed BCs. There is only overall a slight superconvergence behaviour when using the fourth and eighth-order scheme and subconvergent when using fourth-order ones. The results shown are very close to their theoretical orders.

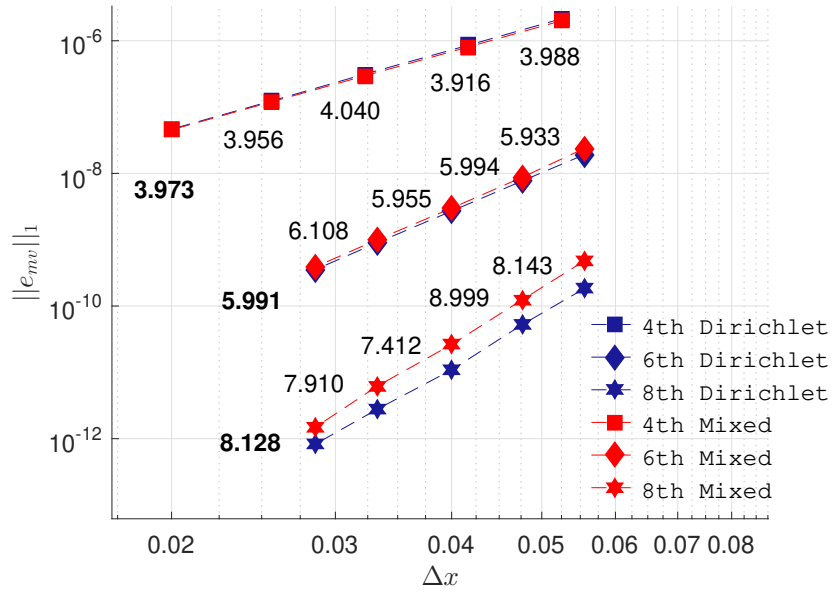
For the Travelling Wave case the fourth and sixth-order schemes show a slight subconvergent behaviour.

Oscillations when using the eight-order scheme for the most refined meshes may be due to round-off and iterative errors becoming dominant.

With these results, the mean value operator is prepared to be used in transient numerical cases.



(a) Dampening Exponential Analytical case



(b) Travelling Wave Analytical case

Figure 2.1: Mean Operator's spatial accuracy order for all schemes for the full Dirichlet and mixed BCs cases

2.3.4 Transient Results

When solving numerically unsteady PDEs the global error is the sum of the spatial and the temporal errors such that:

$$|e| = O(\Delta x^n, \Delta t^p) \quad (2.30)$$

where n is the spatial accuracy order and p is the temporal accuracy order.

So the goal is to increase the number of timesteps for the given grids until the global error stops decreasing, i.e., the global error is no longer dependent on temporal discretization and it is dominated by the spatial discretization.

For both solutions transient results are studied at halftime i.e. time needed for the highest value to be reduced to its half.

Temporal Order Verification for Crank-Nicolson discretization

In this section temporal discretization is done using Crank-Nicolson (second order accurate scheme). When timesteps are coarse the global error is expected to be dominated by temporal error discretization i.e. in this case decaying with Δt^2 over timestep refinements.

To verify the temporal order discretization scheme the following graphs 2.2(a) and (b) show the global error decay with timestep refinement for full Dirichlet and mixed BCs when using Crank-Nicolson time discretization and eighth-order spatial scheme. The temporal orders presented are obtained with the most refined meshes (23 cells for Dampening Exponential case and 35 cells for Travelling Wave case).

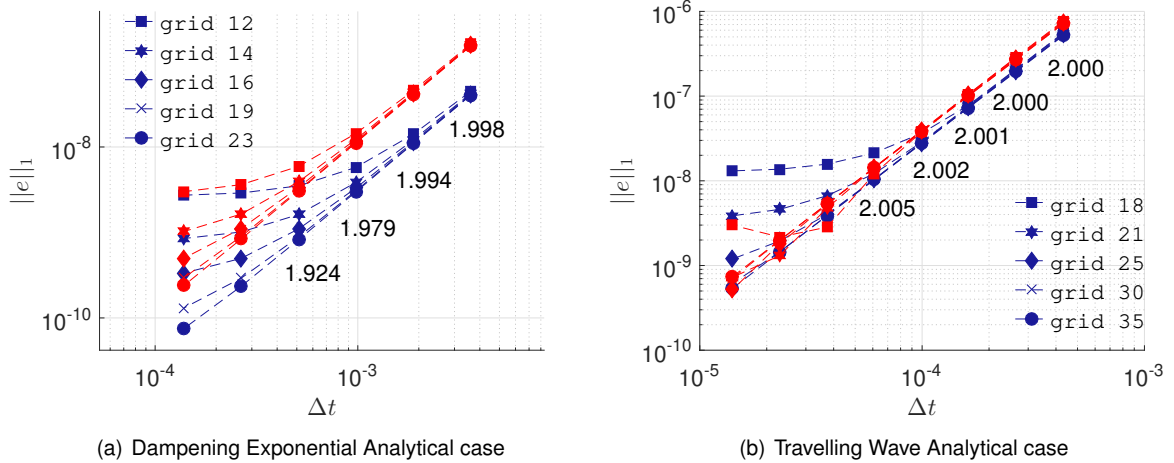


Figure 2.2: $\|e\|_1$ as a function of timesteps for eighth-order using full Dirichlet (blue lines) and mixed BC (red lines) and the Crank-Nicolson time discretization

Global error decays with second order with timestep refinements so temporal discretization is verified. The temporal order is more evident for coarse timesteps since it is when the global is more heavily dominated by the temporal discretization error and it is observed that the order tends to two when considering coarser timesteps for both analytical solutions.

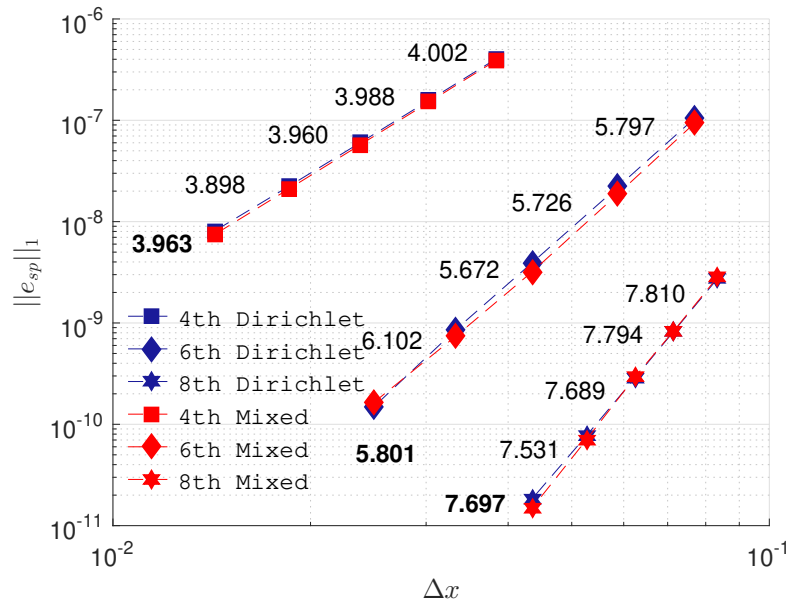
These graphs do not show convergence of global error to spatial error although it is noticeable that coarser meshes start being dominated by spatial error for smaller timesteps.

Spatial Order Verification for Transient Cases

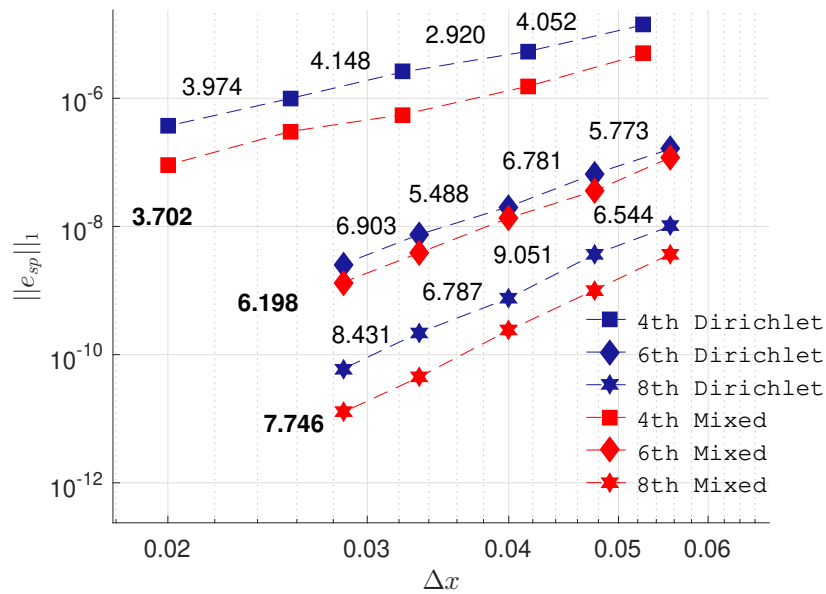
When the timesteps are small enough the global error is dominated by spatial discretization error (solely when the spatial error is orders of magnitude above the round-off and iterative errors). Obtaining spatial

errors for different grid sizes it is possible to verify the theoretical order of the transport equation. In this Thesis the spatial error is obtained when the global error decreases less than one percent with time refinement i.e. when global error was dominated by spatial error.

The following graphs 2.3 (a) and (b) show the spatial error decaying with the mesh refinement for all high order spatial schemes using full Dirichlet (blue lines) and mixed BCs (red lines) for both analytical solutions. The order between each mesh refinement presented concerns the results obtained when using full Dirichlet BC. Average spatial orders are written in bold and for both solutions the spatial errors were obtained at halftime.



(a) Dampening Exponential Analytical case



(b) Travelling Wave Analytical case

Figure 2.3: $\|e_{sp}\|_1$ as a function of grid size for all schemes using full Dirichlet and Mixed BCs and its spatial orders

For Dampening Exponential analytical case the obtained order is slightly subconvergent for all schemes. The numerical average orders are close to the theoretical ones. For this solution both full Dirichlet and mixed BCs provide the very similar results as the curves almost overlap each other. For the Travelling Wave analytical case curves show some oscillations but overall the average spatial orders are kept as expected.

2.3.5 Further Considerations

Implicit Temporal Discretization

In the present Thesis the pointwise to mean value conversion prevents any temporal computation without the use of matrices. Therefore it is inadequate for explicit schemes. Implicit Euler time discretization is first-order accurate so it demands more timesteps to compute spatial error for the same memory storage as the second-order Crank-Nicholson scheme. Its presence here only serves as a verification tool and it will not be further considered in 2D meshes. Figure 2.4 shows the norm-1 of global error decaying with the timestep size for eighth-order spatial scheme when using different meshes (with 12 cells up to 23 cells) for the Dampening Exponential analytical case.

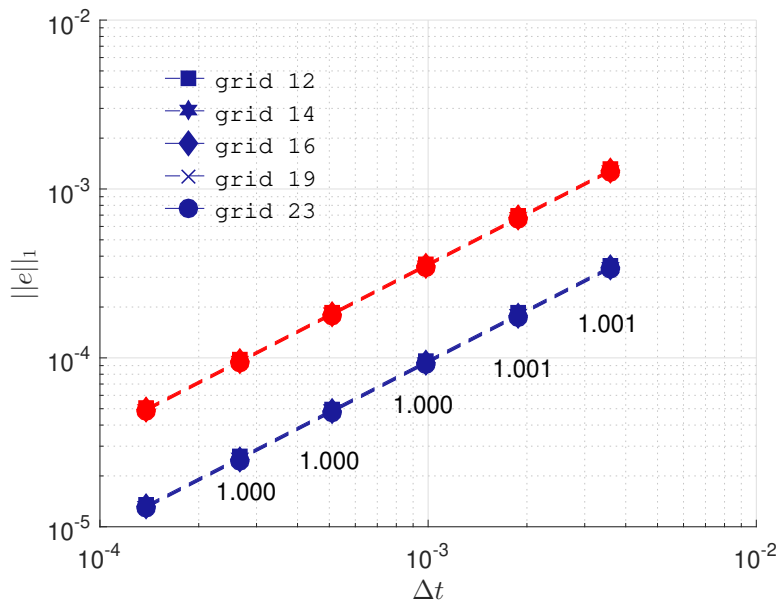


Figure 2.4: $\|e\|_1$ as a function of number of timesteps for Dampening Exponential Solution using implicit Euler time discretization ($\alpha = 1$) with full Dirichlet (blue lines) and mixed BC (red lines)

Since the results shown in Figure 2.4 were obtained when using timestep sizes for which the global error is dominated by the temporal error, so curves for different meshes overlap when using the same BC. The computed temporal orders are very accurate as implicit Euler is a first-order time scheme. Comparing these results with the same grids and timesteps considered when using CN (Figure 2.2(a)) $\|e\|_1$ values are E4 or E5 times higher when using implicit discretization showing therefore a much weaker accuracy.

Non-Uniform Grids

To test the current approach's robustness non-uniform grids with five percent perturbation of its vertices are considered in this case study. Figure 2.5 shows the results of spatial error decaying with grid size for the three spatial schemes at study for the Dampening Exponential case with non-uniform grids. Average spatial orders are written in bold.

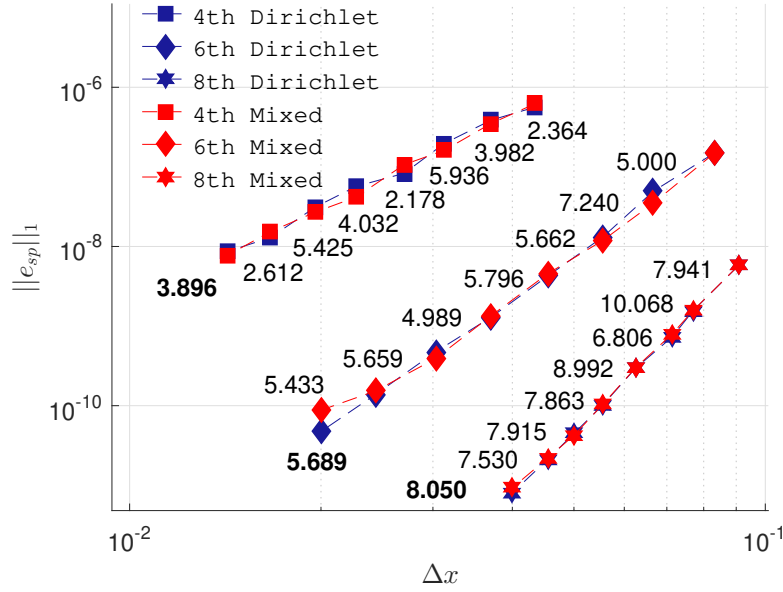


Figure 2.5: Spatial error for all schemes for Dampening Exponential analytical case using Non-Uniform grids with full Dirichlet (blue lines) and mixed BCs (red lines)

For this analytical case curves show some oscillation as meshes were generated randomly at each case instead of being successively refined. Therefore its local order is expected to be weaker. For that reason more meshes for each spatial order scheme were considered.

The average fourth and sixth spatial orders are slightly under the theoretical order and the average eighth-order is slightly above. Therefore spatial orders are successfully verified when using non-uniform meshes.

Spatial errors when using full Dirichlet and mixed BCs are very similar as it had been previously observed when using uniform meshes (figure 2.3).

Peclet Number

Peclet number may impact the stability of the convection scheme [57, 58]. Consequently it will be studied in this subsection. Since the Dampening Exponential analytical case has an artificial source term, the Travelling Wave solution will be used instead. The goal is to verify if expected high spatial order still holds true for a case with high Peclet number.

Figure 2.6 shows the spatial error decay with grid size for the Travelling Wave analytical case when setting $Pe = 100$ and $Pe = 10000$.

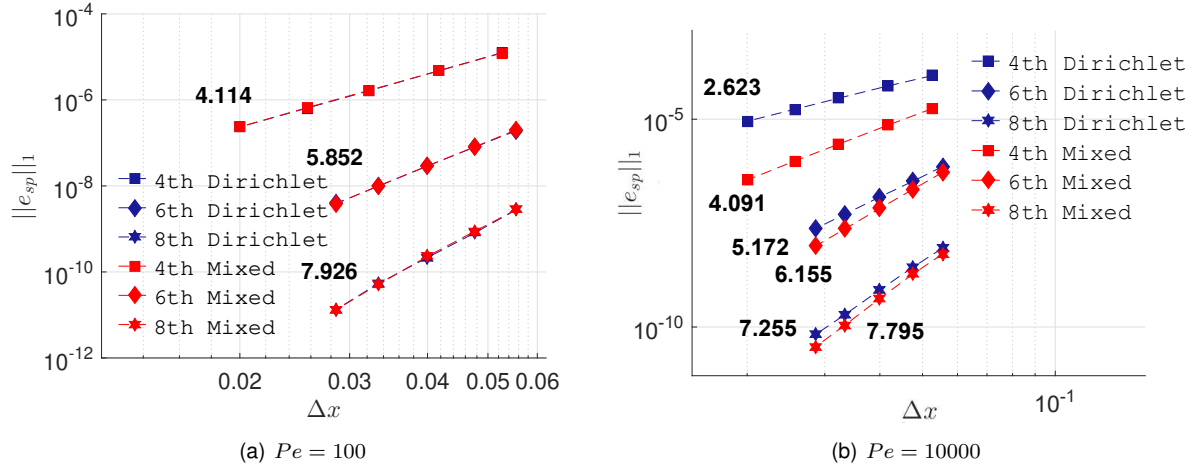


Figure 2.6: Spatial error decay with grid size for the Travelling Wave analytical case for all schemes setting $Pe = 100$ and $Pe = 10000$ using full Dirichlet and mixed BCs

For the Travelling Wave case when setting Peclet number to one hundred fourth-order scheme shows overconvergence and the sixth and eighth-order schemes subconverge. Numerical average order is close to expected theoretical one. In this case results for both BCs are very similar and curves overlap. When setting Peclet equal to one with mixed BCs provided smaller spatial errors than with full Dirichlet BC for the same grid (Figure 2.3 (b)).

When setting Peclet number to ten thousand conclusions are different from previous figures. When using mixed BCs fourth- and sixth-order schemes are overconvergent and the eighth-order scheme is subconvergent but close to theoretical order. On the other hand when using full Dirichlet BCs curves are linear not showing oscillations which is positive observation but the orders are lower than the theoretical ones.

Solver Runtime

This subsection focuses on runtime evaluation which is composed by preparation time (computing and building global matrices) and solver runtime, SRT. In this study only the latter will be considered since preparation time is negligible when compared to SRT.

Using CN the Global System Matrix is sparse. Since its inverse would be a full matrix MATLAB's GMRES solver was performed for each timestep.

Eighth-order global matrix will have more NNZ than the fourth-order one although spatial errors are expected to be lower for the latter. Therefore it is of interest assessing for a given mesh if eighth-order schemes's lower spatial error compensates the increase in NNZ and potentially in runtime. With that in mind figure 2.7 shows the spatial error decay with SRT. Average order for each spatial scheme is shown.

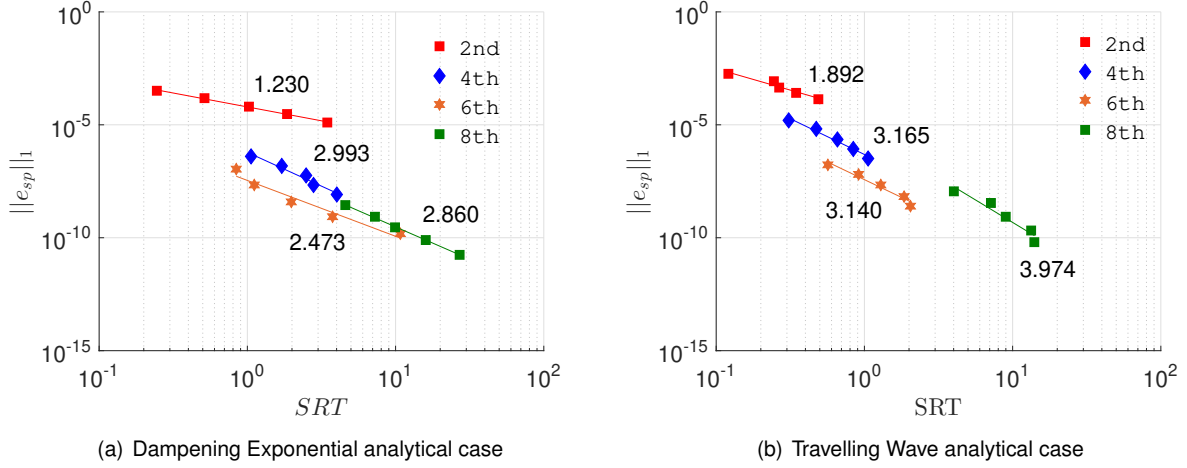


Figure 2.7: $\|e_{sp}\|_1$ as function of SRT for all schemes using CN

For both solutions sixth spatial order shows the optimal SRT since its curve guarantees a lower SRT value for a error level when compared to the other schemes. For the Dampening Exponential case sixth- and eighth-order scheme almost intersect for spatial error E-10 and since the eighth order scheme decays at a higher rate. In this case for errors smaller than E-10 eighth-order might outperform sixth-order but overall the eighth-order scheme underperforms as it needs more SRT for the solution to converge for a similar error magnitude when compared to the sixth order. Eighth-order scheme may need more timestep until global error is dominated by spatial error leading to higher SRT.

For a given spatial scheme, spatial error for Dampening Exponential case decays at a lower rate than the Travelling Wave one. The Dampening Exponential case demands computing an artificial source at time t^{n+1} for each timesteps and since higher spatial order schemes demands a higher number of timesteps for the global error to reach spatial error's value, which ultimately leads to higher values of SRT. This difference increase for the highest spatial order schemes.

For the Dampening Exponential case spatial error decays according to $O(SRT^{n/1.6260})$, $O(SRT^{n/1.3365})$, $O(SRT^{n/2.462})$ and $O(SRT^{n/2.7972})$ for second-, fourth-, sixth- and eighth-order spatial schemes and for the Travelling Wave case it decays $O(SRT^{n/1.057})$, $O(SRT^{n/1.2638})$, $O(SRT^{n/1.9355})$ and $O(SRT^{n/2.0131})$

Number of Non-Zeros

This subsection focuses on memory storage required for each spatial scheme. Using CN an global system matrix and a matrix for explicit multiplication must be stored as well as two face boundary matrices (one computes the convection-diffusion boundary faces sources and another computes the mean-value boundary face sources). Higher orders schemes lead to larger stencils and therefore more memory costs. The present study intends to evaluate if the compromise of memory storage is compensated by smaller spatial errors obtained when using higher order schemes. Figure 2.8 shows spatial error decay with the number of non-zeros (NNZ) stored for each spatial scheme.

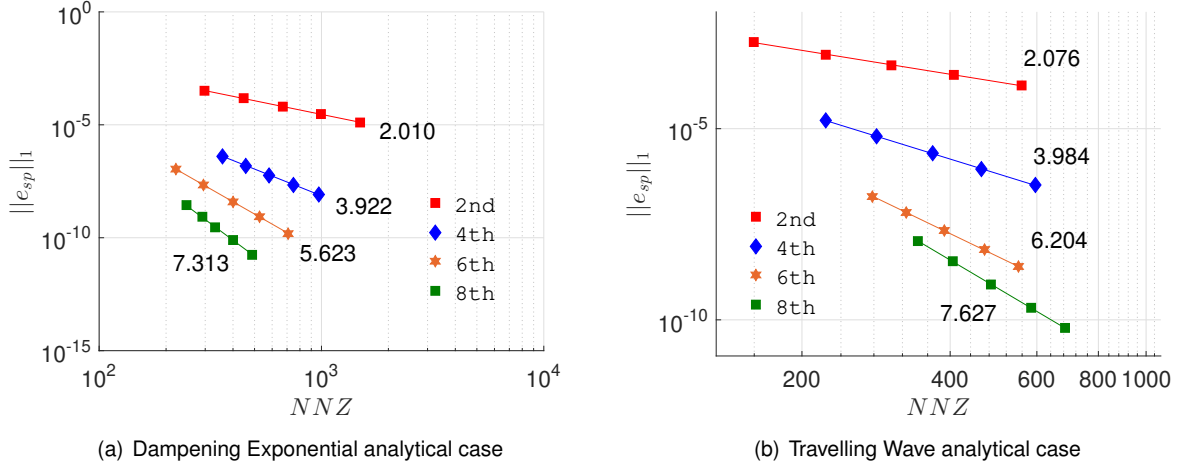


Figure 2.8: $\|e_{sp}\|_1$ as function of NNZ for all schemes using CN

For both Exponential Dampening and Travelling Wave solutions eighth-order provides the optimal memory storage. Therefore eighth spatial order should be interesting for cases where memory costs are a limiting specification but overall showed larger SRT than sixth order as it was concluded previously. Spatial error decays very similarly NNZ are larger for the Travelling Wave case as the meshes are more refined (grids are for consultation in legends of figure 2.2(a) and (b)). For both solutions spatial error decays on average with NNZ according to $\|e\|_1 = O(NNZ^n)$ for each spatial scheme.

2.4 High Order Temporal Schemes

Computing unsteady simulations requires high SRT costs and high-order temporal schemes can provide more efficient results than the traditional second-order time discretization. The global error is composed by the accumulation of each integral step error. High-order temporal schemes aim at decreasing the number of timesteps needed to achieve the same spatial error and consequently decreasing SRT.

Implicit BDF and SDIRK schemes have been proved very efficient for stiff initial-value problems. BDF will be implemented in this Thesis.

BDF schemes above sixth order are not zero-stable. The current thesis compares second-, fourth- and sixth-order schemes using the following list, respectively (BDF2, BDF4 and BDF6):

$$\bar{\phi}^{n+1} - \frac{4}{3}\bar{\phi}^n + \frac{1}{3}\bar{\phi}^{n-1} = \frac{2}{3}\Delta t(A\bar{\phi}^{n+1} + S^{n+1}) \quad (2.31)$$

$$\bar{\phi}^{n+1} - \frac{48}{25}\bar{\phi}^n + \frac{36}{25}\bar{\phi}^{n-1} - \frac{16}{25}\bar{\phi}^{n-2} + \frac{3}{25}\bar{\phi}^{n-3} = \frac{12}{25}\Delta t(A\bar{\phi}^{n+1} + S^{n+1}) \quad (2.32)$$

$$\bar{\phi}^{n+1} - \frac{360}{147}\bar{\phi}^n + \frac{450}{147}\bar{\phi}^{n-1} - \frac{400}{147}\bar{\phi}^{n-2} + \frac{225}{147}\bar{\phi}^{n-3} - \frac{72}{147}\bar{\phi}^{n-4} + \frac{10}{147}\bar{\phi}^{n-5} = \frac{60}{147}\Delta t(A\bar{\phi}^{n+1} + S^{n+1}) \quad (2.33)$$

The following subsection provides verification of the mentioned implicit BDF schemes as well comparisons regarding SRT and memory costs. Spatial orders will not be addressed since the spatial errors will converge to the ones presented in the previous results subsection since they are not affected by the time schemes.

It is worth mentioning the absence of results concerning memory costs in this subsection. The current study is based on assessing spatial error since it is dependent on its mesh refinement and temporal error can be decreased with timestep size. BDF2, BDF4 and BDF6 require respectively two, four and six time stages and the global matrices for all stages have the same NNZ showing that higher BDF schemes require more memory storage but in a negligible level. As its conclusion is straightforward, it will not be studied in this section.

2.4.1 Results and Discussion

Temporal Orders of BDF schemes

Global error is made up of temporal and spatial errors. While global error is dominated by temporal error it can decay with the temporal scheme's order i.e. Δt^p . The following graph concerns the verification of BDF's temporal orders. The finest mesh with the eighth spatial order was chosen in order to postpone global error's convergence and avoid dominance of the spatial error in the global one.

Figure 2.9 shows the global error decay with the timestep using a eighth-order scheme with 35 cells.

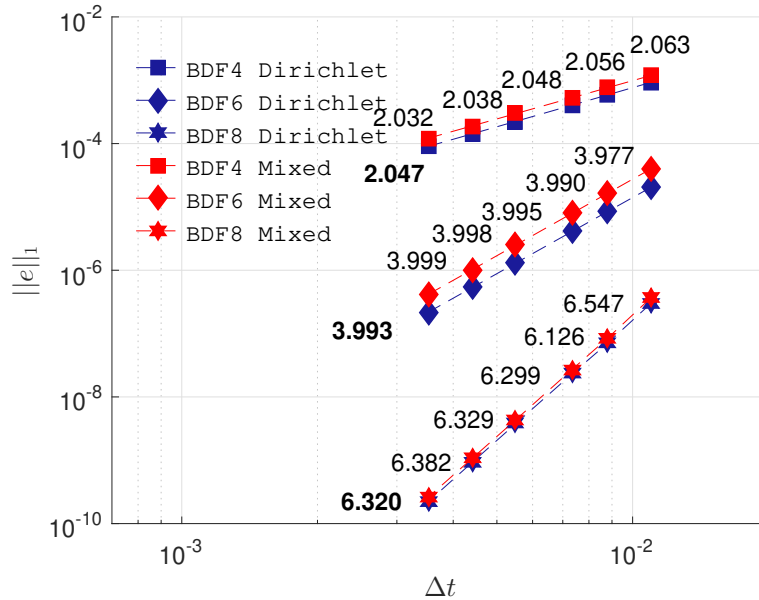


Figure 2.9: $\|e\|_1$ as function of timestep size for BDF2, BDF4 and BDF6 for Travelling Wave analytical solution using eighth spatial order scheme in uniform 1D mesh with 35 cells and using full Dirichlet and mixed BCs

Temporal scheme orders for BDF2, BDF4 and BDF6 are obtained as expected. BDF2 and BDF6

schemes are over convergent and BDF4 is subconvergent and numerical orders are very close to the theoretical ones. Using BDF6 curves for full Dirichlet and mixed BCs almost overlap while for BDF2 and BDF4 scheme full Dirichlet BC provides slightly lower global error than mixed BC.

Solver Runtime - SRT

By increasing the spatial order it would be expected advantageous increasing the temporal order as well since the global error is dominated by the lowest order scheme being used. Figures 2.10 (a) and (b) compare SRT results for the Travelling Wave analytical solution using second- and sixth-order meshes and all BDF temporal schemes considered. Average orders for each temporal scheme are written in bold.

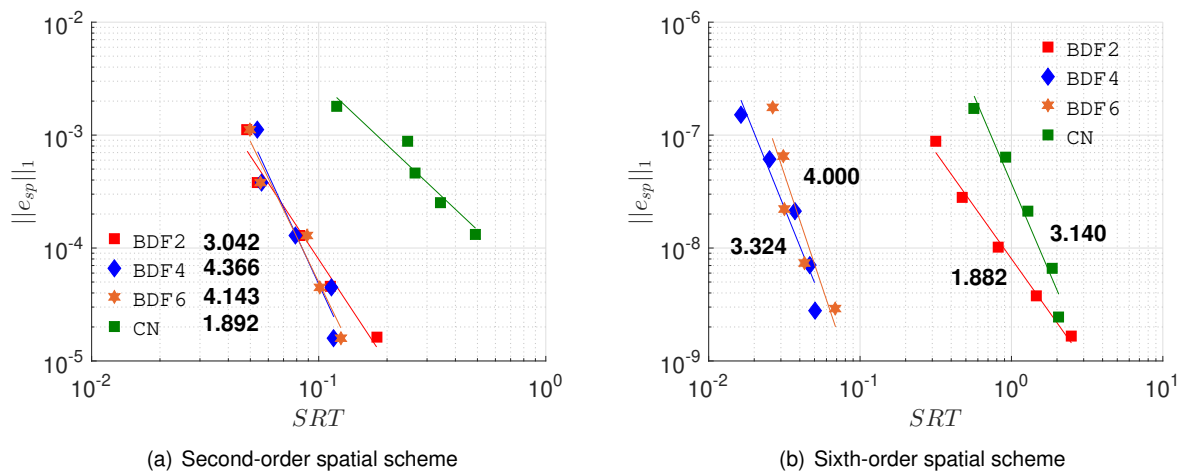


Figure 2.10: $\|e_{sp}\|_1$ as function of SRT for BDF2, BDF4 and BDF6 for Travelling Wave solution in uniform 1D mesh using full Dirichlet BC

Results show that using higher order BDF schemes for second-order spatial discretization improves only slightly the SRT and only for errors under E-4. On the other hand for sixth-order spatial discretization scheme higher order BDF schemes provide faster SRT values. For the same spatial error BDF4 presents SRT values very similar to BDF6 but much smaller than BDF2. On both cases CN underperforms regarding SRT when compared to the three BDF schemes. Table 2.2 compares SRT for two given norm-1 errors (E-7 and E-9) when using sixth-order spatial discretization.

$\ e\ _1$	BDF2		BDF4		BDF6		CN	
	SRT (s)	r_1	SRT (s)	r_1	SRT (s)	r_1	SRT (s)	r_1
E-7	0.2619	1	0.0202	0.0771	0.0259	0.099	0.7308	2.7904
E-9	3.0239	1	0.0809	0.0268	0.0820	0.0271	3.1680	1.0477

Table 2.2: The $\|e\|_1$ estimated for BDF2, BDF4, BDF6 and CN schemes and the respective ratios, using as reference the value from BDF2.

When using a sixth spatial order scheme From table 2.2 shows that for an error level of E-7 time savings go up to ninety percent when using BDF4 and BDF6 (when using BDF2 as reference). For a norm-1 error of E-9 time savings go up to ninety seven percent.

2.5 Conclusion

The temporal term on the transient convection-diffusion equation when integrated in space with the Finite Volume method is approximated to its mean time derivative value over the cell. If no mean operator is created, this term will simply be second-order accurate and the average numerical solution will also revert to this accuracy when using a pointwise framework. Therefore a pointwise to mean operator had to be created because the temporal term needs to be at least as spatially accurate as the convection and diffusion terms are. It is important that the temporal term does not limit the spatial order of the overall transient scheme.

In the present 1D transient studies when using fourth-, sixth- and eighth-order schemes were strongly verified both for Crank-Nicolson (second-order accurate) as well as Implicit discretization (first-order accurate). Due to the presence of a mean operator every timestep results are implicitly computed. So the traditional explicit discretization was not considered. Implicit discretization is as memory consuming as CN but since it is first-order accurate it requires more timesteps to obtain the global error when it is dominated by the spatial error, which leads to greater SRT.

Results obtained using non-uniform grids were less accurate than in uniform grids but the overall high-order tendency can be observed.

Results were also compared in Travelling Wave solutions for different convection-diffusion ratios (or Peclet numbers) and in those cases proved to have high spatial orders. Only when setting Peclet number to ten thousand the order was lowered than the theoretical one when using full Dirichlet BC.

Concerning the SRT when using CN time scheme, the sixth-order spatial scheme showed more efficient results. On the other hand regarding memory costs eighth-order proved to be more advantageous.

In order to reduce even further the Solver Runtime costs high-order temporal schemes were an option worth studying. BDF temporal schemes of orders two, four and six were designed and its accuracy orders were successfully verified. Combining them with second-order spatial discretization did not very yield fruitful. However when combining BDF4 with sixth spatial order schemes showed a much shorter SRT than when using BDF2. It showed considerable time savings. It shows that high order temporal schemes are suitable when paired with spatial high-order schemes. CN underperformed in all cases when compared to high-order BDF schemes.

Chapter 3

High order transient schemes in Two Dimensional Space

This chapter describes the adaptation of the mean operator and the transient convection-diffusion equation to a 2D domain and unstructured grids. When building the proposed operator in 2D the number of interpolation's coefficients should be enough to keep the spatial high-order of the overall transient transport equation. In order to minimize the local polynomial interpolation error weighted least-squares method (WLS) will be used, which is a flexible method that can handle an arbitrary number of neighbours of the unstructured grids.

Initially, using the Crank-Nicolson (CN) time discretization, accuracy spatial orders are verified for Cartesian meshes. Features like different convection velocities or Peclet number, SRT and memory storage are evaluated. It was also studied the influence different grid types have in the accuracy and if for the same scheme SRT and memory storage are influenced.

High temporal orders are achieved using high-order BDF schemes with Cartesian meshes and its impact on SRT is assessed.

3.1 Unstructured Grids

Unstructured grids are useful to compute flows along complex boundaries or when discretizing local grid refinements with smooth grid size. In this Thesis every domain cell has linear faces. When building the mean operator, the WLS is used to minimize the interpolation error the local polynomials using the correspondent values of the cell neighbours and from face boundaries when needed. All information needed regarding stencils is already provided by SOL code database. Some tests are performed using different grids at the end of this chapter in order to assess the operator's robustness. Reference length, H , is used as spatial parameter such that:

$$H = \sqrt{\frac{\text{Domain Area}}{n_{cells}}} \quad (3.1)$$

which in Cartesian grids reverts to the cell face's length and the discretized domain is a square of one per one (meter). It is analogous to Δx of the 1D space.

3.2 Proposed Schemes

The transient transport equation in a 2D domain is discretized similarly to 1D. Remembering that when discretizing the global equation with CN time discretization leads to:

$$(V_p M - 0.5 \Delta t A) \phi^{n+1} = (V_p M + 0.5 \Delta t A) \phi^n + 0.5 \Delta t (S^{n+1} + S^n) \quad (3.2)$$

where A is the matrix related to the convection and diffusion terms. In this pointwise framework mean operator only needs to be considered for high-order spatial schemes with an order higher than two.

3.2.1 Stencil

In unstructured grids the stencil becomes less straightforward than for Cartesian grids and plays therefore a more significant role. A more compact stencil leads to less memory storage and consequently less algebraic computations per timestep.

In the 2D case, the n -th spatial accuracy order local polynomial interpolation is extended to:

$$\phi_C^R(x, y) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} C_k (x - x_C)^i (y - y_C)^j \quad (3.3)$$

and analogously to 1D, the mean value is given by:

$$\bar{\phi}_C(x, y) = \frac{1}{V_C} \int_{CV} \phi dV = \frac{1}{V_C} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} C_k \overline{x^i y^j} \quad (3.4)$$

where $\overline{x^i y^j} = \int_{CV} (x - x_C)^i (y - y_C)^j$. The minimum number of polynomial terms for each spatial order n is $\frac{n(n+1)}{2}$, which is presented in table 3.1.

Accuracy Order	4	6	8
Number of Terms	10	21	36

Table 3.1: Number of interpolation terms demanded by each n -th spatial order for the 2D case.

To create a compact stencil the closest cells to cell C must be considered for the cell-centered reconstruction according to the desired order n . Firstly, the number of neighbours across both 2D directions (x and y) must be at least as large as for 1D reconstructions i.e. n cell neighbours in each coordinate. Secondly, to keep accuracy order stencil's size must be at least as large as the number of terms indicated in table 3.1.

When these requirements are not met the stencil must be extended towards within the domain (see Diogo [37]).

Therefore, using the face neighbours of cell C i.e. cells which share a face with cell C, it is possible to achieve second-order schemes. For fourth-order schemes first-order and second-order face neighbours of cell C are included in the stencil of the cell i.e. cells numbered with 1 and 2 in Figure 3.1. Analogously sixth-order schemes require face neighbours up to third-order and eighth-order schemes require neighbours up to fourth-order.

Figure 3.1 shows the stencils used in a local reconstruction for cell C up to eighth-order schemes. The eighth-order scheme uses a stencil which is composed by all the numbered cells in the figure.

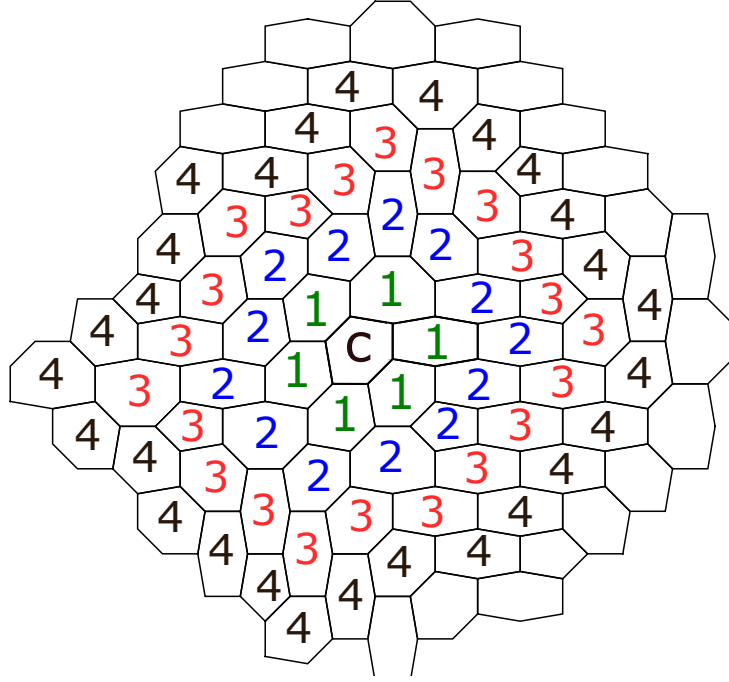


Figure 3.1: Face neighbours of cell C that are used to compute pointwise to mean value conversion operator

3.2.2 Local Polynomial Reconstructions

The n-th order local reconstruction around cell C using its face neighbours and the WLS method leads to the following matrix system:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 1 & (x_{i-1} - x_C) & (y_{i-1} - y_C) & (x_{i-1} - x_C)^2 & (x_{i-1} - x_C)(y_{i-1} - y_C) & \dots & (y_{i-1} - y_C)^{n-1} \\ 1 & (x_{i+1} - x_C) & (y_{i+1} - y_C) & (x_{i+1} - x_C)^2 & (x_{i+1} - x_C)(y_{i+1} - y_C) & \dots & (y_{i+1} - y_C)^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & (x_{n_s} - x_C) & (y_{n_s} - y_C) & (x_{n_s} - x_C)^2 & (x_{n_s} - x_C)(y_{n_s} - y_C) & \dots & (y_{n_s} - y_C)^{n-1} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ \vdots \\ C_{n-1} \end{bmatrix} = \begin{bmatrix} \phi_C \\ \phi_{n-1} \\ \phi_{n+1} \\ \vdots \\ \phi_{n_s} \end{bmatrix}$$

which may be represented as $D_C c_C = \phi_s$.

This is an overdetermined problem since the number of stencil neighbours is greater than the number of terms to be computed. So, to minimize the weighted residual, WLS is used.

In order to obtain an invertible square matrix on the left side, both sides of the equation are multiplied by $D_C^T W_C$:

$$D_C^T W_C D_C c_C = D_C^T W_C \phi_s \quad (3.5)$$

where W_C is a weight matrix that uses a weight function given by $(\frac{1}{d_C/\max(d_C)})^6$, which was optimized by Diogo [37] for both the convection and diffusion schemes.

On the left side of the equation there is now a square matrix with rows linearly independent. Therefore it is possible to invert it.

$$c_C = \underbrace{(D_C^T W_C D_C)^{-1}}_{W_{LS}} D_C^T W_C \phi_s \quad (3.6)$$

The matrix inversion is performed using Gauss Elimination Method. Starting with $[W_{LS}|I]$ an elimination from down to up is performed, obtaining an upper matrix on the left side. Then it is followed by an elimination from up to down leading to a diagonal matrix and finally obtaining $[I|W_{LS}^{-1}]$.

Local Polynomials coefficients can be stored in a the array P such that:

$$c_C = P \phi_s \quad (3.7)$$

for future reference.

3.2.3 Boundary Treatment

A stencil may have face boundaries and so BC must be applied in the local polynomial reconstruction. Dirichlet BC imposes the analytical value at the face boundaries (this contribution is represented in the first line of the equation 3.8). The Neumann BC is implemented as the dot product between the imposed gradient flux and the unitary face boundary's normal pointing outwards the domain (shown in the last line of the equation 3.8).

Equation 3.8 shows how the contribution of the Dirichlet and Neumann boundary conditions can be added to matrix D_C .

$$\begin{bmatrix} 1 & (x_f - x_C) & (y_f - y_C) & (x_f - x_C)^2 & (x_f - x_C)(y_f - y_C) & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ 0 & n_x & n_y & 2(x_{n_f} - x_C)n_x & (y_{n_f} - y_C)n_x + (x_{n_f} - x_C)n_y & \dots \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ \vdots \\ C_{n-1} \end{bmatrix} = \begin{bmatrix} \phi_f \\ \vdots \\ \nabla \phi_f \cdot \vec{n} \end{bmatrix} \quad (3.8)$$

Two sets of boundary condition were considered in this work: full Dirichlet BC is a set with all boundaries imposed with Dirichlet condition, and mixed BC is a case with a Neumann BC at the east boundary and the remaining ones have a Dirichlet condition.

3.2.4 Inertial Momentums

Similar to the 1D case the inertial momentum vector is computed using Gauss quadrature integration for the 2D space. As mentioned in equation 3.4 the 2D inertial momentums are given by:

$$\overline{x^i y^j} = \int_{CV} (x - x_C)^i (y - y_C)^j \quad (3.9)$$

When considering cells from unstructured grids, the polygon cells (with more than three faces) can be split into triangles which use the cell's centroid and the two vertexes of each face. Figure 3.2 shows an example of the points needed to perform a gauss quadrature integration over a polygonal cell.

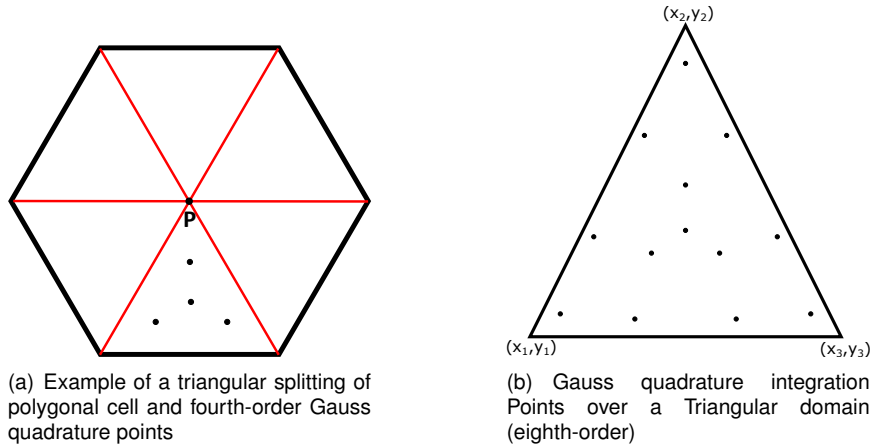


Figure 3.2: Examples of points used in the 2D Gauss quadrature integration [36]

Overall, the cell's inertial momentum is the sum of the Gauss quadrature integrations performed in each triangle.

$$\overline{x^i y^j} = \sum_{b \in J(C)} \left(S_{\Delta b} \sum_{k \in G(\Delta b)} w_{G(k)} (x_k - x_c)^i (y_k - y_c)^j \right) \quad (3.10)$$

where $J_{(C)}$ is the set of triangles, Δ_b represents each triangle.

Triangle S_Δ is computed by:

$$S_\Delta = (\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1) \quad (3.11)$$

In the 2D Cartesian plane the Gauss quadrature points over a triangle can be computed by equation 3.12 using the simplex coordinates (s_1, s_2 and s_3) which are listed in table 3.2 up to spatial order n.

$$x_k = \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \end{bmatrix} \quad (3.12)$$

where the Gauss quadrature weights (w_G) are listed in the table 3.2 according to each order:

n	N	s_1	s_2	s_3	w_{G_i}
4	1	0.3333333333333333	0.3333333333333333	0.3333333333333333	-0.5625
	3	0.6	0.2	0.2	0.5208333333333333
6	1	0.3333333333333333	0.3333333333333333	0.3333333333333333	0.225
	3	0.797426985353087	0.101286577323456	0.101286577323456	0.125939180544827
	3	0.059715871789770	0.470142064105115	0.470142064105115	0.132394152788506
8	1	0.3333333333333333	0.3333333333333333	0.3333333333333333	-0.14957004446767
	3	0.479308067841923	0.260345966079038	0.260345966079038	0.175615257433204
	3	0.869739794195568	0.065130102902216	0.065130102902216	0.053347235608839
	6	0.638444188569809	0.312865496004875	0.048690315425316	0.077113760890257

Table 3.2: Simplex coordinates and Gaussian weights necessary to perform Gauss Integration in a triangle, data taken from [59] and [60]

It is worth mentioning that purely odd inertial momentums such as \overline{x} , \overline{y} , $\overline{x^3}$, $\overline{y^3}$, ... are equal to zero. In triangular cells using eighth-order, it implies computing less 8 inertial moments (out of 36) per cell.

3.2.5 Transport Equation Discretization

Analogously to 1D the pointwise to mean conversion operator for 2D domains is computed such that:

$$\overline{\phi_C} = M_C \phi_s = P_{ij} \overline{x^i y^j} \phi_s \quad (3.13)$$

When designing the computational algorithm if a stencil includes a face boundary its polynomial coefficient is stored in a separate matrix, M_f .

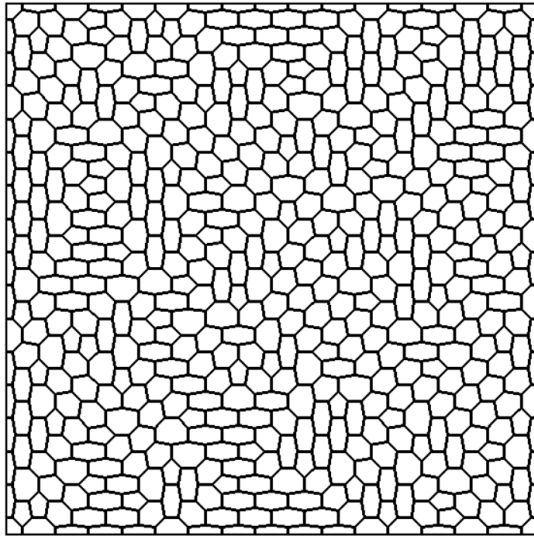
$$\overline{\phi_{cells}} = M_{cells} \phi_{cells} + M_f \phi_f \quad (3.14)$$

Combining global transport equation (see equation 3.2) with equation 3.14

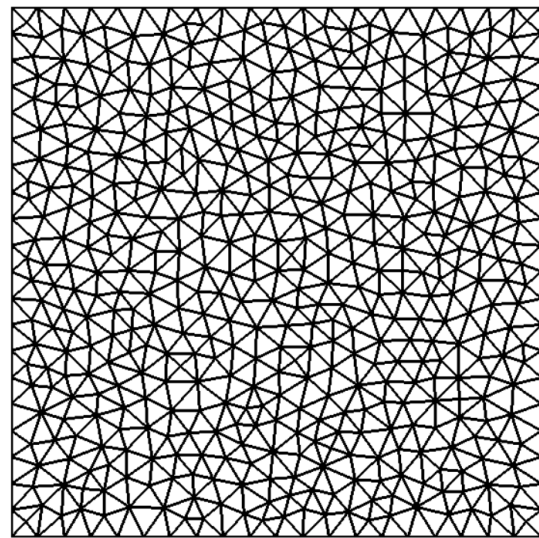
$$\begin{aligned} (V_p M_{cells} - 0.5 \Delta t A_{cells}) \phi_{cells}^{n+1} &= (V_p M_{cells} + 0.5 \Delta t A_{cells}) \phi_{cells}^n + \\ &+ M_f (\phi_f^n - \phi_f^{n+1}) + 0.5 \Delta t (A_f (\phi_f^n + \phi_f^{n+1}) + S^{n+1} + S^n) \end{aligned} \quad (3.15)$$

3.2.6 Unstructured Grids in 2D

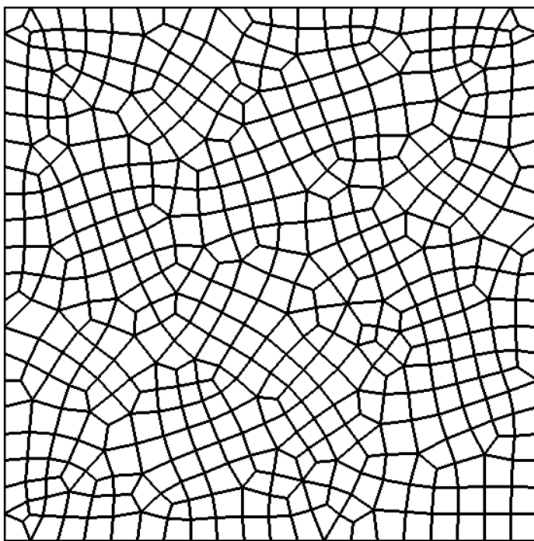
In order to study the applicability of the proposed schemes to unstructured grids. It was chosen a set of irregular meshes such as polyhedral (Figure 3.3(a)), triangular (Figure 3.3(b)), quadrilateral (Figure 3.3(c)) and a hybrid grids (Figure 3.3(d)) which mixes different types of grids and it is used as a final robustness test of the method. The first three sets of grids represent the irregular version of an available grid type. Grids were created changing the orientation of the cells. The size of each cell is not constant but overall the ratio was kept near to one.



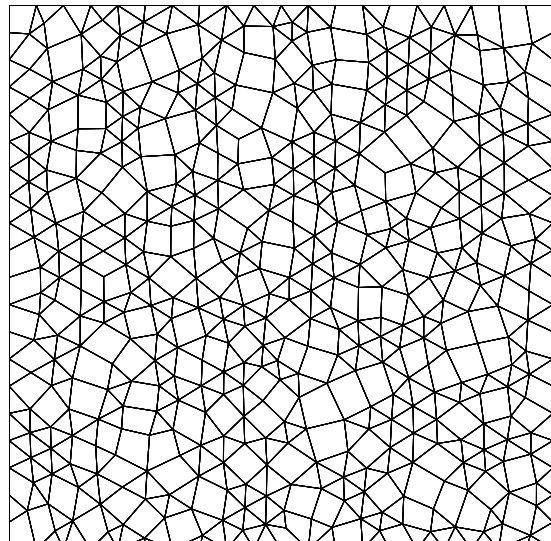
(a) Example of Polyhedral grid (with 545 cells).



(b) Example of Triangular grid (with 1016 cells).



(c) Example of quadrilateral grid (with 504 cells).



(d) Example of hybrid grid (with 726 cells).

Figure 3.3: Samples of irregular grid categories studied in this Thesis and which were also previously studied by Diogo [37] for steady cases.

3.3 Results and Discussion for Cartesian grids using Crank-Nicolson time discretization

In this section the accuracy order of the mean operator and the transient results with 2D domains will be verified as well as the temporal order. Both full Dirichlet (blue lines) and mixed (red lines) boundary conditions are compared alongside. Other features such as convection velocities, runtime and memory costs are also addressed and there is a unstructured grid types comparison at the end of the section.

Two examples are considered as analytical solutions to the following equation:

$$\frac{\partial \phi}{\partial t} + U_x \frac{\partial \phi}{\partial x} + U_y \frac{\partial \phi}{\partial y} = \Gamma_x \frac{\partial^2 \phi}{\partial x^2} + \Gamma_y \frac{\partial^2 \phi}{\partial y^2} + \varphi_\phi \text{ with } U_x, U_y, \Gamma_x, \Gamma_y \in \mathbb{R} \quad (3.16)$$

Example 1 - Dampening Cosine (with artificial source – $\varphi_\phi \neq 0$):

$$\begin{aligned} \phi(x, y, t) &= e^{-t} \cos(3\pi(x + y)), \\ \text{with source term } \varphi_\phi &= (-1 + 18\pi^2(\Gamma_x + \Gamma_y)) \cos(3\pi(x + y)) - 3\pi(U_x + U_y) \sin(3\pi(x + y)) \end{aligned} \quad (3.17)$$

Example 2: Classical Travelling Wave (without any artificial source – $\varphi_\phi = 0$):

$$\phi(x, y, t) = e^{-(\Gamma_x + \Gamma_y)kk^2\pi^2 t} \sin(kk\pi(x + y + (U_x + U_y)t)) \quad (3.18)$$

The present results were obtained by setting $U_x = 1$, $U_y = 1$, $\Gamma_x = 1$ and $\Gamma_y = 1$ and $kk = 1.2$.

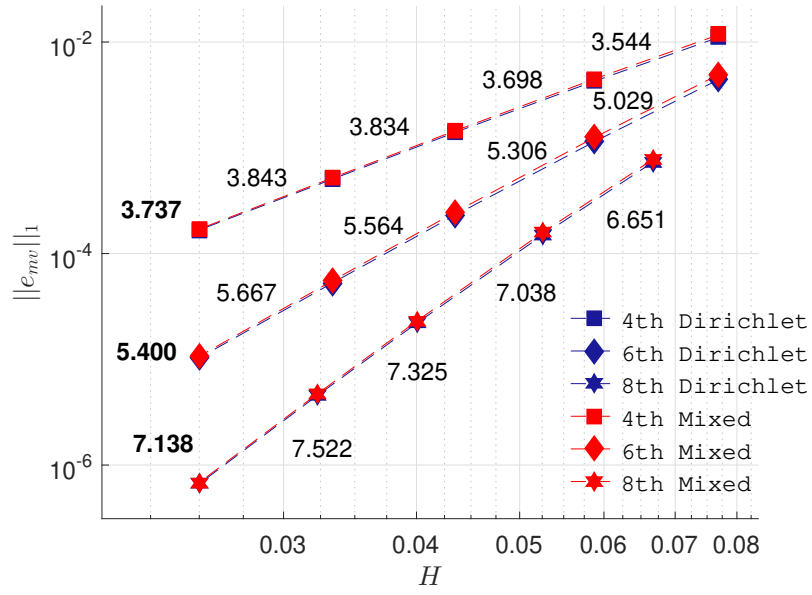
In this chapter 2D Dampening Cosine replaces the former 1D Dampening Exponential solution since for eighth-order the latter one reached the round-off error with very coarse grids which did not allow to verify clearly eighth-order accuracy.

Initially, the pointwise to mean conversion's accuracy order is verified, followed by the Crank-Nicolson temporal order verification and finally the spatial scheme order for Cartesian and unstructured grids.

3.3.1 Mean Operator

Pointwise to mean conversion must be as accurate as the spatial schemes used for the convection and diffusion schemes. Therefore it is important to verify the mean operator's spatial order before applying it to any transient problem. The mean operator error is quantified in similar fashion as previously described for the 1D case in chapter 2.

Figure 3.4 shows norm-1 of pointwise to mean value conversion using the mean operator in 2D space using a Cartesian grid. The average order of each spatial scheme is written in bold.



3.3.2 Temporal Discretization

In this section the temporal discretization was done using the Crank-Nicolson time scheme, which is second-order accurate. While the global error is dominated by the temporal error it must decay with the accuracy of the used time scheme.

Figure 3.5 shows the global error with the timestep. For small timesteps the global error tends to the spatial error. Eighth-order spatial scheme was used for these results. In figure's legend, a "Grid 40" means a mesh with 40 by 40 square cell distribution.

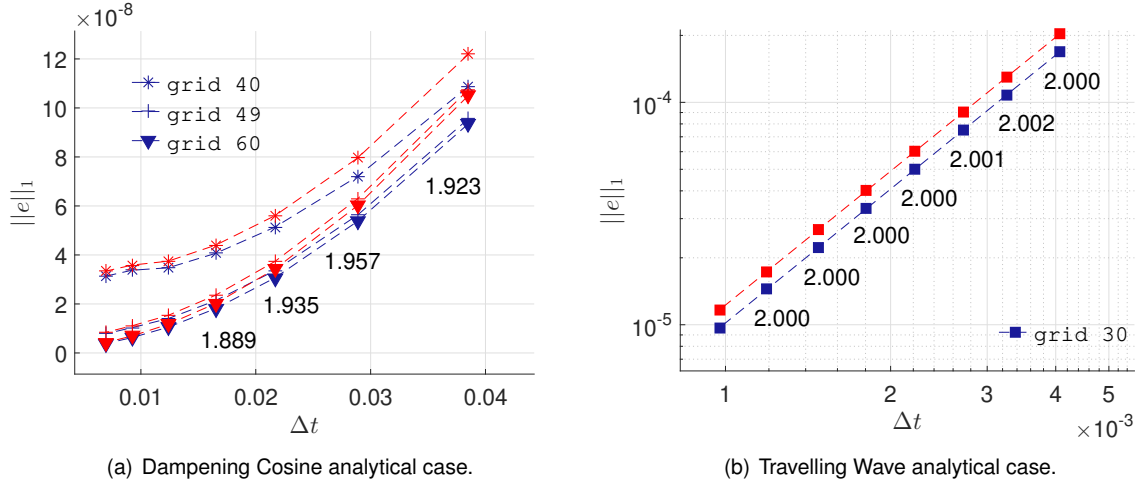


Figure 3.5: $\|e\|_1$ as a function of timestep for the eighth-order spatial scheme with Cartesian grids while using CN temporal scheme and for both full Dirichlet and mixed boundary conditions.

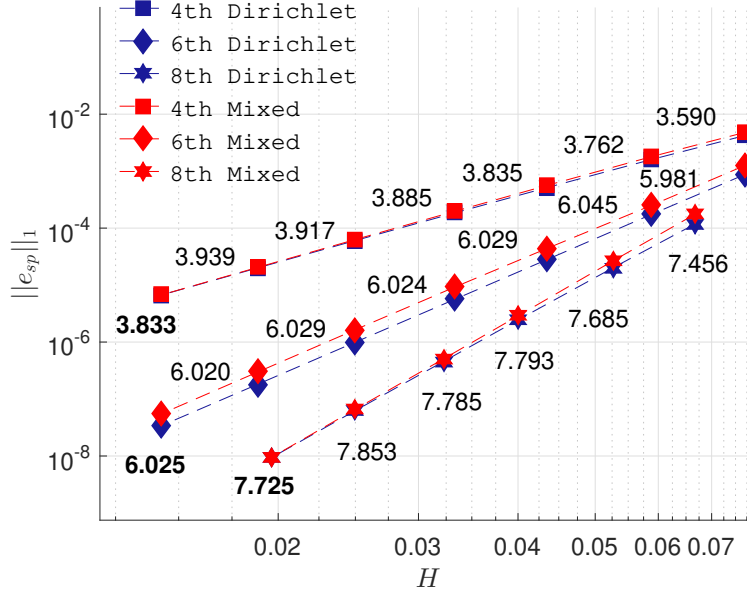
For both analytical solutions, the full Dirichlet BC case showed slightly smaller global errors than the mixed BC case. The computed order between timestep refinements is close to the theoretical value. Coarser timesteps show a tendency to decay at second-order. With the timestep when the curves start to flatten it shows that the global error started to be dominated by the spatial error before reaching round-off error and it is no longer dominated by the temporal one caused by Crank-Nicolson discretization. It can be observed for the Dampening Cosine solution that more refined meshes require smaller timesteps before the curve starts to flatten. Since its spatial error is smaller the temporal error must decrease even further before the spatial error becomes dominant. When the global error is still dominated by the temporal one different meshes should decay with similar values and that can be observed for the Dampening Cosine solution with 40x40 grid and above that the curves overlap for coarser timestep.

3.3.3 Spatial Discretization

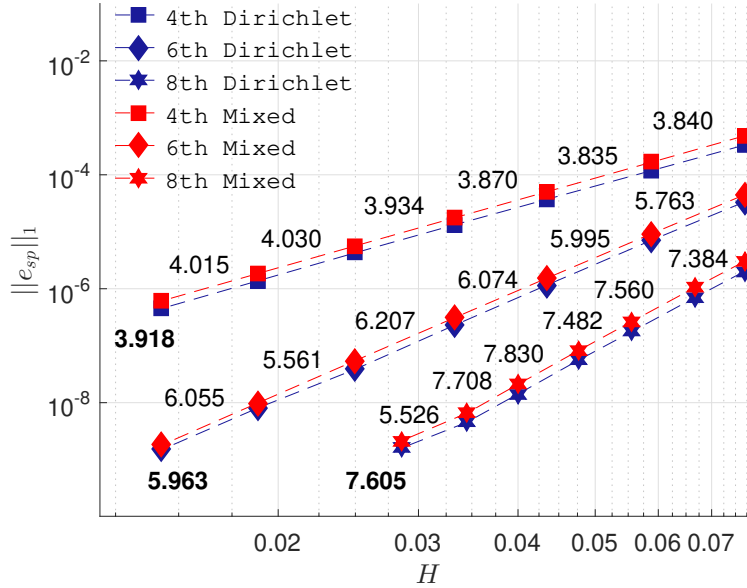
When the timestep is small enough that the global does not change in one percent spatial error is assumed to be obtained.

Figure 3.6 shows the spatial error decay with H (or cell's length in the Cartesian meshes) considered

for these results. The computed order between each successive refinement is shown as well as the average spatial order for each scheme, which is written in bold.



(a) Dampening Cosine analytical case



(b) Travelling Wave analytical case

Figure 3.6: $\|e_{sp}\|_1$ as a function of H with Cartesian grids for all spatial schemes using full Dirichlet and mixed BCs and with the computed order between each mesh refinement.

Only sixth-order scheme in the Dampening Cosine solution is slightly above the theoretical order. All other lines are slightly under its respective theoretical order. In the Dampening Cosine for the most refined meshes, orders 3.939, 6.020 and 7.853, were obtained, which are very close to their theoretical values (four, six and eight respectively).

The computed orders for the Travelling Wave case are also close to their theoretical values. The most

refined mesh refinement for the eighth order starts to lose its order due to round-off error.

3.3.4 Further Considerations

This subsection concerns the study of the performance of the proposed high-order schemes when setting high Peclet numbers and assessing if they provide for a given spatial error faster results (parameter SRT) than the traditional second-order schemes and using less memory storage (parameter NNZ). The results of this subsection were obtained with MATLAB code using Cartesian grids.

Peclet Number

When using full Dirichlet BC all boundary faces are setting to Dirichlet and when using mixed BC the east face is set as a Neumann boundary face while the other ones are set as Dirichlet.

In the 2D space, Peclet number may be defined in both directions as:

$$Pe_x = \frac{U_x}{\Gamma} \text{ and } Pe_y = \frac{U_y}{\Gamma} \quad (3.19)$$

using as reference length the domain's size ($L = 1$).

Figure 3.7 shows the spatial error decay with H for Travelling Wave solution when setting U_x and U_y to one hundred and U_x to one thousand. Full Dirichlet and mixed BCs were comparatively studied. In mixed case, Neumann condition was applied at the right side of the domain. Once again average spatial orders of the schemes are written in bold.

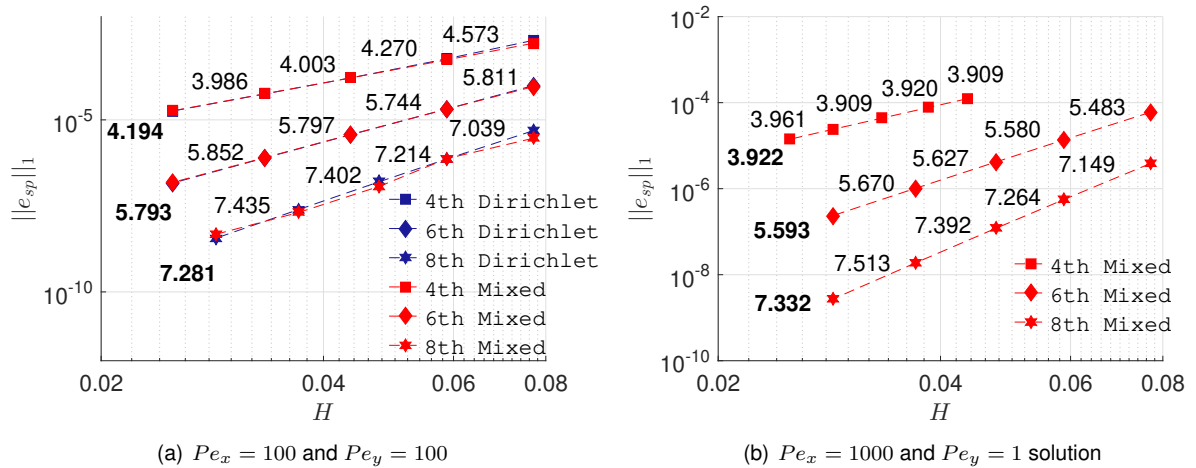


Figure 3.7: Spatial error for Travelling Wave analytical case for all schemes setting different values of convection velocities, U_x and U_y for the cases with full Dirichlet and mixed BCs

In the Travelling Wave solution by increasing the values of the Peclet number it will imply waves with a higher speed although its decay over time will be the same.

For low Peclet numbers ($Pe_x = 1$ and $Pe_y = 1$) presented in figure 3.6 (b) full Dirichlet BC provided slightly better spatial errors for a given mesh compared to the mixed BC counterpart. However, when

increasing Peclet number in both directions to one hundred both BC types provide very similar spatial errors indicating that Dirichlet boundaries except for the eighth-order case. When Pe_x is set to one thousand only mixed BC is able to converge while keeping the spatial order. For these meshes the use with only Dirichlet BC could not be converged for this velocities.

Solver Runtime

For the same mesh a higher order scheme uses larger stencils and so it will require a greater number of algebraic computations. This increases the solver Runtime (SRT) and the computational effort. Therefore studying the SRT is important to answer if this increase in SRT compensates the obtained smaller spatial errors of the high-order schemes.

Figure 3.8 shows the spatial error decay with SRT for both considered analytical solutions when using all spatial order schemes. Full Dirichlet BC was used.

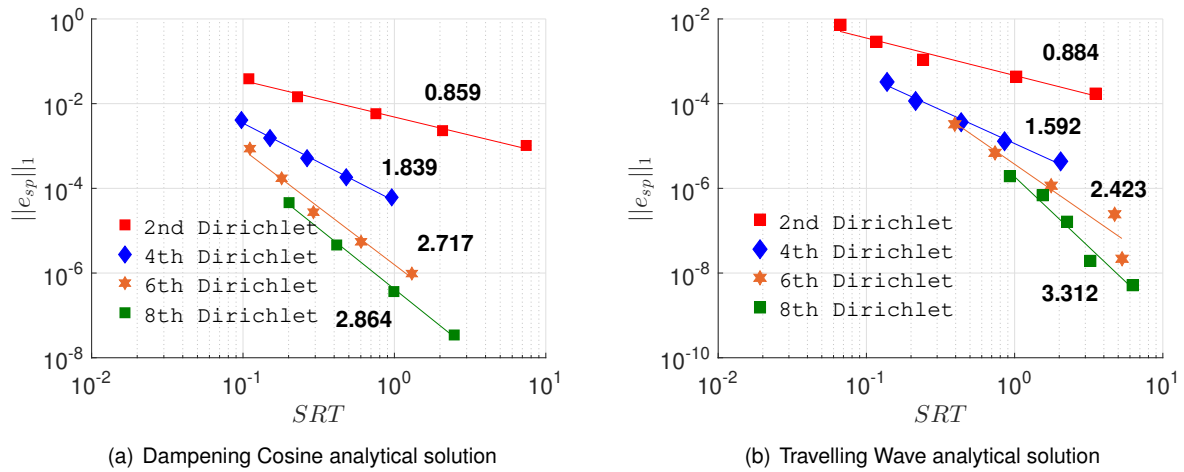


Figure 3.8: $\|e_{sp}\|_1$ as function of SRT for all schemes using CN and full Dirichlet BC

Error levels can be estimated by the intersection between the different curves of Figure 3.8. For both analytical solutions the high-order schemes provide more efficient results than the second-order scheme for errors smaller than $E-1$.

From Figure 3.8(a) in the Dampening Cosine solution the eighth-order scheme provides smaller values of SRT than the other high-order schemes.

In the Travelling Wave solution fourth-order schemes are optimal for error levels between $E-1$ and $E-4$, sixth-order for error levels between $E-4$ and $E-5$ and eighth-order schemes for smaller ones.

These results show the positive impact higher-order schemes can have in the computational cost of these solutions.

By making an average of the decay orders for the second, fourth, sixth and eighth-order spatial error decays at a rate of $O(SRT^{2/2.29})$, $O(SRT^{4/2.33})$, $O(SRT^{6/2.33})$ and $O(SRT^{8/2.59})$ respectively. So it may be estimated that $\|e_{sp}\|_1 = O(n/2.39)$, where n is the spatial order of the scheme.

Memory storage

If memory storage is a computational limiting factor, it is worth considering the impact of the scheme error decay with the Number of Non-Zeros (NNZ). The following study aims at finding the optimal memory storage for a given spatial error level in a similar fashion as it was done with SRT.

Figure 3.9 shows the spatial error decay with NNZ for both analytical solutions using the four spatial schemes and the CN time discretization. Only the case with full Dirichlet BC was considered here.

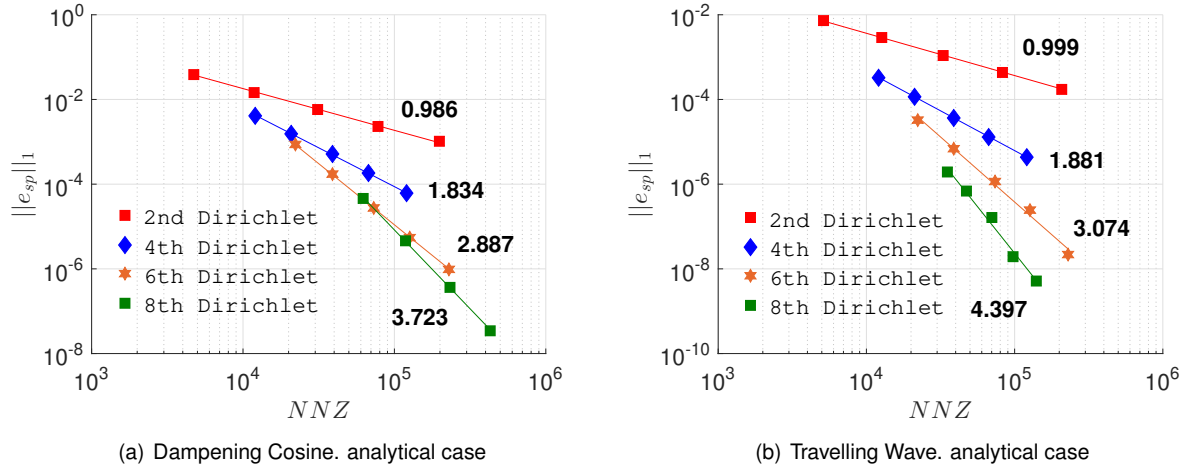


Figure 3.9: $\|e_{sp}\|_1$ as function of NNZ for all schemes using the CN time discretization and only Dirichlet BC.

In the Dampening Cosine solution higher-order schemes are useful for errors below E-1 regarding memory storage. The fourth-order scheme is optimal for spatial error levels between E-1 and E-3, sixth-order for error levels between E-3 and 5E-5 and eighth-order for levels smaller than 5E-5. In the Travelling Wave solution higher-order schemes are useful for errors below E-2 regarding memory storage. Fourth-order scheme is optimal for spatial error levels between E-2 and E-3, sixth-order for error levels between E-3 and E-4 and eighth-order for levels smaller than E-4.

Spatial error decay has a lower rate for the Dampening Cosine than for the Travelling Wave.

3.4 High Order temporal schemes using Cartesian grids

Matrices in 2D spaces have more NNZ for the same number of cells due to two dimensions when compared to the 1D case. As a consequence there is a larger amount of algebraic computations per timestep in 2D cases than in 1D ones. Implementing higher-order temporal schemes may allow a lower number of timesteps needed so that global error is only dominated by the spatial error i.e. improving computational time.

The following section aims at verifying high-order BDF schemes and assessing their impact in SRT when using each high-order spatial scheme. The BDF temporal schemes implemented in this chapter are analogous to the ones performed for the 1D cases. The second-order, fourth-order and sixth-order BDF schemes are given by equations 2.31, 2.32 and 2.33, respectively.

Results were obtained using Cartesian grids with the MATLAB code.

The study concerning memory costs was neglected since its results are straightforward. The stored global matrices are the same in all BDF schemes. BDF2 needs to store a vector containing ϕ^n and another for ϕ^{n-1} . BDF4 needs to store four vectors and BDF6 needs to store six for their respective time stages. Given this reason the study concerning memory costs was neglected when comparing time discretization schemes.

3.4.1 Temporal Orders

In order to verify the implemented time schemes this subsection verifies the temporal orders for all BDF schemes. When timestep is large enough, the global error is dominated by temporal error and should decay with the theoretical order of the used temporal scheme.

Figure 3.10 shows the global error decay with timestep for BDF2, BDF4 and BDF6 schemes. In all results of this section 3.4, only Dirichlet BC was considered.

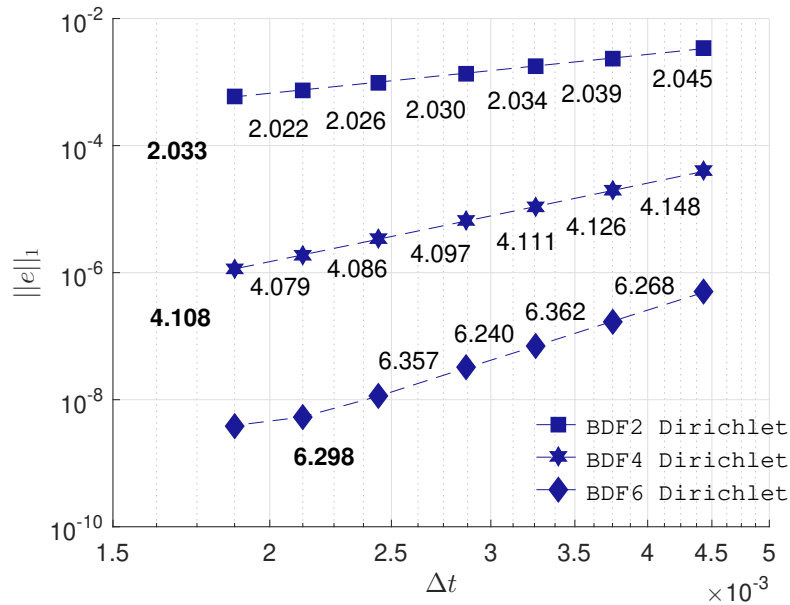


Figure 3.10: $\|e\|_1$ as function of the timestep for BDF2, BDF4 and BDF6 time schemes for Travelling Wave analytical solution using eighth-order scheme in 2D Cartesian mesh with 30 per 30 cells

The temporal orders are very close to the theoretical ones when using cartesian grids and an eighth-order spatial scheme. Temporal order should be read when timestep is coarser and the spatial error of this analytical case, grid with 30x30 cells and eighth-order scheme is around 4E-9 (see Figure 3.6(b)). BDF6 starts to flatten for the smallest timesteps as the spatial error's contribution starts to be relevant for the global error.

3.4.2 Solver Runtime

Once the temporal order of the BDF schemes are verified it is important to perform a comparative study for all spatial schemes using each BDF schemes. The goal is ultimately to assess how fast the high-order temporal schemes provide results.

Figure 3.11 from (a) to (d) show the spatial error decay for second, fourth, sixth and eighth order schemes, respectively, using the three BDF schemes introduced and the CN one. Results were obtained solving the Travelling Wave analytical case with Cartesian meshes and only Dirichlet BCs.

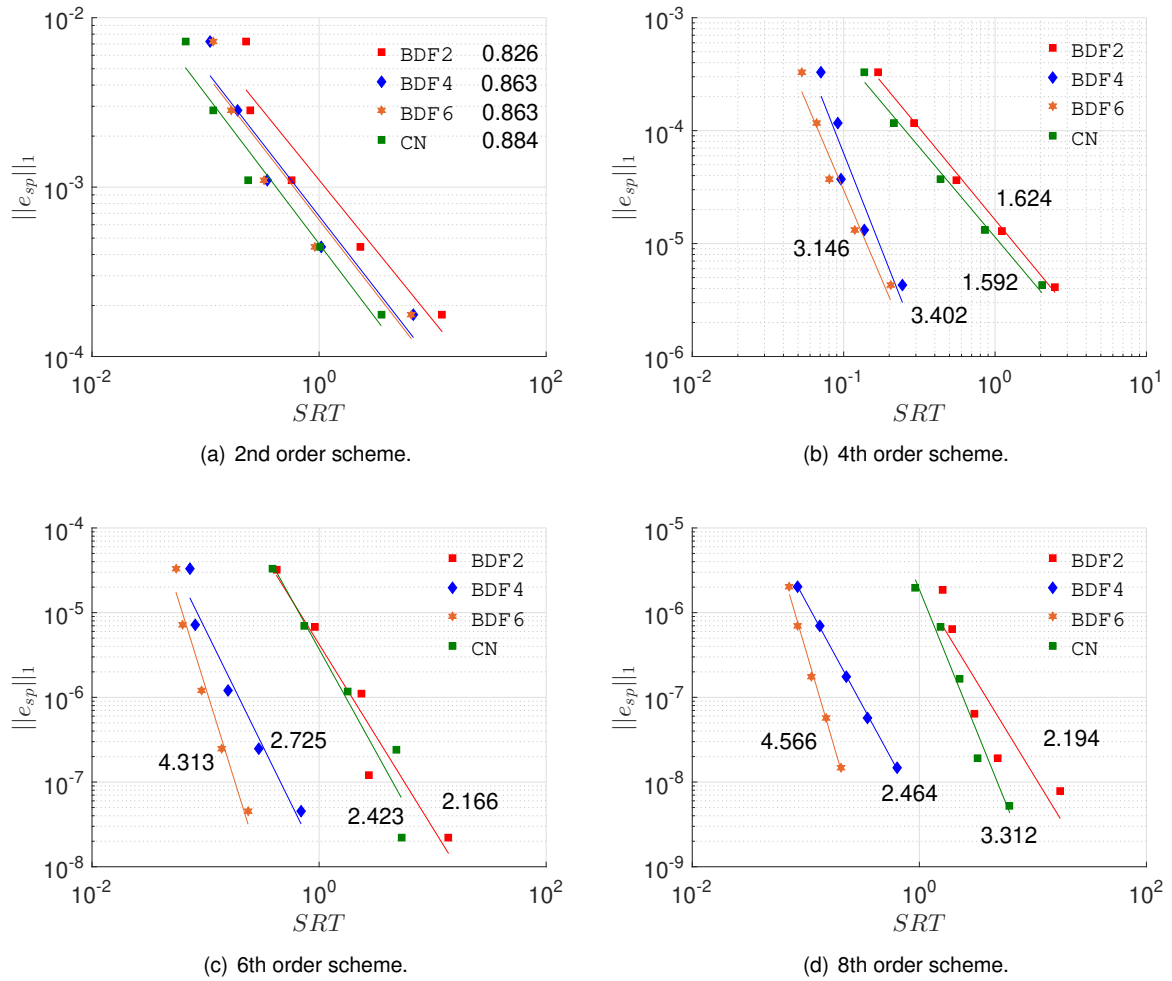


Figure 3.11: Spatial error evolution with SRT for all temporal and spatial schemes at study for Travelling Wave analytical case

These graphs show how high-order temporal schemes decrease SRT when using high order spatial schemes. For second-order spatial scheme CN provides slightly faster results and the high-order BDF schemes do not provide any relevant improvements when compared with BDF2. For the fourth order spatial scheme BDF4 already shows smaller SRT values than BDF2 and CN. Using BDF6 it leads to slightly faster results than with BDF4 but with a similar order. For both sixth and eighth spatial schemes BDF6 allows even faster results than the remaining schemes.

Overall all the high-order spatial schemes seem to have an optimal when the temporal scheme is as

high-order as possible. However, when the temporal order is lower than the spatial one, there is a noticeable underperformance in terms of efficiency regarding SRT.

For sixth-order schemes, for an error level of E-5 table 3.3 shows that BDF4 and BDF6 schemes reach time savings up to ninety percent when compared to BDF2 and for error levels of E-7 time savings reach ninety two percent.

$\ e\ _1$	BDF2		BDF4		BDF6		CN	
	$SRT (s)$	r_1	$SRT (s)$	r_1	$SRT (s)$	r_1	$SRT (s)$	r_1
E-5	0.6723	1	0.0843	0.1254	0.0624	0.0928	0.6642	0.998
E-7	5.6334	1	0.4567	0.0811	0.1816	0.0322	4.4429	0.7887

Table 3.3: SRT estimated for BDF2,BDF4, BDF6 and CN schemes for sixth-order scheme and the respective ratios, using as reference the value from BDF2.

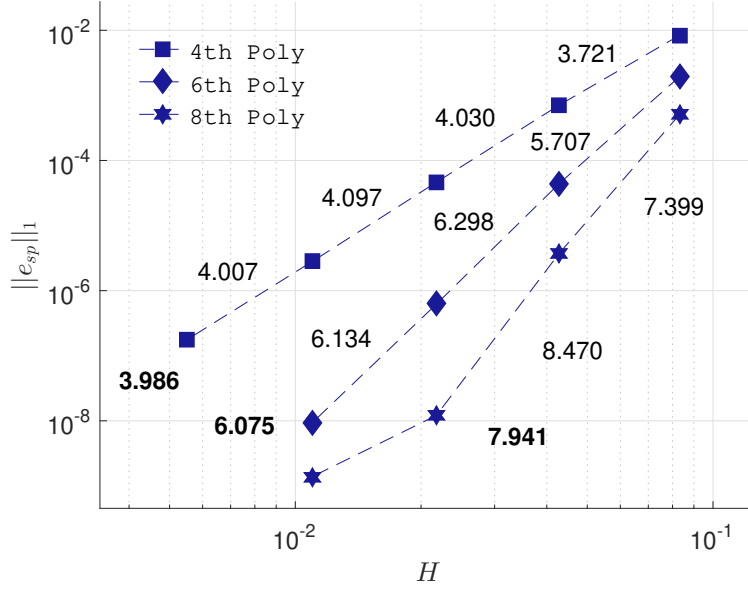
3.5 Results and Discussion for Unstructured Grids

This section concerns the quantification and analysis of spatial error decay with H , SRT and NNZ when using unstructured grids. Only irregular grids of polyhedral, triangular and quadratic meshes are considered as well as an hybrid mesh which is considered a robustness test due to its low grid quality. At the end it was performed a comparative studies regarding SRT and NNZ for different irregular meshes. These results for unstructured grids were obtained using the SOL code. Global system matrix and matrix for explicit multiplication were stored as MSR matrices, explained in AZTEC guide [61], GMRes as a linear solver algorithm was used and the spatial error was obtained when the global error decreased less than one percent with timestep size.

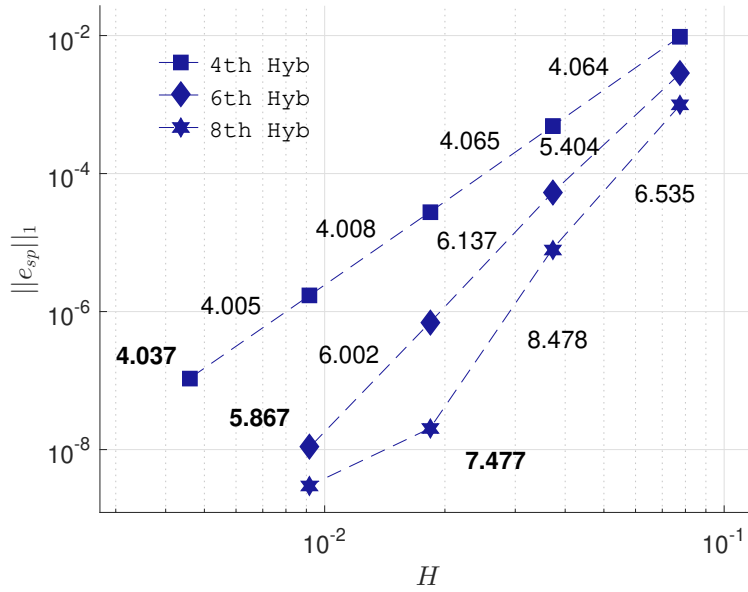
3.5.1 Spatial Error Orders

High spatial orders were verified by extending the SOL code to transient cases using CN time discretization. It is worth assessing if the proposed schemes are applicable to unstructured grids.

Figure 3.12 shows the spatial error decay with H for the Dampening Cosine analytical case when using irregular polyhedral (Figure 3.12(a)) and hybrid meshes (Figure 3.12(b)) for the three spatial schemes.



(a) Irregular Polyhedral mesh



(b) Hybrid mesh

Figure 3.12: $\|e_{sp}\|_1$ as function of H for the Dampening Cosine analytical case using irregular unstructured grids, the CN time scheme and full Dirichlet BC.

For the Dampening Cosine solution using an irregular polyhedral mesh curves show a very small overconvergent behaviour for the most refined meshes and the computed average orders are very close to the theoretical ones.

Using hybrid mesh curves display a slight overconvergent behaviour but the computed order for the last grid refinement of the fourth and sixth-order schemes has a very close value to the theoretical one.

From Figure 3.12, for both polyhedral and hybrid meshes the eighth-order schemes reach very small errors for coarse grids and the scheme for the most refined mesh is affected by round-off and iterative

errors. Due to this for the eighth-order scheme the computed order for the last grid refinement has not been considered.

3.5.2 Solver Runtime

This subsection evaluates the impact high-order schemes have on SRT when the global error is only dominated by the spatial one. This study aims at assessing which scheme order provides the fastest results for a given error level.

Figure 3.13 shows the spatial error decay with SRT when using irregular polyhedral and hybrid meshes for the second and three high-order schemes.

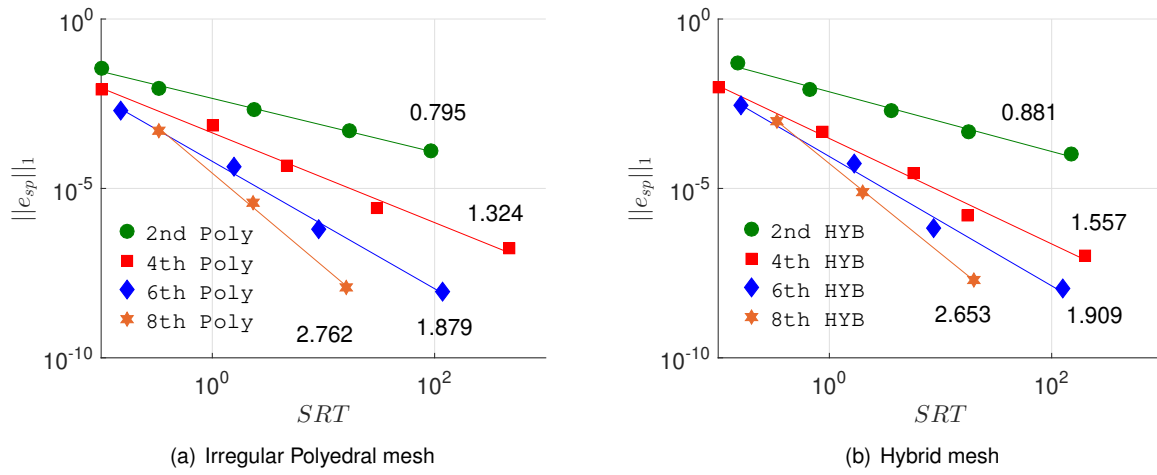


Figure 3.13: $\|e_{sp}\|_1$ as function of SRT for Dampening Cosine analytical case for the case with only Dirichlet BC.

Using an irregular polyhedral mesh with the sixth-order scheme provides faster results for spatial errors between E-3 and E-4 while the eighth-order is a better option for errors smaller than E-4.

When using a hybrid mesh the sixth-order scheme obtains faster results for errors between E-2 and E-4 when compared to the other schemes while eighth-order schemes provide faster results for smaller errors. From these results the following equation can be estimated: $\|e_{sp}\|_1 = O(SRT^{n/2.967})$.

3.5.3 Memory Storage

When the memory storage is a limiting factor, it is worthwhile studying how the spatial error decays with the total Number of Non-Zeros of the global matrices for each order.

Figure 3.14 shows the spatial error decay with the Number of Non-Zeros of the global matrices.

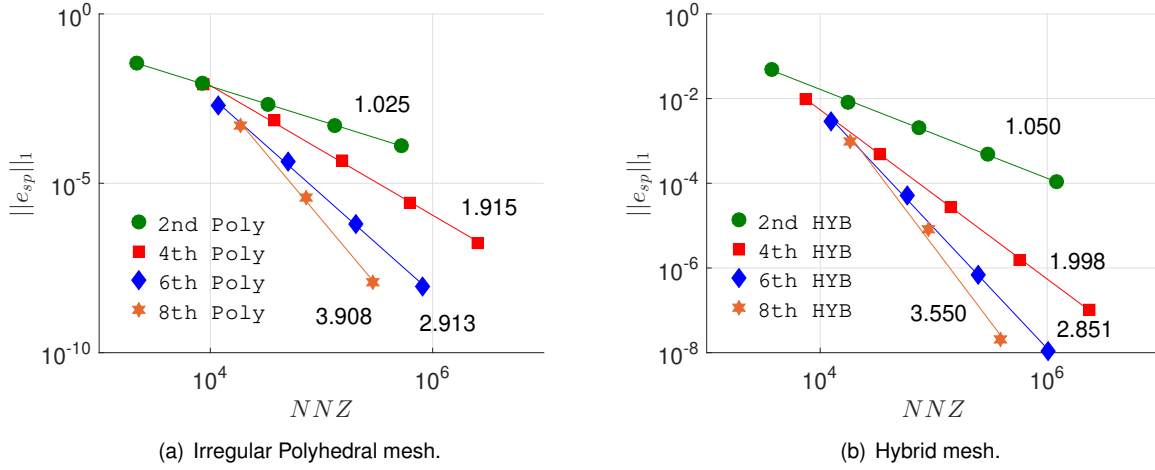


Figure 3.14: $\|e_{sp}\|_1$ as function of NNZ for Dampening Cosine analytical case using CN time scheme and full Dirichlet BC.

For the irregular polyhedral mesh fourth-order is optimal for errors greater than E-2 regarding memory costs, while sixth-order scheme is the most efficient one for error levels between E-2 and E-4. For errors smaller than E-4, the eighth-order schemes requires less memory than the other ones. For the hybrid mesh fourth-order schemes requires less memory for errors above E-2, sixth-order for errors between E-2 and E-3 and eighth-order is less memory costing for errors smaller than E-3. From these results the following equation $\|e_{sp}\|_1 = O(NNZ^{n/2})$ is verified.

Grid Topology

The following comparative study aims at evaluating how the four sets of selected irregular meshes perform regarding SRT and memory storage for the sixth-order scheme.

Figure 3.15 compares the spatial error evolution with SRT and memory storage when using sixth-order spatial schemes and for all irregular grids at study.

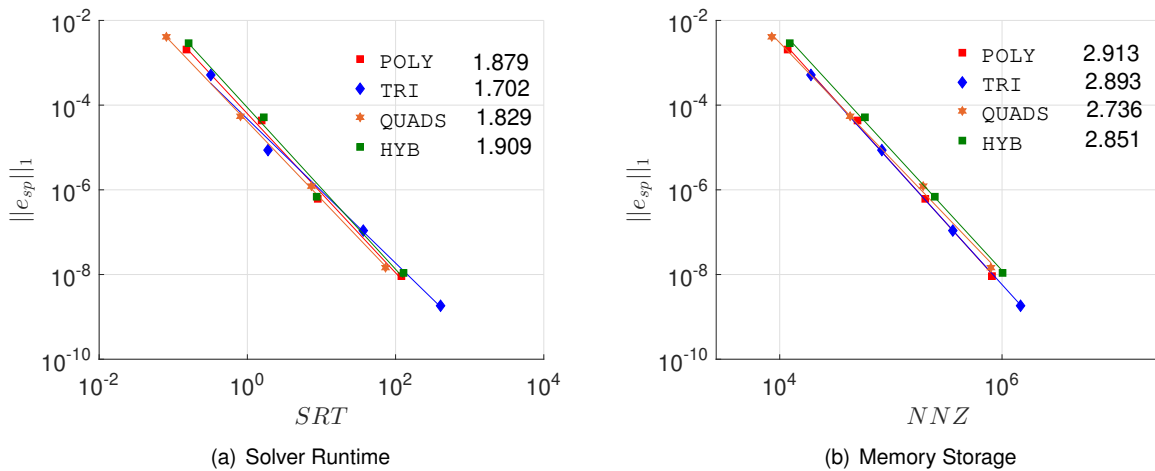


Figure 3.15: $\|e_{sp}\|_1$ as function of SRT and NNZ for sixth-order spatial scheme in the Dampening Cosine analytical case.

Regarding SRT all curves almost overlap each other but the irregular quadrilateral mesh shows time savings of ten percent when compared to triangular and polyhedral meshes and of thirty percent when compared to hybrid ones, from table 3.4.

Memory cost results for the irregular triangular mesh are slightly better than the remaining ones. From table 3.5 triangular meshes show memory savings of only three percent when compared to quadrilateral meshes.

The hybrid grid as expected can be considered the worst for both criteria (SRT and NNZ), although it has a similar order decay as the other irregular grids have.

$\ e\ _1$	POLY		TRI		QUADS		HYB	
	<i>SRT</i> (s)	r_1	<i>SRT</i> (s)	r_1	<i>SRT</i> (s)	r_1	<i>SRT</i> (s)	r_1
E-5	2.6593	0.8532	2.4879	0.7982	2.1437	0.6878	3.1167	1

Table 3.4: SRT estimated for a spatial error of E-5 with irregular polyhedral, triangular, quadrilateral and hybrid grids for sixth-order scheme and the respective ratios, using as reference the value from hybrid meshes.

$\ e\ _1$	POLY		TRI		QUADS		HYB	
	<i>NNZ</i>	r_1	<i>NNZ</i>	r_1	<i>NNZ</i>	r_1	<i>NNZ</i>	r_1
E-5	7.6626E4	0.8022	7.5991E4	0.7936	8.0205E4	0.8376	9.578E4	1

Table 3.5: NNZ estimated for spatial error of E-5 with irregular polyhedral, triangular, quadrilateral and hybrid grids for sixth-order scheme and the respective ratios, using as reference the value from hybrid meshes.

3.6 Conclusion

In the 2D space the local polynomial reconstructions lead to overdetermined matrices. So in order to minimise residual error, the weighted least-squares method was used. The pointwise to mean conversion orders were effectively verified for two analytical solutions before applying the method to transient problems.

The 2D unsteady cases were temporally discretized using Crank-Nicolson method. Numerical temporal orders were close to the theoretical ones. When the timestep is small enough, the global error starts to be dominated by the spatial one. The numerical spatial orders were close to each expected theoretical value.

Once the implemented schemes were successfully verified, the impact of the Peclet number on the accuracy was studied. For low Pe , full Dirichlet BC proved to be slightly more accurate. However, for high Pe the mixed BC case was able to keep the spatial order while the full Dirichlet counterpart did not converge.

In order to reduce SRT, high-order BDF schemes were considered for the Travelling Wave analytical case and their temporal orders were effectively verified. When comparing all BDF schemes using each spatial scheme, it was evident that the best results regarding SRT were achieved when BDF6 was used.

Afterwards spatial orders for all schemes were verified for irregular polyhedral and hybrid meshes for the Dampening Cosine analytical solution.

Regarding both SRT and memory storage case, high-order schemes proved to be advantageous for errors smaller than E-1 with both Cartesian and unstructured grids. Overall fourth-order schemes seem to be the best one for error levels between E-1 and E-3, the fourth-order for error levels between E-3 and E-5, while the eighth-order for even smaller errors.

With sixth-order scheme, irregular quadrilateral meshes showed the fastest SRT, with time savings of ten percent when compared to triangular and polyhedral meshes and thirty percent when compared to hybrid ones.

Chapter 4

Conclusion

4.1 Contributions

For the Finite Volume Method a mean operator, which converts pointwise values to mean ones, was created to keep the spatial high-order accuracy in the pointwise framework previously developed for the diffusion schemes by Vasconcelos [36] and for convection schemes by Diogo [37]. In the 1D space and when using Crank-Nicolson time discretization the spatial (up to the eighth-order) and the temporal orders were verified. Additionally, cases with non-uniform grids and for high Peclet numbers were also studied.

In the 2D space, weighted least-squares method was used to minimize the local interpolation error when using cell-centred stencils based on face neighbours. The cell inertial momentums were computed using 2D Gauss quadrature integration after splitting the polygonal cells into triangles. The proposed mean operator order was verified in Matlab environment with Cartesian grids.

Once again, the spatial and time orders were verified (up to eighth order) for 2D unsteady analytical cases. For coarse meshes full Dirichlet BC was not able to keep the high spatial order for high Peclet numbers while the mixed BC case was. Regarding Solver Runtime (SRT) and memory storage high-order schemes prove to be advantageous over the second-order ones for errors smaller than $E-1$ as they require less solver time.

In Matlab environment high temporal orders (up to the sixth) were verified using BDF schemes with all high-order spatial schemes. Higher-order temporal schemes did not reduce considerably Solver Runtime for second-order spatial schemes. However, this is not true for the case of high-order spatial schemes, since high-order BDF schemes were more efficient than the CN and BDF2.

In SOL code high spatial orders (up to the eighth one) were verified for both polyhedral and hybrid grids. When comparing irregular unstructured grids quadrilateral meshes provided SRT values ten percent lower than with triangular and polyhedral grids and thirty percent lower than with hybrid meshes.

4.2 Future Work

The present unsteady convection-diffusion results obtained for 2D unstructured grids can be adapted to 3D ones since all mathematical principles are the same and it is just an implementation issue. The SOL code already has the 3D capabilities.

Another next step is the contribution for a high-order unsteady Navier-Stokes solver that uses the already implemented schemes. This is an integral part of HIBforMBI project since the moving immersed boundary problems are considered to be unsteady.

This approach is applicable to other high-order schemes such as Adams-Moulton or SDIRK schemes. A comparison study can be performed using the same parameters discussed in this Thesis (Solver Runtime and memory).

One final future work should be the incorporation of turbulence modelling based on the Large-Eddy Simulation and quantifying the performance of the implemented high-order schemes in turbulent spectrum when compared to the second-order schemes.

Bibliography

- [1] N. K. Johnson and L. F. Richardson. Weather Prediction by Numerical Process. *The Mathematical Gazette*, 1922. ISSN 00255572. doi: 10.2307/3603284.
- [2] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen*, 1928. ISSN 00255831. doi: 10.1007/BF01448839.
- [3] R. Richtmeyer and K. Morton. Difference Methods for Initial-Value Problems. *Physics Today*, 1959. ISSN 19450699. doi: 10.1063/1.3060779.
- [4] B. Lax, Peter, Wendroff. Systems of Conservation Laws. *Communications on Pure and Applied Mathematics*, XIII:217–237, 1960.
- [5] J. P. Boris and D. L. Book. Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works. *Journal of Computational Physics*, 1973. ISSN 10902716. doi: 10.1016/0021-9991(73)90147-2.
- [6] B. van Leer. Towards the ultimate conservative difference scheme I. The quest of monotonicity. 1973. doi: 10.1007/bfb0118673.
- [7] A. JAMESON, W. SCHMIDT, and E. TURKEL. Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes. 1981. doi: 10.2514/6.1981-1259.
- [8] B. Van Leer. CFD education: Past, present, future. In *37th Aerospace Sciences Meeting and Exhibit*, 1999. doi: 10.2514/6.1999-910.
- [9] J. R. Chawner, J. F. Dannenhoffer, and N. J. Taylor. Geometry, mesh generation, and the CFD 2030 vision. In *46th AIAA Fluid Dynamics Conference*, 2016. ISBN 9781624104367. doi: 10.2514/6.2016-3485.
- [10] F. D. Witherden and A. Jameson. Future directions of computational fluid dynamics. In *23rd AIAA Computational Fluid Dynamics Conference, 2017*, 2017. ISBN 9781624105067. doi: 10.2514/6.2017-3791.
- [11] F. D. Witherden, A. M. Farrington, and P. E. Vincent. PyFR: An open source framework for solving advection-diffusion type problems on streaming architectures using the flux reconstruction approach. *Computer Physics Communications*, 2014. ISSN 00104655. doi: 10.1016/j.cpc.2014.07.011.

- [12] F. Cappello. Fault tolerance in petascale/exascale systems: Current knowledge, challenges and research opportunities. In *International Journal of High Performance Computing Applications*, 2009. doi: 10.1177/1094342009106189.
- [13] S. Laurent, S. Dutz, U. O. Häfeli, and M. Mahmoudi. Magnetic fluid hyperthermia: Focus on superparamagnetic iron oxide nanoparticles, 2011. ISSN 00018686.
- [14] H. H. Pennes. Analysis of tissue and arterial blood temperatures in the resting human forearm. *Journal of Applied Physiology*, 1998. ISSN 87507587. doi: 10.1152/jappl.1998.85.1.5.
- [15] B. Sawicki and A. Miaskowski. Nonlinear higher-order transient solver for magnetic fluid hyperthermia. *Journal of Computational and Applied Mathematics*, 2014. ISSN 03770427. doi: 10.1016/j.cam.2014.02.008.
- [16] R. Metzler and J. Klafter. The random walk's guide to anomalous diffusion: A fractional dynamics approach, 2000. ISSN 03701573.
- [17] C. N. Angstmann, B. I. Henry, and A. V. McGann. A fractional-order infectivity SIR model. *Physica A: Statistical Mechanics and its Applications*, 2016. ISSN 03784371. doi: 10.1016/j.physa.2016.02.029.
- [18] T. J. Osler. Taylor's Series Generalized for Fractional Derivatives and Applications. *SIAM Journal on Mathematical Analysis*, 1971. ISSN 0036-1410. doi: 10.1137/0502004.
- [19] J. A. Ekaterinaris. High-order accurate, low numerical diffusion methods for aerodynamics, 2005. ISSN 03760421.
- [20] S. De Rango and D. W. Zingg. Higher-order spatial discretization for turbulent aerodynamic computations. *AIAA Journal*, 2001. ISSN 00011452. doi: 10.2514/3.14868.
- [21] T. J. Baker. Mesh generation: Art or science?, 2005. ISSN 03760421.
- [22] A. N. Tikhonov and A. A. Samarskii. Homogeneous difference schemes. *USSR Computational Mathematics and Mathematical Physics*, 1962. ISSN 00415553. doi: 10.1016/0041-5553(62)90005-8.
- [23] S. V. Patankar. Numerical heat transfer and fluid flow. 1980. ISSN 0029-5639. doi: 10.13182/nse81-a20112.
- [24] T. J. Barth and P. O. Frederickson. Higher order solution of the euler equations on unstructured grids using quadratic reconstruction. In *28th Aerospace Sciences Meeting, 1990*, 1990. doi: 10.2514/6.1990-13.
- [25] C. Ollivier-Gooch and M. Van Altena. A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. *Journal of Computational Physics*, 2002. ISSN 00219991. doi: 10.1006/jcph.2002.7159.

- [26] A. Nejat and C. Ollivier-Gooch. A high-order accurate unstructured finite volume Newton-Krylov algorithm for inviscid compressible flows. *Journal of Computational Physics*, 2008. ISSN 10902716. doi: 10.1016/j.jcp.2007.11.011.
- [27] K. Michalak and C. Ollivier-Gooch. Limiters for unstructured higher-order accurate solutions of the euler equations. In *46th AIAA Aerospace Sciences Meeting and Exhibit*, 2008. ISBN 9781563479373. doi: 10.2514/6.2008-776.
- [28] S. L. Clain, G. J. Machado, and R. M. S. Pereira. A new very high-order finite volume method for the 2D convection diffusion problem on unstructured meshes. In *IV Conferência Nacional em Mecânica dos Fluidos, Termodinâmica e Energia*, 2012.
- [29] R. Costa, S. Clain, and G. J. Machado. New cell-vertex reconstruction for finite volume scheme: Application to the convection-diffusion-reaction equation. *Computers and Mathematics with Applications*, 2014. ISSN 08981221. doi: 10.1016/j.camwa.2014.08.015.
- [30] R. Costa, S. Clain, and G. J. Machado. A sixth-order finite volume scheme for the steady-state incompressible Stokes equations on staggered unstructured meshes. *Journal of Computational Physics*, 2017. ISSN 10902716. doi: 10.1016/j.jcp.2017.07.047.
- [31] X. Nogueira, L. Cueto-Felgueroso, I. Colominas, H. Gómez, F. Navarrina, and M. Casteleiro. On the accuracy of finite volume and discontinuous Galerkin discretizations for compressible flow on unstructured grids. *International Journal for Numerical Methods in Engineering*, 2009. ISSN 00295981. doi: 10.1002/nme.2538.
- [32] C. F. Ollivier-Gooch. High-order ENO schemes for unstructured meshes based on least-squares reconstruction. In *35th Aerospace Sciences Meeting and Exhibit*, 1997. doi: 10.2514/6.1997-540.
- [33] C. Hu and C. W. Shu. Weighted Essentially Non-oscillatory Schemes on Triangular Meshes. *Journal of Computational Physics*, 1999. ISSN 00219991. doi: 10.1006/jcph.1998.6165.
- [34] A. Jalali and C. Ollivier-Gooch. Higher-order unstructured finite volume RANS solution of turbulent compressible flows. *Computers and Fluids*, 2017. ISSN 00457930. doi: 10.1016/j.compfluid.2016.11.004.
- [35] A. Jalali and C. Ollivier-Gooch. Higher-order unstructured finite volume methods for turbulent aerodynamic flows. In *22nd AIAA Computational Fluid Dynamics Conference*, 2015. ISBN 9781624103667. doi: 10.2514/6.2015-2284.
- [36] A. Vasconcelos. *A Very High-Order Finite Volume Method Based on Weighted Least Squares for the Solution of Poisson Equation on Unstructured Grids (MSc thesis)*. 2017.
- [37] F. J. M. Diogo. *A Very High-Order Finite Volume Technique for Convection-Diffusion Problems on Unstructured Grids (MSc thesis)*. 2019.

- [38] F. Bashforth and J. C. Adams. An attempt to test the theories of capillary action by comparing the theoretical and measured forms of drops of fluid, with an explanation of the method of integration employed in constructing the tables which give the theoretical forms of such drops. *University Press, Cambridge*, 1883.
- [39] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 1987. ISSN 10902716. doi: 10.1016/0021-9991(87)90031-3.
- [40] C. F. Curtiss and J. O. Hirschfelder. Integration of Stiff Equations. *Proceedings of the National Academy of Sciences*, 1952. ISSN 0027-8424. doi: 10.1073/pnas.38.3.235.
- [41] A. B. Farnell and P. Henrici. Discrete Variable Methods in Ordinary Differential Equations. *The American Mathematical Monthly*, 1962. ISSN 00029890. doi: 10.2307/2311275.
- [42] H. Bijl, M. H. Carpenter, and V. N. Vatsa. Time integration schemes for the unsteady navier-stokes equations. In *15th AIAA Computational Fluid Dynamics Conference*, 2001. doi: 10.2514/6.2001-2612.
- [43] J. R. Cash. On the integration of stiff systems of O.D.E.s using extended backward differentiation formulae. *Numerische Mathematik*, 1980. ISSN 0029599X. doi: 10.1007/BF01396701.
- [44] J. R. Cash. The integration of stiff initial value problems in ODEs using modified extended backward differentiation formulae. *Computers and Mathematics with Applications*, 1983. ISSN 08981221. doi: 10.1016/0898-1221(83)90122-0.
- [45] G. Psihoyios. A class of implicit advanced step-point methods with a parallel feature for the solution of stiff initial value problems. *Mathematical and Computer Modelling*, 2004. ISSN 08957177. doi: 10.1016/j.mcm.2005.01.014.
- [46] B. Sanderse. Energy-conserving runge-kutta methods for the incompressible navier-stokes equations. *Journal of Computational Physics*, 2013. ISSN 10902716. doi: 10.1016/j.jcp.2012.07.039.
- [47] M. Crouzeix. Sur la B-stabilité des méthodes de Runge-Kutta. *Numerische Mathematik*, 1979. ISSN 0029599X. doi: 10.1007/BF01397651.
- [48] S. P. Nørsett and P. G. Thomsen. Embedded sdirk-methods of basic order three. *BIT*, 1984. ISSN 00063835. doi: 10.1007/BF01934920.
- [49] J. Qiu and C. W. Shu. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method II: Two dimensional case. *Computers and Fluids*, 2005. ISSN 00457930. doi: 10.1016/j.compfluid.2004.05.005.
- [50] J. Li and Z. Du. A two-stage fourth order time-accurate discretization for Lax-Wendroff type flow solvers I. Hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 2016. ISSN 10957200. doi: 10.1137/15M1052512.

- [51] J. Magalhães. *An adaptive framework for the numerical simulation of environmental flows in complex geometries (PhD thesis)*. 2011.
- [52] D. M. S. Albuquerque. *Numerical Computation of Incompressible Flows on Adaptive and Unstructured Grids (PhD thesis)*. 2013.
- [53] J. A. S. Pereira. *A 2-D Immersed Boundary Method on Low Reynolds Moving Body (MSc thesis)*. Instituto Superior Técnico, Universidade de Lisboa, 2014.
- [54] D. Martins. *On the suppression of spurious pressure oscillations in immersed boundary methods with unstructured grids (MSc thesis)*. 2016.
- [55] A. L. A. Leite. *Structured and unstructured finite volume calculations of a laminar and a turbulent axisymmetric jet flow (MSc thesis)*. 2014.
- [56] H. Pina. *Métodos Numéricos*. McGraw-Hill, Portugal, 1st editio edition, 1995.
- [57] H. Versteeg and W. Malalasekera. An Introduction to Computational Fluid Dynamics. In *Pearson Education Limited*, chapter Chapter 5. 2007. ISBN 9780131274983. doi: 10.2514/1.22547.
- [58] F. Moukalled, L. Mangani, and M. Darwish. Erratum to: The Finite Volume Method in Computational Fluid Dynamics. chapter Chapter 11. 2016. doi: 10.1007/978-3-319-16874-6_21.
- [59] S. Deng. *Quadrature formulas in two dimensions*. Dept. of Mathematics and Statistics, UNC at Charlotte, 2010.
- [60] J. E. Akin. *Finite Element Analysis with Error Estimators*. 2005. ISBN 9780750667227. doi: 10.1016/B978-0-7506-6722-7.X5030-9.
- [61] R. S. Hutchinson, Scott A., Shadid, J. N., Tuminaro. *Aztec User's Guide Version 1.1*. Massively Parallel Computing Research Laboratory, Sandia National Laboratories, 1999.

