UNIVERSITY OF CAPE TOWN

MASTERS DISSERTATION

A Finite Volume Discretization Method for Flow on Structured and Unstructured Anisotropic Meshes

Author:
DANE G. MERRICK

Supervisor:
PROF. A. G. MALAN

Dissertation submitted to the University of Cape Town in candidature for the degree of Master of Science in Mechanical Engineering

July 2015

Plagiarism Declaration

I, Dane G. Merrick, hereby declare that the work on which this dissertation is based is my original work (except where acknowledgements indicate otherwise) and that neither the whole work nor any part of it has been, is being, or is to be submitted for another degree in this or any other university. I authorise the University to reproduce for the purpose of research either the whole or any portion of the contents in any manner whatsoever. I confirm that:

1. I know that plagiarism is a serious form of academic dishonesty.
2. I have read the document about avoiding plagiarism, am familiar with its contents and have avoided all forms of plagiarism mentioned there.
3. Where I have used the words of others, I have indicated this by the use of quotation marks.
4. I have referenced all quotations and properly acknowledged other ideas borrowed from others.
5. I have not and shall not allow others to plagiarise my work.
6. I declare that this is my own work.

Signed: ______________________________________

Date: ______________________________________

UNIVERSITY OF CAPE TOWN

Abstract

Engineering and the Built Environment
Mechanical Engineering

Master of Science

A Finite Volume Discretization Method for Flow on Structured and Unstructured Anisotropic Meshes

by Dane G. Merrick

This project is concerned with advection discretization technology within the field of Computational Fluid Dynamics (CFD). To this end, two novel methods are proposed which are dubbed the Enhanced Taylor (ET) Schemes. The model equation for this work is the advection-diffusion equation with the industrial application being incompressible flow. The objective of the proposed schemes is to achieve increased accuracy on structured and unstructured anisotropic meshes. One of the schemes focuses on improving advection accuracy, and the other on improving total advection-diffusion accuracy. Fundamental to the design of the ET schemes is the primary focus on face accuracy, with the additional incorporation of the up and downwind mesh stretching factors and flow gradients. Additionally, non-linear blending with the existing NVSF scheme was effected in the interest of robustness and stability, particularly on equispaced meshes. The developed schemes, along with prominent linear κ-Upwind schemes were critically assessed and compared. Current methods were shown to be at best 3rd and 1st-order accurate at non-equispaced faces and nodes respectively. In contrast, the developed schemes were shown to be up to 4th and 2nd-order accurate. Numerical experiments followed. This involved applying the prominent and developed schemes to solve the 1D advection-diffusion equation on stretched meshes. The 2D case involved incompressible flow in a lid-driven cavity. Anisotropic structured and unstructured meshes were employed. Significant improvements in accuracy were found with the ET schemes, with average reductions in error measuring up to a 50%. In comparison to existing methods, it is proposed that state-of-the-art technology has been developed.

Acknowledgements

I would like to acknowledge and extend my deepest gratitude to my supervisor, Prof. Arnaud Malan, for his invaluable guidance, support, encouragement, and motivation throughout the duration of this work. Without his help, this work would surely never have seen the light of day. To the very dear friends that I have made throughout my University career, I would like to thank you for all the times that we have spent together and, most particularly, for those times spent away from the books. Those times, after all, will be the ones that we remember the most. I would also like to thank Prof. Bob Tait and Robs Tait for providing me with a much loved home-away-from-home at Smuts Hall. I will always cherish and think back fondly on my time spent there. Finally, I would like to thank and extend my deepest love to my parents, Dave and Charmain, and sister, Katherine, for all the love and support that they have given to me throughout my life. The sacrifices that they have made to aid my development and education have meant the world to me, and I will forever appreciate it.

Contents

List of Figures

List of Tables

Abbreviations

CDS	**C**entral **D**ifference **S**cheme ($\kappa = 1$)
CFD	**C**omputational **F**luid **D**ynamics
CM	**C**omputational **M**echanics
CSM	**C**omputational **S**olid **M**echanics
CBC	**C**onvective **B**oundedness **C**riteria
CUI	**C**ubic-**U**pwind **I**nterpolation $\left(\kappa = \dfrac{1}{3}\right)$
EAP	**E**rror **A**nalysis **P**rocess
ET	**E**nhanced-**T**aylor
GFI	**G**reen's **F**irst **I**dentity
LHS	**L**eft **H**and **S**ide
LUI	**L**inear-**U**pwind **I**nterpolation ($\kappa = -1$)
NVD	**N**ormalised **V**ariable **D**iagram
NVSF	**N**ormalised **V**ariable and **S**pace **F**ormulation $\left(\kappa_L = \dfrac{3f-1}{2f},\ \ \kappa_R = \dfrac{3g-1}{2g}\right)$
1D	One-**D**imensional
OoA	**O**rder-**o**f-**A**ccuracy
QUICK	**Q**uadratic-**U**pwind **I**nterpolation $\left(\kappa = \dfrac{1}{2}\right)$
RHS	**R**ight **H**and **S**ide
3D	Three-**D**imensional
TSE	**T**aylor **S**eries **E**xpansion
TVD	**T**otal **V**ariation **D**iminishing
2D	Two-**D**imensional

Nomenclature

All values are dimensionless unless otherwise stated.

A	Area	m^2
$\dfrac{d}{dx}$	Derivative with respect to x	
f	Stretching factor of node u	
f	Face	
g	Stretching factor of node c	
h	Stretching factor of node d	
k	Diffusion constant	$\mathrm{m}^2\mathrm{s}^{-1}$
L	Characteristic length	m
n	Outward pointing unit normal at a face	
u	Advection value	ms^{-1}
U	Free-stream velocity	ms^{-1}
V	Volume	m^3
x	x-Coordinate	m
Δ	Delta difference symboliser	
κ	Kappa-Upwind constant	
μ	Dynamic viscosity	$\mathrm{kgm}^{-1}\mathrm{s}^{-1}$
∇	Nabla differential operator	
ν	Kinematic viscosity	$\mathrm{m}^2\mathrm{s}^{-1}$
ϕ	An arbitrary function	
ρ	Density	kgm^{-3}

Subscripts:

0	The coordinate about which a TSE is carried out
c	Node c
d	Node d
dd	Node dd
L	Left
R	Right
u	Node u
uu	Node uu
uu^*	An upwind projection of node uu
x	x-Direction

In the interest of readability throughout this dissertation, when a subscript is applied to a bracketed or compound term it is followed by a vertical bar (i.e. $\left.\frac{a}{b}\right|_c$ or $(a+b)|_c$); when it is applied to a single term, however, no vertical bar is used (i.e. a_c).

For my parents, Dave and Charmain, and sister, Katherine; I love and appreciate you all more than you could ever know.

Chapter 1

Introduction

Scientists, mathematicians, and engineers have been using mathematical models and equations to represent and solve physical phenomena for centuries. Before the invention of modern day solving technology, however, the ultimate potential of such modelling was limited to the capacity of the human mind. With the birth of the computer, these practitioners found themselves aided by an ally with powerful yet limited solving capacity. The implications and potential of the newfound solving technology lead to the birth of a broad scientific discipline known as Computational Mechanics (CM). The discipline combines the fields of physics, mathematics (particularly numerical methods and differential calculus), and computer science to model physical phenomena on a continuum level. This has revolutionised science and engineering by allowing practitioners to describe phenomena in a quantitative manner that could not previously practically be done, as well as allowing experimentation to be both hastened and cheapened. This has lead to advancement for science and the human race that was not possible before.

Within CM, many specialised fields have emerged, with two of the most popular being Computational Fluid Dynamics (CFD) and Computational Solid Mechanics (CSM). As the names imply, CFD is used to model fluid flows, whereas CSM is used to model the mechanics of solid materials. CFD is the field for which this work is intended. CFD, just like CM, envelops a wide range of applications including, but not limited to, the modelling of aircraft and vehicle aerodynamics, ship hydrodynamics, internal combustion engines, electronic unit coolings, pollutant distributions, the flow of bodies of water, weather predictions, and the flow of blood through the body.

These simulations generally involve complex physics modelled on or around complex geometries, which often leads to the absence of analytical, or “exact”, solutions. This means that the accuracy of the numerical methods used on the required anisotropic computational meshes is of high importance. Improving the accuracy comes at a cost (i.e. greater computational usage). This compromise between accuracy and computational cost is what is widely known as “efficiency”, and has a massive influence on the amount of work that can realistically be carried out within any finite budget (time and money). As a result, improving efficiency is a crucial concern within CFD research. This dissertation will focus on this aspect and in the arena of modelling flows involving smooth fields. The 1D advection-diffusion equation is therefore used as the model equation with the application study involving incompressible flow.

The majority of incompressible CFD problems consist of both advective and diffusive flow components. When considering the widely used finite volume method, discretization of the diffusive component is a fairly simple matter. The treatment of the advective component, however, is frought with complexity. In recent history, CFD advection discretization research has generally been expressed within the context of either the Normalised Variable Diagram (NVD) (Leonard [16]), or the Total Variation Diminishing (TVD) scheme (Harten [11][12]). Both of these philosophies have associated criteria (that were developed independently) that must be met by a scheme in order to ensure the boundedness of any resulting solution. The Convective Boundedness Criteria (CBC) (Gaskell and Lau [9]) is used for the NVD, while the Flux-Limiting Diagram (Sweby [18]) is used with the TVD scheme. Many schemes created before the advent of the two advection philosophies have subsequently been converted to be expressed as either an NVD or TVD scheme. It is also possible, and fairly simple, to express NVD schemes as TVD schemes, and vice versa. This conversion process is eloquently described by Waterson and Deconinck [22]. The grouping of schemes into the two philosophies has naturally helped analysis and comparison.

The above advection discretization methods have historically been dominated by asymmetric, upwind-biased schemes that are defined as either linear or non-linear. Many non-linear schemes were developed by adapting existing linear schemes to conform to the CBC, or Flux-Limiting Diagram. As such, linear schemes form the foundations of many advection discretization schemes. A convenient way of describing a linear scheme for ease of analysis and comparison is as a member of what is known as the κ-Upwind class, which was originally developed by van Leer [19][20]. While many schemes have been developed over the history of CFD, some have naturally attained a greater degree of popularity. These select schemes, along with their reasons for popularity are described below. In addition, the less popular Normalised Variable and Space Formulation (NVSF) scheme (Darwish and Moukalled [5][4]) will also be described due to its relevance to this dissertation.

It is generally accepted by the finite-volume CFD community that in terms of pure accuracy (which is a primary component of efficiency) the Cubic-Upwind Interpolation (CUI) Scheme (Agarwal [1]), which is represented by a κ-value of $\frac{1}{3}$ and marketed as being 3rd-order-accurate at a node, is the most accurate linear advection discretization scheme available. Interestingly, however, the scheme is only 2nd-order-accurate at a face, and was predominantly designed to achieve these accuracies on equispaced grids. Closely following CUI in the accuracy race is the Quadratic-Upwind Interpolation (QUICK) scheme (Leonard [15]), which is represented by a κ-value of $\frac{1}{2}$. QUICK is 2nd-order-accurate at a node and 3rd-order-accurate at a face. Similarly to CUI, QUICK was also designed using an equispaced mesh as starting point. There are few numerical comparisons of CUI and QUICK and, due to their differing accuracies (CUI maximised at a node; QUICK maximised at a face), it is difficult to tell which scheme is superior for CFD calculations when using non-equispaced grids. Both schemes will be formally assessed in this work. None of the other well-known upwind-biased advection discretization schemes reviewed are considered to be primarily focused on efficiency for general CFD applications to the same degree as that of CUI and QUICK. These, however, have advantages in certain niche areas and are therefore mentioned below for sake of completeness.

Fromm's scheme (Fromm [8]), represented by a κ-value of 0, is marketed primarily for its ability to minimise dispersion error. The Linear-Upwind Interpolation (LUI) scheme (Warming and Beam [21]), represented by a κ-value of -1, is said to be the only κ-Upwind scheme that is completely upwind biased (ie. the face discretization has no influence of any kind from downwind nodes). The Central Difference Scheme (CDS), represented by a κ-value of 1, is not strictly an upwind biased scheme as the face discretization is not influenced by the upwind gradient. Despite this, it is a well-known advection discretization scheme and can be grouped with other κ-Upwind schemes for ease of analysis. CDS, Fromm's scheme, and LUI are all marketed as attaining 2nd-order-accuracy at both a face and node and were all predominantly designed for use on equispaced grids. The usage of κ-Upwind schemes on non-equispaced grids will very likely result in a reduction of both nodal and face accuracies, and this is analysed further in Chapter 2. In the case of Fromm's scheme and LUI, it is highly likely that usage on non-equispaced grids will also result in a reduction of their dispersion error minimisation abilities and complete upwind biases respectively. Analysis of the latter, however, is beyond the scope of this work, which is elaborated on in Section 1.1.2 below.

The NVSF scheme is, to the knowledge of the author, the only documented finite volume scheme designed to maintain accuracy on non-equispaced grids. Note that Jasak et al. [14] adapted the vertex-centered NVSF for use on a cell-centered grids. This work, however, is focused on vertex-centered grids which will be elaborated on in Section 1.1.2 below. The NVSF method accounts for mesh stretching by calculating a nodal stretching ratio of the three nodes contained within the NVD, and using this ratio to optimise the κ-value (Note that NVSF did not originally utilise the κ-Upwind format, however, the author of this work has made the conversion which is detailed in Appendix A). As a result, the accuracy at the face of the NVD is maximised. Interestingly, but understandably, NVSF simplifies to QUICK on an equispaced grid (understandable because QUICK maximises accuracy at a face). NVSF is marketed as achieving 3rd-order-accuracy at a face, and 2nd-order-accuracy at a node, for both equi and non-equispaced grids. Despite the apparent advantages of the scheme, NVSF is not without limitations and detractors. A fundamental limitation of the scheme is that it can only be utilised to discretize a face when an immediately upwind and collinear neighbouring node is present. This essentially limits the scheme to use on Cartesian structured grids, and forces the CFD practitioner to use an alternative scheme at an upwind-boundary-adjacent face. In addition, Waterson and Deconinck [22] dismissed the NVSF scheme, stating that it "increases the complexity of the resulting NV[D] scheme". While this may be strictly true, it is still possible that the NVSF results in an increase in efficiency under favourable conditions. Much of the inspiration for this work comes from NVSF so, for the above mentioned reason it will be analysed and discussed in more detail later on.

Since the design of NVSF in 1994, there has been a lull in original, linear, upwind-biased, convection discretization schemes focusing on improving efficiency. Since then, however, many new non-linear schemes with novel bounding and blending methods of essentially linear schemes have appeared. This work will hopefully go some way to revitalising and stimulating the aforementioned.

1.1 Project Motivation and Overview

It is obvious that producing CFD work more efficiently, by producing it either quicker and/or more accurately, will have significant and tangible economic and social benefits. It is therefore easy to understand why researchers have placed a large amount of effort on improving the efficiency of numerical methods. Additionally, reviewing CFD advection discretization literature reveals that there is good potential for industrially relevant research within the field. This is because previous work and research has laid a sound foundation for industrially concerned researchers to build upon.

Industrial CFD involves a large range of practical applications, many of which require the use of non-equispaced structured and unstructured meshes. As described above, the vast majority of finite volume advection discretization schemes were however developed using equispaced meshes as basis (formal accuracy is accordingly quoted assuming equispacing). In advancing on existing work, this dissertation aims to table a novel philosophy aimed at increasing accuracy with regards to two key aspects of advection-diffusion problems. Firstly, non-equispacing is assumed as fundamental basis when designing spatial discretization algorithms. Secondly, consideration will be given to treating the advection and diffusion components in a more holistic manner. This stands in contrast to previous work which has taken a mutually exclusive attitude.

1.1.1 Aim

The aim of this work is to develop a novel, improved, and industrially relevant CFD advection discretization methodology for smooth flow systems containing both advective and diffusive flow components. Therefore, the model problem used will be the 1D-advection-diffusion equation, shown below:

$$\frac{d\left(u\phi\right)}{dx} = k\frac{d^2\left(\phi\right)}{dx^2} \tag{1.1}$$

The methodology developed will be considered “improved” if it results in an advection discretization methodology that displays a greater level of efficiency than that of currently used methodologies, maintains both local and global conservation and is stable when modelling smooth fields. As noted previously, industrial relevance requires suitability to non-equispaced structured and unstructured grids.

A methodology’s “efficiency” in CFD is typically taken as the ratio of accuracy to computational cost. This work will focus on increasing the accuracy rather than on decreasing the computational cost. When used as part of an implicit flow solver, however, it should not result in any significant increase in cost per iteration. Once again, in designing an “industrially relevant” methodology it is important from an efficiency and parallel processing point of view that locality of data is maintained. This is interpreted, in the context of this work, to mean that only neighbouring nodal data is available when discretizing a face value.

1.1.2 Assumptions, Limitations, and Scope

The application of this work is limited to that of smooth flow fields. It is therefore restricted to the smooth advection-diffusion equation as well as single phase laminar steady incompressible flows. Bearing this in mind, an important philosophical belief (expressed here as an assumption due to the lack of substantiating experimental data) held by the author is that it is more important to design a scheme to achieve a high order-of-accuracy (OoA) at a face, rather than at an arbitrary node. This is as the two (note the exact known number as it is mesh independent) neighbouring nodes of an arbitrary face are easier to work with than the number of neighbouring faces of an arbitrary node.

Finally, a vertex-centered strategy is employed throughout this work (though a cell-centered method could also be employed). Boundary values are assumed known (Dirichlet boundary conditions). It is further assumed that certain information is easily accessible from neighbouring nodes, when discretizing a face. This includes the diffusion and advection values, the first derivative of the advection value, and the location of the node. Metrics such as “mesh stretching” may not be available, and algorithms to compute this will be developed in this work.

1.2 Dissertation Layout

This work consists of six chapters and six appendices. A brief summary of those to follow is given next:

- **Chapter 2, *Tools of Analysis and Error Analysis of Existing Schemes***: The fundamental mathematics, advection-diffusion discretizations, and generic error analyses are expressed and derived. Then, accuracy analyses of prominent existing advection discretization schemes are carried out for both equi- and non-equispaced meshes. An assessment of all schemes concludes the chapter.

- **Chapter 3, *Development of Enhanced-Taylor (ET) Discretization Schemes***: Two novel advection discretization schemes are developed. The first is aimed at achieving higher-order advection-diffusion accuracy, the other at achieving higher-order advection accuracy.

- **Chapter 4, *Developed ET Schemes: A Critical Analysis***: Accuracy analyses on both of the developed discretization schemes are carried out for a range of grid spacing topologies. An assessment, which includes the observed weaknesses of both schemes, is also presented.

- **Chapter 5, *Numerical Examples***: The accuracies of the author's two developed schemes are compared to existing methods. Two problem types are employed for this, viz. the 1D Advection-Diffusion equation, and the 2D Lid-Driven Cavity.

- **Chapter 6, *Conclusion***: The conclusions and findings from this work are presented, along with areas for potential future research.

- **Appendix A, *NVSF: Conversion to κ-Upwind Format***: The NVSF Scheme is converted into κ-Upwind format for ease of analysis, comparison, and implementation within this work

- **Appendix B, *Boundary-Adjacent-Face Analysis***: Boundary-adjacent-face accuracy analyses are displayed for prominent existing advection discretization schemes and the ET schemes.

- **Appendix C, *Leading Error Terms***: The exact leading error terms acquired when assessing the accuracies of the various discretization schemes, in Chapters 2, 4, and Appendix B, are given here for completeness.
- **Appendix D, *Advection-Diffusion ET Scheme: Diffusion Decoupling***: The Advection-Diffusion ET Scheme is assessed for odd-even decoupling of the diffusion term.
- **Appendix E, *1D Advection-Diffusion Test Case Data***: Extensive data from the 1D Advection-Diffusion test case is presented for completeness.
- **Appendix F, *2D Lid-Driven Cavity Test Case Data***: Extensive data from the 2D Lid-Driven Cavity test case is presented for completeness.

Chapter 2

Tools of Analysis and Error Analysis of Existing Schemes

In the interest of clarity and to describe notation, the basic mathematical tools are next developed. Following this, advection-diffusion discretization, and generic error analysis for the κ-Upwind schemes are described.

2.1 Green's First Identity

Green's First Identity (GFI) is a mathematical tool used to aid in the CFD discretization process. It is expressed as:

$$\int\limits_V (\nabla.\phi)\, dV = \oint\limits_A (\phi.n)\, dS \tag{2.1}$$

or, for a 1D case in the x-direction:

$$\int\limits_V \left(\frac{d\phi}{dx}\right) dV = \oint\limits_A (\phi.n_x)\, dS \tag{2.2}$$

where ∇ is the Nabla differential operator, n is the outward pointing unit normal at a face, and n_x is the outward pointing unit normal at a face in the x-direction.

2.2 Basic Advection-Diffusion Discretization

The basic 1D discretization of the advection-diffusion equation remains the same regardless of which scheme, Upwind or otherwise, is utilised. The discretization is described here in order to avoid later repetition.

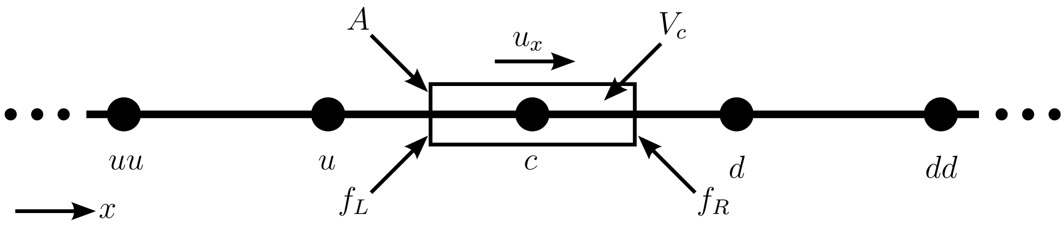

Figure 2.1: A generic and complete 1D stencil.

With regards to Figure 2.1, it is convenient to define the following notation:

- $f = \dfrac{x_u - x_{uu}}{x_c - x_{uu}}$
- $g = \dfrac{x_c - x_u}{x_d - x_u}$
- $h = \dfrac{x_d - x_c}{x_{dd} - x_c}$
- $\Delta x = x_c - x_u$

where f, g and h are the mesh stretching factors associated with the respective edges, and Δx is the edge length over the upwind control volume face. Aided by simple arithmetic, the remaining edge lengths and volumes can then be expressed as:

- $x_u - x_{uu} = \left(\dfrac{f}{1-f}\right)\Delta x$
- $x_d - x_c = \left(\dfrac{1-g}{g}\right)\Delta x$
- $x_{dd} - x_d = \left(\dfrac{1-h}{h}\right)\left(\dfrac{1-g}{g}\right)\Delta x$
- $V_c = \dfrac{A\Delta x}{2g}$

- $V_u = \dfrac{A\Delta x}{2(1-f)}$

As alluded to earlier, the model equation is the steady-state advection-diffusion equation with no source term, which is written as:

$$\frac{d(u_x\phi)}{dx} = k\frac{d^2(\phi)}{dx^2} \tag{2.3}$$

In order to begin discretization, the equation is cast into weak form as follows:

$$\frac{1}{V_c}\int\limits_{V_c}\left(\frac{d(u_x\phi)}{dx}\right)dV = \frac{1}{V_c}\int\limits_{V_c}\left(k\frac{d^2(\phi)}{dx^2}\right)dV \tag{2.4}$$

By making use of GFI, and substituting in a value for V_c, Equation (2.4) can be re-written as:

$$\frac{2g}{A\Delta x}\oint\limits_{A}(u_x\phi n_x)\,dS = \frac{2gk}{A\Delta x}\oint\limits_{A}\left(\frac{d(\phi)}{dx}n_x\right)dS \tag{2.5}$$

As the 1D control volume is aligned with the x-axis, and only consists of two faces, f_L and f_R, with x-direction unit normals of -1 and 1 respectively, the following results:

$$\frac{2g}{\Delta x}[(u_x\phi)|_{f_R} - (u_x\phi)|_{f_L}] = \frac{2gk}{\Delta x}\left[\left(\frac{d\phi}{dx}\right)\bigg|_{f_R} - \left(\frac{d\phi}{dx}\right)\bigg|_{f_L}\right] \tag{2.6}$$

where subscript f_L and f_R denotes the values at the left and right faces respectively.

By grouping common terms, the following is obtained:

$$\frac{2g}{\Delta x}\left[\left(u_x\phi - k\frac{d\phi}{dx}\right)\bigg|_{f_R} - \left(u_x\phi - k\frac{d\phi}{dx}\right)\bigg|_{f_L}\right] = 0 \tag{2.7}$$

When carrying out error analysis at a face, only one of the faces' terms from Equation (2.7) need to be considered. Conversely, when carrying out a nodal error analysis, the complete basic discretization needs to be considered.

2.3 Taylor Series Expansion

The Taylor Series Expansion (TSE) in one-dimension (1D) follows as:

$$\phi(x) = \phi(x_0) + \frac{d\phi}{dx}\bigg|_{x_0}(x-x_0) + \frac{d^2\phi}{dx^2}\bigg|_{x_0}\frac{(x-x_0)^2}{2!} + \ldots + \frac{d^n\phi}{dx^n}\bigg|_{x_0}\frac{(x-x_0)^n}{n!} \tag{2.8}$$

where x_0 is a point within the domain where ϕ, as well as all of its derivatives, are known.

TSE's are utilised extensively in this work, both when analysing accuracy, as well as when developing an alternative technique in Chapter 3. The TSE's are either carried out about the control volume node (c) when considering nodal accuracy, or about the left or right faces (f_L or f_R) of the control volume when considering accuracy at a face. All the required TSE's are set-out in Sections 2.3.1 and 2.3.2 below for later reference.

The exact calculation of error via a TSE for the general case is an impossble task due to the potentially infinite number of derivaties. Therefore, the expansion will typically have a certain number of terms truncated by the method's approximation. Due to the factorial nature of the denominator in every Taylor Series term, it is typically assumed that the first truncated term in an approximation will be significantly larger than all of the subsequent truncated terms combined. Hence, this term is defined as the "leading error term". Within the leading error term, the Order-of-Accuracy (OoA) of a method is typically defined as the power to which the grid spacing $(x - x_0)$ is raised and represents the rate at which a method converges to the exact solution as the grid spacing approaches zero. Therefore, whenever the accuracy of a numerical method is discussed it is the OoA that is considered, with a higher value being considered more accurate as this implies a faster rate of convergence to the exact solution.

2.3.1 TSE About a Node

Next, the TSE's for the various nodes in Figure 2.1 are developed. The TSE for ϕ_{uu} about node c is expressed as:

$$\begin{aligned}\phi_{uu} =& \phi_c - \left(\frac{1}{1-f}\right) \Delta x \left.\frac{d\phi}{dx}\right|_c + \left(\frac{1}{1-f}\right)^2 \left(\frac{1}{2}\right) \Delta x^2 \left.\frac{d^2\phi}{dx^2}\right|_c \\ & - \left(\frac{1}{1-f}\right)^3 \left(\frac{1}{6}\right) \Delta x^3 \left.\frac{d^3\phi}{dx^3}\right|_c + \dots\end{aligned} \tag{2.9}$$

This equation is expressed in row 2 of Table 2.1 below. All other nodal approximations are similarly listed.

Table 2.1: Coefficients of the first four TSE terms about the control volume node (c) for all required nodes in the stencil

Approximation	$\boldsymbol{\phi_c}$	$\boldsymbol{\Delta x \left.\frac{d\phi}{dx}\right\|_c}$	$\boldsymbol{\left(\frac{1}{2}\right) \Delta x^2 \left.\frac{d^2\phi}{dx^2}\right\|_c}$	$\boldsymbol{\left(\frac{1}{6}\right) \Delta x^3 \left.\frac{d^3\phi}{dx^3}\right\|_c}$
ϕ_{uu}	1	$-\left(\frac{1}{1-f}\right)$	$\left(\frac{1}{1-f}\right)^2$	$-\left(\frac{1}{1-f}\right)^3$
ϕ_u	1	-1	1	-1
ϕ_d	1	$\left(\frac{1-g}{g}\right)$	$\left(\frac{1-g}{g}\right)^2$	$\left(\frac{1-g}{g}\right)^3$
ϕ_{dd}	1	$\left(\frac{1-g}{gh}\right)$	$\left(\frac{1-g}{gh}\right)^2$	$\left(\frac{1-g}{gh}\right)^3$

2.3.2 TSE About Faces

The TSE's expressed around the left face (f_L) follows:

Table 2.2: Coefficients of the first four TSE terms about the left face of the control volume (f_L) for all required nodes in the stencil

Approximation	$\boldsymbol{\phi_{f_L}}$	$\boldsymbol{\Delta x \left.\frac{d\phi}{dx}\right\|_{f_L}}$	$\boldsymbol{\left(\frac{1}{2}\right) \Delta x^2 \left.\frac{d^2\phi}{dx^2}\right\|_{f_L}}$	$\boldsymbol{\left(\frac{1}{6}\right) \Delta x^3 \left.\frac{d^3\phi}{dx^3}\right\|_{f_L}}$
ϕ_{uu}	1	$-\left(\frac{f+1}{2(1-f)}\right)$	$\left(\frac{f+1}{2(1-f)}\right)^2$	$-\left(\frac{f+1}{2(1-f)}\right)^3$
ϕ_u	1	$-\left(\frac{1}{2}\right)$	$\left(\frac{1}{2}\right)^2$	$-\left(\frac{1}{2}\right)^3$
ϕ_c	1	$\left(\frac{1}{2}\right)$	$\left(\frac{1}{2}\right)^2$	$\left(\frac{1}{2}\right)^3$
ϕ_d	1	$\left(\frac{2-g}{2g}\right)$	$\left(\frac{2-g}{2g}\right)^2$	$\left(\frac{2-g}{2g}\right)^3$

Similarly, the TSE's about the right face of the control volume (f_R) can also be expressed:

Table 2.3: Coefficients of the first four TSE terms about the right face of the control volume (f_R) for all required nodes in the stencil

Approximation	$\boldsymbol{\phi_{f_R}}$	$\boldsymbol{\Delta x \left.\frac{d\phi}{dx}\right\vert_{f_R}}$	$\boldsymbol{\left(\frac{1}{2}\right) \Delta x^2 \left.\frac{d^2\phi}{dx^2}\right\vert_{f_R}}$	$\boldsymbol{\left(\frac{1}{6}\right) \Delta x^3 \left.\frac{d^3\phi}{dx^3}\right\vert_{f_R}}$
ϕ_u	1	$-\left(\frac{1+g}{2g}\right)$	$\left(\frac{1+g}{2g}\right)^2$	$-\left(\frac{1+g}{2g}\right)^3$
ϕ_c	1	$-\left(\frac{1-g}{2g}\right)$	$\left(\frac{1-g}{2g}\right)^2$	$-\left(\frac{1-g}{2g}\right)^3$
ϕ_d	1	$\left(\frac{1-g}{2g}\right)$	$\left(\frac{1-g}{2g}\right)^2$	$\left(\frac{1-g}{2g}\right)^3$

2.4 The Error Analysis Process (EAP)

The process used to carry out error analyses later in this report is described in this section. For ease of explanation, an error analysis example will be carried out on the following simple and fictional discretization at a left face:

$$\phi_{f_L} \approx A\phi_u + B\phi_c \tag{2.10}$$

where ϕ_{f_L} is the value being approximated, ϕ_u an ϕ_c are known values, and A and B are coefficients to be determined (which could be based on some criteria such as error minimization).

Once the discretization has been decided upon, the next step is to write TSE's for each of the known values about the point of the desired approximation. The resulting TSE for ϕ_u is shown below:

$$\begin{aligned} A\phi_u =& A\phi_{f_L} - A\left(\frac{1}{2}\right)\Delta x \left.\frac{d\phi}{dx}\right|_{f_L} + A\left(\frac{1}{2}\right)^2\left(\frac{1}{2}\right)\Delta x^2 \left.\frac{d^2\phi}{dx^2}\right|_{f_L} \\ & - A\left(\frac{1}{2}\right)^3\left(\frac{1}{6}\right)\Delta x^3 \left.\frac{d^3\phi}{dx^3}\right|_{f_L} + \ldots \end{aligned} \tag{2.11}$$

Returning to the original approximation (2.10) it follows that:

$$\begin{aligned}(A+B)\phi_{f_L} =& A\phi_u + B\phi_c - (-A+B)\left(\frac{1}{2}\right)\Delta x \left.\frac{d\phi}{dx}\right|_{f_L} \\ & - (A+B)\left(\frac{1}{2}\right)^2\left(\frac{1}{2}\right)\Delta x^2 \left.\frac{d^2\phi}{dx^2}\right|_{f_L} \\ & - (-A+B)\left(\frac{1}{2}\right)^3\left(\frac{1}{6}\right)\Delta x^3 \left.\frac{d^3\phi}{dx^3}\right|_{f_L} - \dots\end{aligned} \tag{2.12}$$

Recalling that we are approximating ϕ_{f_L}, it stands to reason that $A + B = 1$. The extraneous terms represent the truncation error. The outlined process will be utilised throughout this work for all error analyses.

2.5 Generic Error Analyses of κ-Upwind Schemes

The generic error analyses described in this section will later be used to assess the accuracy of the currently used κ-Upwind schemes at various location types within a mesh, namely an internal face, a boundary-adjacent face, and an internal node. Both pure advection error analysis and combined advection-diffusion error analysis are next carried out at each.

2.5.1 Internal Face

As per Equation (2.7), the basic advection-diffusion discretization at a face is:

$$\left.\left(u_x\phi - k\frac{d\phi}{dx}\right)\right|_f \tag{2.13}$$

When considering pure advection error, only the approximation method of ϕ needs to be considered. In the case of the total advection-diffusion error, however, the complete approximation needs to be considered.

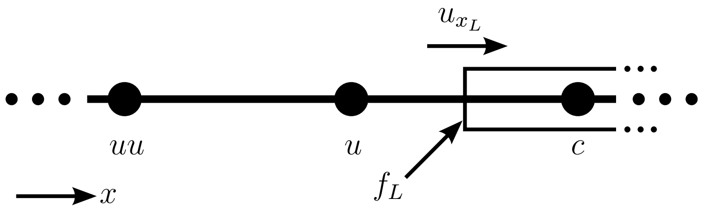

FIGURE 2.2: A κ-Upwind Scheme internal face stencil.

On a mesh stencil similar to that shown in Figure 2.2, Waterson and Deconinck [22] show that a 1D κ-Upwind face approximation on an equispaced mesh is:

$$\phi_{f_L} \approx \frac{1}{2}\left(\phi_u + \phi_c\right) - \frac{1-\kappa}{4}\left(\phi_c - 2\phi_u + \phi_{uu}\right) \tag{2.14}$$

In keeping with an industrially relevant philosophy (applicable to stretched and unstructured meshes), it is important to replace the upwind node (ϕ_{uu}) with a generalised upwind projection (ϕ_{uu^*}) as:

$$\phi_{f_L} \approx \frac{1}{2}\left(\phi_u + \phi_c\right) - \frac{1-\kappa}{4}\left(\phi_c - 2\phi_u + \phi_{uu^*}\right) \tag{2.15}$$

The generalised upwind projection must be designed to utilise only readily available information at the face, as well as to return the exact value of the upwind node on an equispaced cartesian mesh. It should further be supportive of upwind dominance as well as accuracy, with the employed expression reading:

$$\phi_{uu^*} = \phi_c - 2\left.\frac{d\phi}{dx}\right|_u \Delta x \tag{2.16}$$

For the 1D internal face stencil shown in Figure 2.2, $\left.\frac{d\phi}{dx}\right|_u$ can be expressed in a more analysable nodal form by making use of a simple discretization as follows:

$$\begin{aligned}\left.\frac{d\phi}{dx}\right|_u &= \frac{1}{V_u}\int\limits_V \left(\frac{d\phi}{dx}\right) dV \\ &= \frac{2(1-f)}{A\Delta x}\oint\limits_A (\phi n_x) dS \\ &= \frac{2(1-f)}{A\Delta x}(\phi_{f_R} - \phi_{f_L})A \\ &\approx \frac{2(1-f)}{\Delta x}\left(\frac{\phi_u + \phi_c}{2} - \frac{\phi_{uu} + \phi_u}{2}\right) \\ &\approx \frac{1-f}{\Delta x}\left(\phi_c - \phi_{uu}\right)\end{aligned} \tag{2.17}$$

A generalised κ-Upwind left internal face approximation is derived by substituting Equation (2.17) into Equation (2.16), followed by Equation (2.15).

$$\phi_{f_L} \approx \left(\frac{-1+f+\kappa-f\kappa}{2}\right)\phi_{uu} + \left(\frac{2-\kappa}{2}\right)\phi_u + \left(\frac{1-f+f\kappa}{2}\right)\phi_c \tag{2.18}$$

The approximation of the diffusive term is a much simpler affair and results, for f_L, as:

$$\left.\frac{d\phi}{dx}\right|_{f_L} \approx \left(\frac{-1}{\Delta x}\right)\phi_u + \left(\frac{1}{\Delta x}\right)\phi_c \tag{2.19}$$

Pure advection error analysis at an internal (non-boundary) face can be carried out on Equation (2.18). To suitably prepare for the advection-diffusion error analysis at an internal face, however, Equations (2.18) and (2.19) need to be substituted into Equation (2.13) as follows:

$$\begin{aligned}\left.\left(u_x\phi - k\frac{d\phi}{dx}\right)\right|_{f_L} \approx & \left[u_x\left(\frac{-1+f+\kappa-f\kappa}{2}\right)\right]\phi_{uu} \\ & + \left[u_x\left(\frac{2-\kappa}{2}\right) + \frac{k}{\Delta x}\right]\phi_u \\ & + \left[u_x\left(\frac{1-f+f\kappa}{2}\right) - \frac{k}{\Delta x}\right]\phi_c\end{aligned} \tag{2.20}$$

Due to the nature of the internal face discretization, it can also be suitably applied for use at a downwind-boundary-adjacent face. The upwind-boundary-adjacent face, however, requires a separate discretization and error analysis. This is done in Section 2.5.2 below.

By applying the EAP to Equation (2.18), the advection error at an internal face results as follows:

$$\begin{aligned}\phi_{f_L} = & \left(\frac{-1+f+\kappa-f\kappa}{2}\right)\phi_{uu} + \left(\frac{2-\kappa}{2}\right)\phi_u + \left(\frac{1-f+f\kappa}{2}\right)\phi_c \\ & - \left(\frac{-2f\kappa+3f-1}{8\,(f-1)}\right)\Delta x^2 \left.\frac{d^2\phi}{dx^2}\right|_{f_L} \\ & - \left(\frac{f\,(f+1)\,(1-\kappa)}{24\,(f-1)^2}\right)\Delta x^3 \left.\frac{d^3\phi}{dx^3}\right|_{f_L} - \ldots\end{aligned} \tag{2.21}$$

All κ-Upwind Schemes clearly produce at least 2nd-order advection accuracy at a face, regardless of the mesh stretching factor (f). It is further conceivable that a higher OoA is possible, and will depend on the combination of κ and mesh stretching factor.

By applying the EAP to Equation (2.20), the total advection-diffusion error at an internal face results as follows:

$$
\begin{aligned}
\left.\left(u_x\phi - k\frac{d\phi}{dx}\right)\right|_{f_L} =& \left[u_x\left(\frac{-1+f+\kappa-f\kappa}{2}\right)\right]\phi_{uu} + \left[u_x\left(\frac{2-\kappa}{2}\right) + \frac{k}{\Delta x}\right]\phi_u \\
&+ \left[u_x\left(\frac{1-f+f\kappa}{2}\right) - \frac{k}{\Delta x}\right]\phi_c \\
&- \left[u_x\left(\frac{2f\kappa-3f+1}{8\left(1-f\right)}\right)\left.\frac{d^2\phi}{dx^2}\right|_{f_L} - \left(\frac{k}{24}\right)\left.\frac{d^3\phi}{dx^3}\right|_{f_L}\right]\Delta x^2 \\
&+ u_x\left(\frac{f\left(f+1\right)\left(\kappa-1\right)}{24\left(f-1\right)^2}\right)\Delta x^3\left.\frac{d^3\phi}{dx^3}\right|_{f_L} - \ldots
\end{aligned}
\tag{2.22}
$$

Theoretically, for very specific cases, higher than 2nd-order accuracy is again possible. Practically, however, manipulating the κ value at each face to achieve these accuracies is problematic due to the difficulty in computing higher-order face gradients ($\left.\frac{d^2\phi}{dx^2}\right|_{f_L}$ and $\left.\frac{d^3\phi}{dx^3}\right|_{f_L}$). It would clearly be more useful to therfore use a scheme which is not dependent on higher-order gradients in field variables.

2.5.2 Upwind-Boundary-Adjacent Face

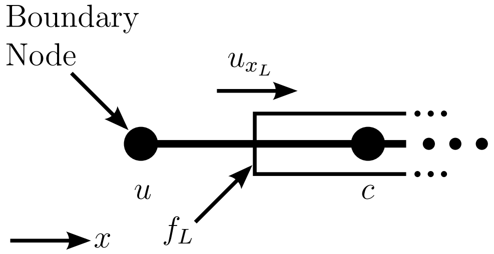

FIGURE 2.3: A κ-Upwind Scheme upwind-boundary-adjacent face stencil.

With reference to Figure 2.3, the volume at the boundary node is calculated as:

$$V_u = \frac{A\Delta x}{2} \tag{2.23}$$

In reference to Figure 2.3, $\left.\frac{d\phi}{dx}\right|_u$ as employed for the purpose of upwind value projection may be computed as:

$$\begin{aligned}
\left.\frac{d\phi}{dx}\right|_u &= \frac{1}{V_u}\int\limits_V \left(\frac{d\phi}{dx}\right) dV \\
&= \frac{2}{A\Delta x}\oint\limits_A (\phi n_x) dS \\
&= \frac{2}{A\Delta x}(\phi_{f_R} - \phi_{f_L})A \\
&\approx \frac{2}{\Delta x}\left(\frac{\phi_u + \phi_c}{2} - \phi_u\right) \\
&\approx \frac{1}{\Delta x}\left(\phi_c - \phi_u\right)
\end{aligned} \tag{2.24}$$

A generalised κ-Upwind left boundary-adjacent face approximation results as:

$$\phi_{f_L} \approx \frac{1}{2}\left(\phi_u + \phi_c\right) \tag{2.25}$$

This is equivalent to the use of CDS with $\kappa = 1$. In order to prepare for the advection-diffusion error analysis, Equations (2.25) and (2.19) must be substituted into Equation (2.13) as follows:

$$\left.\left(u_x\phi - k\frac{d\phi}{dx}\right)\right|_{f_L} \approx \left[\frac{u_x}{2} + \frac{k}{\Delta x}\right]\phi_u + \left[\frac{u_x}{2} - \frac{k}{\Delta x}\right]\phi_c \tag{2.26}$$

Advection and advection-diffusion error analyses on the above two equations are carried out next.

By applying the EAP to Equation (2.25), the pure advection error at a boundary-adjacent face results as follows:

$$\phi_{f_L} = \left(\frac{1}{2}\right)\phi_u + \left(\frac{1}{2}\right)\phi_c - \left(\frac{1}{8}\right)\Delta x^2 \left.\frac{d^2\phi}{dx^2}\right|_{f_L} - \left(\frac{1}{384}\right)\Delta x^4 \left.\frac{d^4\phi}{dx^4}\right|_{f_L} - \dots \tag{2.27}$$

As expected, 2nd-order advection accuracy is achievable at a boundary-adjacent face regardless of the κ-value.

By applying the EAP to Equation (2.26), the total advection-diffusion error at a boundary-adjacent face results in:

$$\begin{aligned}\left(u_x\phi - k\frac{d\phi}{dx}\right)\Bigg|_{f_L} &= \left[\frac{u_x}{2}+\frac{k}{\Delta x}\right]\phi_u + \left[\frac{u_x}{2}-\frac{k}{\Delta x}\right]\phi_c \\ &\quad - \left[u_x\left(\frac{1}{8}\right)\frac{d^2\phi}{dx^2}\bigg|_{f_L} - k\left(\frac{1}{24}\right)\frac{d^3\phi}{dx^3}\bigg|_{f_L}\right]\Delta x^2 - \dots\end{aligned} \tag{2.28}$$

As expected, the OoA at a boundary-adjacent face is 2nd-order accurate and not dependant on the κ-value used.

2.5.3 Internal Node

Error analysis at an internal node considers Equation (2.3), and the assumption of constant u_x is made (i.e. $u_{x_L} = u_{x_R} = u_x$). Discretization of the advection-diffusion equation commences as:

$$u_x\,\frac{d\phi}{dx}\bigg|_c - k\,\frac{d^2\phi}{dx^2}\bigg|_c = \frac{2g}{\Delta x}\left[u_x\left(\phi_{f_R}-\phi_{f_L}\right) - k\left(\frac{d\phi}{dx}\bigg|_{f_R} - \frac{d\phi}{dx}\bigg|_{f_L}\right)\right] \tag{2.29}$$

Using Equation (2.18), ϕ_{f_L} can be represented nodally as:

$$\phi_{f_L} \approx \left(\frac{-1+f+\kappa-f\kappa}{2}\right)\phi_{uu} + \left(\frac{2-\kappa}{2}\right)\phi_u + \left(\frac{1-f+f\kappa}{2}\right)\phi_c \tag{2.30}$$

While from Equation (2.19) we may write:

$$\frac{d\phi}{dx}\bigg|_{f_L} \approx \left(\frac{-1}{\Delta x}\right)\phi_u + \left(\frac{1}{\Delta x}\right)\phi_c \tag{2.31}$$

The right (downwind) face similarly results in:

$$\phi_{f_R} \approx \left(\frac{-1+g+\kappa-g\kappa}{2}\right)\phi_u + \left(\frac{2-\kappa}{2}\right)\phi_c + \left(\frac{1-g+g\kappa}{2}\right)\phi_d \tag{2.32}$$

and

$$\frac{d\phi}{dx}\bigg|_{f_R} \approx \left(\frac{-g}{(1-g)\,\Delta x}\right)\phi_c + \left(\frac{g}{(1-g)\,\Delta x}\right)\phi_d \tag{2.33}$$

The following nodal advection discretization now results as:

$$\begin{aligned} u_x \left.\frac{d\phi}{dx}\right|_c \approx & \frac{g}{\Delta x}\left[-u_x\left(-1+f+\kappa-f\kappa\right)\right]\phi_{uu} \\ & + \frac{g}{\Delta x}\left[u_x\left(-3+g+2\kappa-g\kappa\right)\right]\phi_u \\ & + \frac{g}{\Delta x}\left[u_x\left(-\kappa+1+f-f\kappa\right)\right]\phi_c \\ & + \frac{g}{\Delta x}\left[u_x\left(1-g+g\kappa\right)\right]\phi_d \end{aligned} \tag{2.34}$$

The complete advection-diffusion nodal discretization is similarly found as:

$$\begin{aligned} u_x \left.\frac{d\phi}{dx}\right|_c - k\left.\frac{d^2\phi}{dx^2}\right|_c \approx & \frac{g}{\Delta x}\left[-u_x\left(-1+f+\kappa-f\kappa\right)\right]\phi_{uu} \\ & + \frac{g}{\Delta x}\left[u_x\left(-3+g+2\kappa-g\kappa\right)-\frac{2k}{\Delta x}\right]\phi_u \\ & + \frac{g}{\Delta x}\left[u_x\left(-\kappa+1+f-f\kappa\right)+\frac{2k}{(1-g)\,\Delta x}\right]\phi_c \\ & + \frac{g}{\Delta x}\left[u_x\left(1-g+g\kappa\right)-\frac{2gk}{(1-g)\,\Delta x}\right]\phi_d \end{aligned} \tag{2.35}$$

By applying the EAP to Equation (2.34), the pure advection error at a node results as:

$$\begin{aligned} & u_x \left.\frac{d\phi}{dx}\right|_c = \\ & \frac{g}{\Delta x}\left[-u_x\left(-1+f+\kappa-f\kappa\right)\right]\phi_{uu} + \frac{g}{\Delta x}\left[u_x\left(-3+g+2\kappa-g\kappa\right)\right]\phi_u \\ & +\frac{g}{\Delta x}\left[u_x\left(-\kappa+1+f-f\kappa\right)\right]\phi_c + \frac{g}{\Delta x}\left[u_x\left(1-g+g\kappa\right)\right]\phi_d \\ & -u_x\left(\frac{1}{2}\right)\left(\frac{g(\kappa-1)}{f-1}+\frac{1}{g}+\kappa-3\right)\Delta x\left.\frac{d^2\phi}{dx^2}\right|_c \\ & -u_x\left(\frac{1}{6}\right)\left(\frac{g\left(f^2-2f+2\right)\left(\kappa-1\right)}{(f-1)^2}+\frac{1}{g^2}+\frac{\kappa-4}{g}-3\kappa+6\right)\Delta x^2\left.\frac{d^3\phi}{dx^3}\right|_c \\ & -\frac{u_x}{24}\left(\frac{g(\kappa-1)}{(f-1)^3}+\frac{(g-1)^4(g(\kappa-1)+1)}{g^3}-g^2\kappa+g^2+2g\kappa-3g\right)\Delta x^3\left.\frac{d^4\phi}{dx^4}\right|_c \\ & -\dots \end{aligned} \tag{2.36}$$

As can be seen, all κ-Upwind Schemes achieve at least 1st-order accuracy at a node. As previously mentioned, depending on the stretching factors and κ value used, higher OoA's are also theoretically possible.

By applying the EAP to Equation (2.35), the total advection-diffusion error at a node results as:

$$\begin{aligned}
& u_x \left.\frac{d\phi}{dx}\right|_c - k \left.\frac{d^2\phi}{dx^2}\right|_c \\
&= \frac{g}{\Delta x}\left[-u_x\left(-1+f+\kappa-f\kappa\right)\right]\phi_{uu} + \frac{g}{\Delta x}\left[u_x\left(-3+g+2\kappa-g\kappa\right) - \frac{2k}{\Delta x}\right]\phi_u \\
&\quad + \frac{g}{\Delta x}\left[u_x\left(-\kappa+1+f-f\kappa\right) + \frac{2k}{(1-g)\Delta x}\right]\phi_c \\
&\quad + \frac{g}{\Delta x}\left[u_x\left(1-g+g\kappa\right) - \frac{2gk}{(1-g)\Delta x}\right]\phi_d \\
&\quad - \left[u_x\left(\frac{1}{2}\right)\left(\frac{g(\kappa-1)}{f-1} + \frac{1}{g} + \kappa - 3\right)\left.\frac{d^2\phi}{dx^2}\right|_c + \frac{(2g-1)k}{3g}\left.\frac{d^3\phi}{dx^3}\right|_c\right]\Delta x \qquad (2.37) \\
&\quad - u_x\left(\frac{1}{6}\right)\left(\frac{g\left(f^2-2f+2\right)(\kappa-1)}{(f-1)^2} + \frac{1}{g^2} + \frac{\kappa-4}{g} - 3\kappa + 6\right)\Delta x^2 \left.\frac{d^3\phi}{dx^3}\right|_c \\
&\quad + \left(\frac{(3g^2-3g+1)k}{12g^2}\right)\Delta x^2 \left.\frac{d^4\phi}{dx^4}\right|_c \\
&\quad - \frac{u_x}{24}\left(\frac{g(\kappa-1)}{(f-1)^3} + \frac{(g-1)^4(g(\kappa-1)+1)}{g^3} - g^2\kappa + g^2 + 2g\kappa - 3g\right)\Delta x^3 \left.\frac{d^4\phi}{dx^4}\right|_c \\
&\quad - \ldots
\end{aligned}$$

As previously, the leading error term is 1st-order accurate and contains both advection and diffusion terms. As a result, all κ-Upwind Schemes will achieve at least 1st-order advection-diffusion nodal accuracy. Higher OoA's are theoretically possible, but will only occur with a very convenient set of solution values and mesh conditions. Manipulating the κ values is, once again, not practically feasible due to the presence of the higher-order nodal gradients ($\left.\frac{d^2\phi}{dx^2}\right|_c$ and $\left.\frac{d^3\phi}{dx^3}\right|_c$).

2.6 Current Methods: A Critical Analysis

As explained in Chapter 1, six of the more popular and commonly used advection discretization methods can be expressed as members of the κ-Upwind method. These schemes:

- CDS ($\kappa = 1$),
- CUI $\left(\kappa = \frac{1}{3}\right)$,
- Fromm's ($\kappa = 0$),

- LUI ($\kappa = -1$),
- NVSF $\left(\kappa_{f_L} = \dfrac{3f-1}{2f},\ \ \kappa_{f_R} = \dfrac{3g-1}{2g}\right)$, and
- QUICK $\left(\kappa = \dfrac{1}{2}\right)$,

The OoA of each scheme, for both pure advection and combined advection-diffusion accuracy, is now evaluated at an internal face (using Section 2.5.1) and at an internal node (using Section 2.5.3) (boundary-adjacent-faces are additionally carried out in Appendix B). In order to extend on the understanding of each scheme, this will be done for equispaced, constantly stretched, and non-equispaced meshes.

For completeness, in Tables 2.4, 2.5, and 2.6, each OoA is accompanied by a superscript number which corresponds to the actual leading error term (given in Appendix C). When a scheme resulted in an OoA greater than that achieved by the other schemes, the OoA appears in bold.

Table 2.4 contains the OoA's resulting from the various schemes when applied on equispaced grids. By definition, an equispaced mesh results when $f = g = 0.5$.

TABLE 2.4: OoA's of various advection discretization schemes on equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
CDS ($\kappa = 1$)	Internal Face	2nd^{1}	2nd^{2}
	Internal Node	2nd^{3}	2nd^{4}
CUI $\left(\kappa = \frac{1}{3}\right)$	Internal Face	2nd^{5}	2nd^{6}
	Internal Node	**3rd**^{7}	2nd^{8}
Fromm's ($\kappa = 0$)	Internal Face	2nd^{9}	2nd^{10}
	Internal Node	2nd^{11}	2nd^{12}
LUI ($\kappa = -1$)	Internal Face	2nd^{13}	2nd^{14}
	Internal Node	2nd^{15}	2nd^{16}
NVSF $\left(\kappa_L = \kappa_R = \frac{1}{2}\right)$	Internal Face	**3rd**^{17}	2nd^{18}
	Internal Node	2nd^{19}	2nd^{20}
QUICK $\left(\kappa = \frac{1}{2}\right)$	Internal Face	**3rd**^{17}	2nd^{18}
	Internal Node	2nd^{19}	2nd^{20}

Superscript values correspond to error terms in Appendix C.

As can be observed, all κ-Upwind schemes achieve their claimed advection accuracies; CUI achieves 3rd-order nodal accuracy, QUICK and NVSF achieve identical 3rd-order face accuracy, and all remaining κ-Upwind schemes achieve 2nd-order face and nodal accuracy. As expected, CUI, QUICK, and NVSF do not manage to maintain higher-order accuracies when total advection-diffusion accuracy is considered.

The case of a constantly stretched mesh is considered next. It must be noted that while constant stretching is technically a type of non-equispacing, it does generally result in a somewhat simplified discretization. Hence, it is assessed here as its own case. Constant stretching occurs when $0 < f < 1$, $0 < g < 1$, and $f = g \neq 0.5$. By replacing every f with g, and by substituting the various κ values into the appropriate error expressions, the OoA's shown in Table 2.5 are acquired. Note that face accuracies are not presented due to the presence of only one stretching factor per face (i.e. face accuracies are considered in the generic non-equispaced case).

Table 2.5: OoA's of various advection discretization schemes on constantly stretched grids. Criteria shown in brackets is the constant stretching ratio at which the OoA holds (a single OoA greater is achieved when the criteria is not met).

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
CDS $(\kappa = 1)$	Internal Node	1st^{21}	1st^{22}
CUI $\left(\kappa = \frac{1}{3}\right)$	Internal Node	1st^{42} $\left(g \neq \frac{3}{5}\right)$	1st^{43}
Fromm's $(\kappa = 0)$	Internal Node	1st^{44}	1st^{45}
LUI $(\kappa = -1)$	Internal Node	1st^{46} $\left(g \neq \frac{1}{3}\right)$	1st^{47}
NVSF $\left(\kappa = \frac{3g-1}{2g}\right)$	Internal Node	1st^{36}	1st^{37}
QUICK $\left(\kappa = \frac{1}{2}\right)$	Internal Node	1st^{48} $\left(g \neq \frac{2}{3}\right)$	1st^{49}

Superscript values correspond to error terms in Appendix C.

As can be seen, every κ-Upwind scheme is reduced to 1st-order nodal accuracy (both advection and total accuracy). This shows that, while the κ-Upwind schemes (particularly CUI) are adept at achieving decent accuracies on equispaced grids, this accuracy is lost as soon as mesh stretching occurs. This is also the case with NVSF.

Table 2.6 contains the OoA's resulting from the various schemes on non-equispaced grids. By definition, a non-equispaced grid results when $0 < f < 1$ ($f \neq 0.5$) and $0 < g < 1$ ($g \neq 0.5$).

Table 2.6: OoA's of various advection discretization schemes on non-equispaced grids. Criteria shown in brackets are the stretching ratios at which the OoA holds (a single OoA greater is achieved when the criteria is not met).

Scheme	**Mesh Location**	**OoA of:** **Adv.**	**OoA of:** **Adv.-Dif.**
CDS ($\kappa = 1$)	Internal Face	2nd[1]	2nd[2]
	Internal Node	1st[21]	1st[22]
CUI $\left(\kappa = \frac{1}{3}\right)$	Internal Face	2nd[23] $\left(f \neq \frac{3}{7}\right)$	2nd[24]
	Internal Node	1st[25]	1st[26]
Fromm's ($\kappa = 0$)	Internal Face	2nd[27] $\left(f \neq \frac{1}{3}\right)$	2nd[28]
	Internal Node	1st[29]	1st[30]
LUI ($\kappa = -1$)	Internal Face	2nd[31] $\left(f \neq \frac{1}{5}\right)$	2nd[32]
	Internal Node	1st[33]	1st[34]
NVSF $\left(\kappa_L = \frac{3f-1}{2f}\right)$	Internal Face	**3rd**[35]	2nd[18]
	Internal Node	1st[36]	1st[37]
QUICK $\left(\kappa = \frac{1}{2}\right)$	Internal Face	2nd[38]	2nd[39]
	Internal Node	1st[40]	1st[41]

Superscript values correspond to error terms in Appendix C.

As expected, all schemes achieve only 1st-order accuracy. In general, the face accuracies in comparison to the equispaced face accuracies are relatively unchanged. The crucial exception, however, is that QUICK loses its face accuracy superiority and joins the other κ-Upwind schemes with only 2nd-order accuracy. NVSF lives up to its design intentions and achieves 3rd-order advection face accuracy. Once again, no scheme manages to achieve a higher-order advection-diffusion accuracy.

2.6.1 Weaknesses of Current Schemes

In conclusion, the following weaknesses exist with current advection discretization technology (in no particular order):

- OoA's greater than 3rd-order are not possible at internal faces. Further, OoA's greater than 2nd-order are not possible at upwind-boundary-adjacent faces (see Appendix B), regardless of the scheme used or the mesh conditions present.
- All the schemes are reduced to first order-accuracy on meshes which are not equispaced.

This work embarks on the task of designing an advection discretization scheme that addresses the above.

Chapter 3

Development of Enhanced Taylor Discretization Schemes

As described in Chapter 2, currently used advection discretization schemes mostly reduce to first order-accuracy on non-equispaced meshes. This chapter sees the development of two different schemes, one that attempts to improve the overall advection-diffusion accuracy, and the other which attempts to improve the pure advection accuracy.

In the interest of readability, development and analysis of the new methods are placed in seperate sections. It is understood that both schemes will be developed in an attempt to maximise accuracy, however, the exact OoA's will be assessed in Chapter 4.

3.1 A Higher-Order Advection-Diffusion ET Scheme

As already mentioned, when attempting to approximate an advection value at a face, only data from the two nodes connecting to the face's edge are to be used. Considering the left face in Figure 3.1, this essentially means that only the following data from nodes u and c are available for use (in a 1D case):

- Function values

- Computed gradients
- Up and downwind mesh stretching factors (which are computed as detailed later)
- Edge length

Despite the presence and availability of the above data at every face, no method utilises the downwind nodal gradient ($\left.\frac{d\phi}{dx}\right|_c$) or downwind stretching factor (g). Further, only NVSF uses the upwind stretching factor (f). It therefore stands to reason that improvements in accuracy are possible if more of the available data is used in each face advection approximation. Let us begin by creating a generic left face approximation as follows:

$$\phi_{f_L} = A\phi_u + B\phi_c + C\left.\frac{d\phi}{dx}\right|_u \Delta x + D\left.\frac{d\phi}{dx}\right|_c \Delta x \tag{3.1}$$

where values A, B, C, and D are the unknown coefficients to be determined. The three mesh-type locations for which this is to be done are the internal face, the upwind-boundary-adjacent face, and the downwind-boundary-adjacent face. Note that the intention in computing the coefficient is to enhance accuracy via the use of TSE's. Hence the new scheme is dubbed the Enhanced Taylor (ET) scheme. Sub-schemes appropriate for each of these mesh locations are developed below.

3.1.1 Internal Face

Due to the addition of the downwind face gradient ($\left.\frac{d\phi}{dx}\right|_c$) the internal face stencil used by the ET Scheme is different to that of the stencil used by the κ-Upwinding Scheme (Figure 2.2). The internal face stencil is shown in Figure 3.1 below.

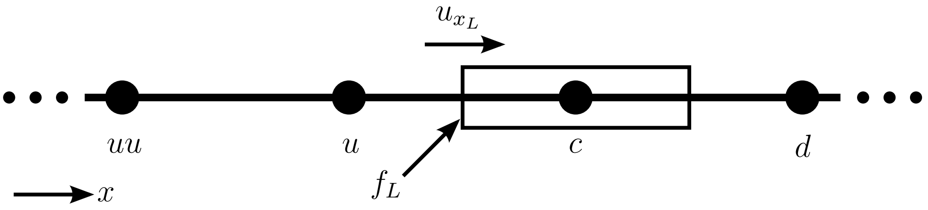

FIGURE 3.1: An ET Scheme internal face stencil.

Before the error analysis can proceed, the discrete expressions for the two gradients that appear in Equation (3.1) are to be developed. Here we employ the standard finite volume method as:

$$\left.\frac{d\phi}{dx}\right|_u \approx \frac{1-f}{\Delta x}\left(\phi_c - \phi_{uu}\right) \tag{3.2}$$

and

$$\left.\frac{d\phi}{dx}\right|_c \approx \frac{g}{\Delta x}\left(\phi_d - \phi_u\right) \tag{3.3}$$

The generic fully discrete expression for the internal face now follows as:

$$\phi_{f_L} \approx \left[-C(1-f)\right]\phi_{uu} + \left[A - Dg\right]\phi_u + \left[B + C(1-f)\right]\phi_c + \left[Dg\right]\phi_d \tag{3.4}$$

The expression resulting for the advection-diffusion equation results (via bringing in Equations (2.13) and (2.19)) as:

$$\begin{aligned} u_x\phi_{f_L} - k\left.\frac{d\phi}{dx}\right|_{f_L} \approx & \left[-u_xC(1-f)\right]\phi_{uu} + \left[u_xA - u_xDg + \frac{k}{\Delta x}\right]\phi_u \\ & + \left[u_xB + u_xC(1-f) - \frac{k}{\Delta x}\right]\phi_c + \left[u_xDg\right]\phi_d \end{aligned} \tag{3.5}$$

The EAP can now be applied to the above equation in order to acquire the optimum values for A, B, C, and D:

$$\begin{aligned} & u_x\left(A+B\right)\phi_{f_L} + \left(\frac{1}{2}\right)\left[u_x\left(-A+B+2C+2D\right) - \frac{2k}{\Delta x}\right]\Delta x\left.\frac{d\phi}{dx}\right|_{f_L} \\ & = \left[-u_xC(1-f)\right]\phi_{uu} + \left[u_xA - u_xDg + \frac{k}{\Delta x}\right]\phi_u \\ & \quad + \left[u_xB + u_xC(1-f) - \frac{k}{\Delta x}\right]\phi_c + \left[u_xDg\right]\phi_d \\ & \quad - u_x\left(\frac{1}{8}\right)\left[A+B+\left(\frac{4f}{f-1}\right)C + \left(\frac{4(1-g)}{g}\right)D\right]\Delta x^2\left.\frac{d^2\phi}{dx^2}\right|_{f_L} \\ & \quad - u_x\left(\frac{1}{48}\right)\left[-A+B+\left(\frac{6f^2+2}{(f-1)^2}\right)C + \left(\frac{2\left(3g^2-6g+4\right)}{g^2}\right)D\right]\Delta x^3\left.\frac{d^3\phi}{dx^3}\right|_{f_L} \\ & \quad + \left(\frac{1}{48}\right)\left(\frac{2k}{\Delta x}\right)\Delta x^3\left.\frac{d^3\phi}{dx^3}\right|_{f_L} - \dots \end{aligned} \tag{3.6}$$

The four unknown coefficients require four equations. The first two are found by examining the ϕ_{f_L} and $k \left.\dfrac{d\phi}{dx}\right|_{f_L}$ terms as:

$$u_x \left(A + B\right) \phi_{f_L} = u_x \phi_{f_L} \tag{3.7}$$

$$\left(\frac{1}{2}\right) \left[u_x \left(-A + B + 2C + 2D\right) - \frac{2k}{\Delta x}\right] \Delta x \left.\frac{d\phi}{dx}\right|_{f_L} = -k \left.\frac{d\phi}{dx}\right|_{f_L} \tag{3.8}$$

The remaining two equations are employed to increase accuracy via equating the two lowest-order error terms to zero. This would gaurantee 4th-order accuracy as:

$$u_x \left(\frac{1}{8}\right) \left[A + B + \left(\frac{4f}{f-1}\right) C + \left(\frac{4(1-g)}{g}\right) D\right] \Delta x^2 \left.\frac{d^2\phi}{dx^2}\right|_{f_L} = 0 \tag{3.9}$$

$$\begin{aligned} &u_x \left(\frac{1}{48}\right) \left[-A + B + \left(\frac{6f^2+2}{\left(f-1\right)^2}\right) C + \left(\frac{2\left(3g^2-6g+4\right)}{g^2}\right) D\right] \Delta x^3 \left.\frac{d^3\phi}{dx^3}\right|_{f_L} \\ &- \left(\frac{1}{48}\right) \left(\frac{2k}{\Delta x}\right) \Delta x^3 \left.\frac{d^3\phi}{dx^3}\right|_{f_L} = 0 \end{aligned} \tag{3.10}$$

Simplification of Equations (3.7) to (3.10) result in the following four equations that require simultaneous solution:

$$A + B = 1 \tag{3.11}$$

$$-A + B + 2C + 2D = 0 \tag{3.12}$$

$$A + B + \left(\frac{4f}{f-1}\right) C + \left(\frac{4(1-g)}{g}\right) D = 0 \tag{3.13}$$

$$-A + B + \left(\frac{6f^2+2}{(f-1)^2}\right) C + \left(\frac{2(3g^2-6g+4)}{g^2}\right) D = \frac{2k}{u_x \Delta x}, u_x \neq 0 \tag{3.14}$$

The resulting symbolic solutions are:

$D = \dfrac{g^2(2fk + fu_x\Delta x - 2k + u_x\Delta x)}{8u_x\Delta x(g-1)(fg-f+1)}$	$C = \left(\dfrac{(1-f)(1-g)}{fg}\right) D + \dfrac{1-f}{4f}$
$B = \dfrac{1}{2} - C - D$	$A = 1 - B$

The above expressions are only undefined when $u_x = 0$, which would also imply zero convection.

3.1.2 Upwind-Boundary-Adjacent Face

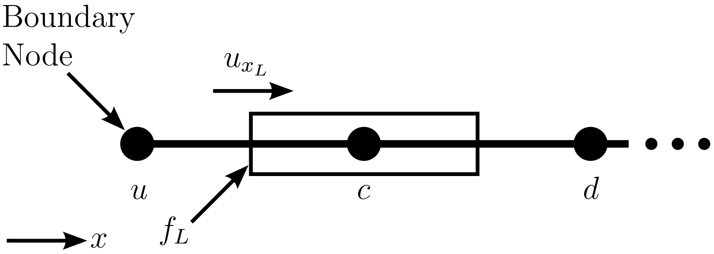

FIGURE 3.2: An ET Scheme upwind-boundary-adjacent face stencil.

The upwind-boundary-adjacent face is depicted in Figure 3.2. The ET scheme again includes the downwind node gradient ($\left.\frac{d\phi}{dx}\right|_c$). The upwind gradient is discretized as:

$$\begin{aligned}\left.\frac{d\phi}{dx}\right|_u &= \frac{1}{V_u}\int\limits_V \left(\frac{d\phi}{dx}\right) dV \\ &= \frac{2}{A\Delta x}\oint\limits_A (\phi n_x) dS \\ &= \frac{2}{A\Delta x}(\phi_{f_R} - \phi_{f_L})A \\ &\approx \frac{2}{\Delta x}\left(\frac{\phi_u + \phi_c}{2} - \phi_u\right) \\ &\approx \frac{1}{\Delta x}\left(\phi_c - \phi_u\right)\end{aligned} \tag{3.15}$$

This results in the following advection and advection-diffusion upwind-boundary-adjacent face discretizations, respectively:

$$\phi_{f_L} \approx [A - C - Dg]\,\phi_u + [B + C]\,\phi_c + [Dg]\,\phi_d \tag{3.16}$$

and

$$\begin{aligned}u_x\phi_{f_L} - k\left.\frac{d\phi}{dx}\right|_{f_L} \approx& \left[u_xA - u_xC - u_xDg + \frac{k}{\Delta x}\right]\phi_u \\ &+ \left[u_xB + u_xC - \frac{k}{\Delta x}\right]\phi_c + [u_xDg]\,\phi_d\end{aligned} \tag{3.17}$$

The EAP applied to the above advection-diffusion equation results in:

$$
\begin{aligned}
& u_x\,(A+B)\,\phi_{f_L} + \left(\frac{1}{2}\right)\left[u_x\,(-A+B+2C+2D) - \frac{2k}{\Delta x}\right]\Delta x\,\left.\frac{d\phi}{dx}\right|_{f_L} \\
&= \left[u_x A - u_x C - u_x D g + \frac{k}{\Delta x}\right]\phi_u + \left[u_x B + u_x C - \frac{k}{\Delta x}\right]\phi_c + \left[u_x D g\right]\phi_d \\
&\quad - u_x\left(\frac{1}{8}\right)\left[A+B+\left(\frac{4(1-g)}{g}\right)D\right]\Delta x^2\,\left.\frac{d^2\phi}{dx^2}\right|_{f_L} \\
&\quad - u_x\left(\frac{1}{48}\right)\left[-A+B+2C+\left(\frac{2\,(3g^2-6g+4)}{g^2}\right)D\right]\Delta x^3\,\left.\frac{d^3\phi}{dx^3}\right|_{f_L} \\
&\quad + \left(\frac{1}{48}\right)\left(\frac{2k}{\Delta x}\right)\Delta x^3\,\left.\frac{d^3\phi}{dx^3}\right|_{f_L} - \ldots
\end{aligned}
\tag{3.18}
$$

Following a similar process to that of the internal face, the following simultaneous equations now require solution:

$$A + B = 1 \tag{3.19}$$

$$-A + B + 2C + 2D = 0 \tag{3.20}$$

$$A + B + \left(\frac{4(1-g)}{g}\right)D = 0 \tag{3.21}$$

$$-A + B + 2C + \left(\frac{2(3g^2-6g+4)}{g^2}\right)D = \frac{2k}{u_x\Delta x},\ u_x \neq 0 \tag{3.22}$$

The above system of equations has a determinant of zero which results in singularity. As such the system cannot be solved. By dropping the highest-order error term (3.22), the system simplifies to three equations with four unknowns. C is conveniently set to zero (due to it being the coefficient of the boundary-node gradient), and the remaining three coefficients are acquired as:

$$
\begin{array}{ll}
D = \dfrac{g}{4(g-1)} & C = 0 \\
B = \dfrac{1}{2} - D & A = 1 - B
\end{array}
$$

It is interesting to note that the optimum advection discretization at an upwind-boundary-adjacent advection-diffusion face has no diffusion influence or reliance on the boundary gradient (due to the value of C equaling zero). In addition, the accuracy is now reduced to 3rd-order. Ultimately, the decrease in accuracy is understandable as an upwind-boundary-adjacent face has less information available due to the smaller stencil left of the face.

3.1.3 Downwind-Boundary-Adjacent Face

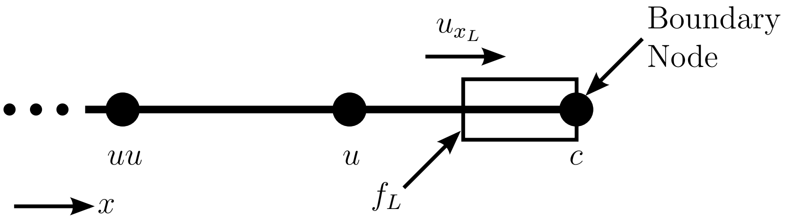

FIGURE 3.3: An ET Scheme downwind-boundary-adjacent face stencil.

The downwind-boundary-adjacent face (Figure 3.3) only has access to nodal information as per the κ-upwind scheme (i.e. no additional information exists due to the boundary).

With the boundary gradient being discretized as:

$$\left.\frac{d\phi}{dx}\right|_c \approx \frac{1}{\Delta x}\left(\phi_c - \phi_u\right) \tag{3.23}$$

the face value is computed for the advection and advection-diffusion cases as follows:

$$\phi_{f_L} \approx \left[-C(1-f)\right]\phi_{uu} + \left[A - D\right]\phi_u + \left[B + C(1-f) + D\right]\phi_c \tag{3.24}$$

and

$$\begin{aligned} u_x\phi_{f_L} - k\left.\frac{d\phi}{dx}\right|_{f_L} \approx & \left[-u_xC(1-f)\right]\phi_{uu} + \left[u_xA - u_xD + \frac{k}{\Delta x}\right]\phi_u \\ & + \left[u_xB + u_xC(1-f) + u_xD - \frac{k}{\Delta x}\right]\phi_c \end{aligned} \tag{3.25}$$

The EAP results in the following equation:

$$\begin{aligned}
&u_x\left(A+B\right)\phi_{f_L}+\left(\frac{1}{2}\right)\left[u_x\left(-A+B+2C+2D\right)-\frac{2k}{\Delta x}\right]\Delta x\left.\frac{d\phi}{dx}\right|_{f_L}\\
&=\left[-u_xC(1-f)\right]\phi_{uu}+\left[u_xA-u_xD+\frac{k}{\Delta x}\right]\phi_u\\
&+\left[u_xB+u_xC(1-f)+u_xD-\frac{k}{\Delta x}\right]\phi_c\\
&-u_x\left(\frac{1}{8}\right)\left[A+B+\left(\frac{4f}{f-1}\right)C\right]\Delta x^2\left.\frac{d^2\phi}{dx^2}\right|_{f_L}\\
&-u_x\left(\frac{1}{48}\right)\left[-A+B+\left(\frac{6f^2+2}{(f-1)^2}\right)C+2D\right]\Delta x^3\left.\frac{d^3\phi}{dx^3}\right|_{f_L}\\
&+\left(\frac{1}{48}\right)\left(\frac{2k}{\Delta x}\right)\Delta x^3\left.\frac{d^3\phi}{dx^3}\right|_{f_L}-\dots
\end{aligned} \tag{3.26}$$

The simplified equations required for simultaneous solving follows:

$$A+B=1 \tag{3.27}$$

$$-A+B+2C+2D=0 \tag{3.28}$$

$$A+B+\left(\frac{4f}{f-1}\right)C=0 \tag{3.29}$$

$$-A+B+\left(\frac{6f^2+2}{(f-1)^2}\right)C+2D=\frac{2k}{u_x\Delta x},u_x\neq 0 \tag{3.30}$$

As is the case for the upwind-boundary-adjacent face, Equations (3.27) to (3.30) are also singular. By setting $D=0$ (due to it being the coefficient of the boundary-node gradient), and reducing the system of equations to Equations (3.27) to (3.29), the system solves as:

$$\begin{array}{ll}
\hline
D=0 & C=\dfrac{1-f}{4f}\\
B=\dfrac{1}{2}-C & A=1-B\\
\hline
\end{array}$$

Similarly to that of the upwind-boundary-adjacent face, the optimum advection discretization at a downwind-boundary-adjacent advection-diffusion face has no diffusion influence or reliance on the boundary gradient. The accuracy is again only 3rd-order. These occurences are, once again, all quite understandable due to the absence of the downwind node (d).

3.2 A Higher-Order Advection ET Scheme

This section focuses on transforming the Higher-Order Advection-Diffusion ET scheme into a Higher-Order Advection ET scheme. This is important for flow regimes where little or no diffusion is present, as well as for comparison against the advection-diffusion ET scheme.

The advection only scheme is obtained easily by setting $k = 0$ in the advection-diffusion method. At an internal face the coefficient values are:

$$\begin{array}{ll}
D = \dfrac{g^2(f+1)}{8(g-1)(fg-f+1)} & C = \left(\dfrac{(1-f)(1-g)}{fg}\right) D + \dfrac{1-f}{4f} \\
B = \dfrac{1}{2} - C - D & A = 1 - B
\end{array}$$

At an upwind-boundary-adjacent face the coefficient values are:

$$\begin{array}{ll}
D = \dfrac{g}{4(g-1)} & C = 0 \\
B = \dfrac{1}{2} - D & A = 1 - B
\end{array}$$

At a downwind-boundary-adjacent face the coefficient values are:

$$\begin{array}{ll}
D = 0 & C = \dfrac{1-f}{4f} \\
B = \dfrac{1}{2} - C & A = 1 - B
\end{array}$$

3.2.1 Face Convection Approximation

In concluding the ET scheme development, it is similarly recommended that the speed of convection at a face (u_x) be similarly discretized to higher-order. We again commence by considering the following expression for the left face:

$$u_{x_{f_L}} = Au_{x_u} + Bu_{x_c} + C \left.\frac{du_x}{dx}\right|_u \Delta x + D \left.\frac{du_x}{dx}\right|_c \Delta x \tag{3.31}$$

where:

$$
\begin{aligned}
D &= \frac{g^2(f+1)}{8(g-1)(fg-f+1)} & C &= \left(\frac{(1-f)(1-g)}{fg}\right)D + \frac{1-f}{4f} \\
B &= \frac{1}{2} - C - D & A &= 1 - B
\end{aligned}
$$

This results in 4th-order accuracy regardless of the mesh stretching. The respective leading error terms are displayed in the table below:

TABLE 3.1: Leading error terms of the face convection approximation when utilising the Advection ET Scheme.

Mesh Type	**Leading Error Term**
Equispaced	$\left(\frac{3}{128}\right)\left.\frac{d^4u_x}{dx^4}\right\vert_{f_L}\Delta x^4$
Constantly-stretched	$-\left(\frac{(g+1)(2-g)}{384g(g-1)}\right)\left.\frac{d^4u_x}{dx^4}\right\vert_{f_L}\Delta x^4$
Non-equispaced	$-\left(\frac{(f+1)(2-g)}{384g(f-1)}\right)\left.\frac{d^4u_x}{dx^4}\right\vert_{f_L}\Delta x^4$

3.3 Extension to 2D and 3D

In order to maintain an industrially relevant scheme, it is necessary that the ET schemes be easily extendable to 2D and 3D unstructured meshes. The equivalence between the 1D and multi-dimensional case is defined per face as follows:

- ϕ_u and ϕ_c (no change)
- $\left.\frac{d\phi}{dx}\right\vert_u$ and $\left.\frac{d\phi}{dx}\right\vert_c$ becomes $\mathbf{t}\cdot\nabla\left(\phi_u\right)$ and $\mathbf{t}\cdot\nabla\left(\phi_c\right)$ respectively (nodal gradients in the direction of the edge's unit tangent vector, $\mathbf{t}$)
- Δx becomes l_{uc} (edge length)
- u_x becomes $\mathbf{t}\cdot u_f$ (advection value at the face, f, in the direction of the edge's unit tangent vector, $\mathbf{t}$)

What still remains to be computed are the stretching factors. This is detailed next.

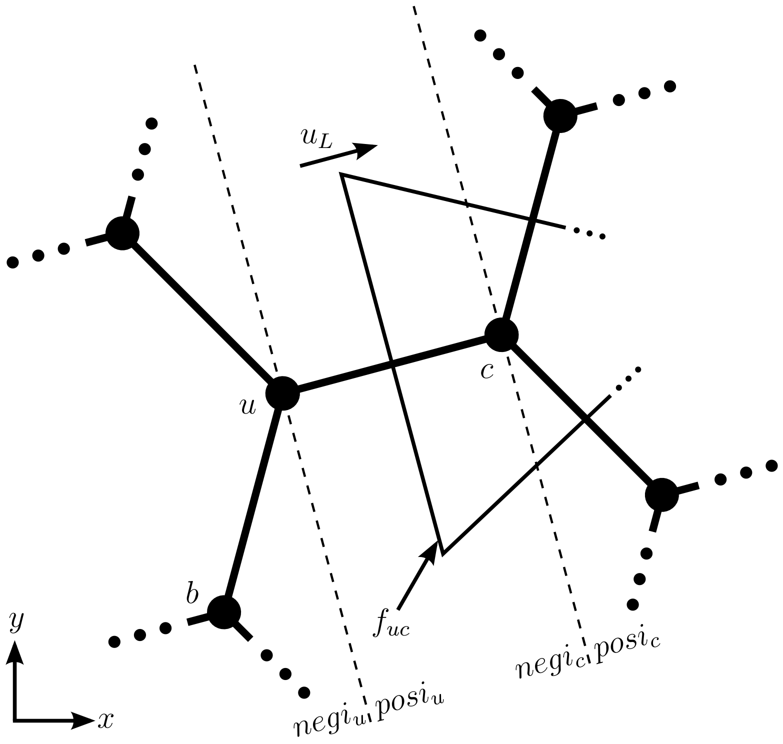

Figure 3.4: An equispaced hexagonal mesh.

Consider the equispaced hexagonal mesh depicted in Figure 3.4. The dotted lines represent perpendicular lines to edge uc (attached to nodes u and c). As the mesh is equispaced (edge lengths are constant throughout), it is desired that the mesh stretching factors f and g equate to 0.5. An algorithm is therefore required that accurately computes this while reverting back to the simple 1D solution when required.

Let us begin by defining two types of edges:

- Primary edge: An edge for which two nodal stretching factors are desired.
- Attached edge: An edge attached to a node of a primary edge.

Then define an arbitrary edge's (edge ub, for example) unit tangent vector as:

$$\mathbf{t_{ub}} = \frac{x_b - x_u}{l_{ub}}\hat{\mathbf{e}}_{\mathbf{x}} + \frac{y_b - y_u}{l_{ub}}\hat{\mathbf{e}}_{\mathbf{y}} \tag{3.32}$$

where $\hat{e}_x$ and $\hat{e}_y$ are the unit vector components in the x and y directions, respectively. This tangent vector is written for the 2D case, but is easily extended to 3D.

Let us continue by defining an attached edge's relative component (relative to that of a primary edge) as:

$$comp = l_{ub}\left(\mathbf{t_{ub}} \cdot \mathbf{t_{uc}}\right) \tag{3.33}$$

where $comp$ is the attached edge's relative component, l_{ub} is the length of the attached edge, $\mathbf{t_{ub}}$ is the unit tangent vector of the attached edge, and $\mathbf{t_{uc}}$ is the unit tangent vector of the primary edge. This calculation gives a sense of both the length of an attached edge, as well as to how aligned the edge is with the stretching direction. Edges aligned perfectly with the stretching direction will therefore have a component equivalent to their edge length. Conversely, edges aligned perpendicularly to the stretching direction will have a component of zero.

Finally, we define the stretching factors in a way that mimics the philosophy of the 1D stretching factors:

$$f = g = \frac{negi}{negi + posi} \tag{3.34}$$

where $negi$ represents the absolute sum of all upwind edge components ($comp$) attached to node u (absolute sum of all edge components on the upwind side of the perpendicular plane). Further, $posi$ represents the absolute sum of all downwind edge components (ie. absolute sum of all edge components on the downwind side of the perpendicular plane). It is important to note that when calculating the stretching factor at a node, components from all attached nodes must be considered, including the primary edge. It is also important to note that the component of the primary edge is on the downwind side of the upwind node, and on the upwind side of the downwind node.

Application of this algorithm at a node results in one of three possible values for stretching factor (f), regardless of the number of dimensions:

1. $f = 0$: Only possible at an upwind-boundary node.
2. $0 < f < 1$: All internal nodes and, in extremely rare cases, some boundary nodes.
3. $f = 1$: Only possible at a downwind-boundary node.

Chapter 4

Developed ET Schemes: A Critical Analysis

As done in Chapter 2 for the analysis of the selected κ-Upwind Schemes, the Advection-Diffusion ET Scheme and the Advection ET Scheme are analysed. Both OoA as well as odd-even decoupling is considered.

4.1 Accuracy

The computed OoA's are shown in Tables 4.1, 4.2, and 4.3. As before, the superscript number corresponds to a leading error term in Appendix C, and the localised accuracies of boundary-adjacent-faces are given in Appendix B. OoA's greater than that achieved by other schemes is once again expressed in bold.

An equispaced grid results when $f = g = h = 0.5$. These values are substituted into the appropriate error analyses (derived earlier in Chapter 3) in order to acquire the OoA's shown below.

Table 4.1: OoA's of the ET Schemes on equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
Adv.-Dif. ET	Internal Face	**4th**^{50}	**4th**^{51}
	Internal Node	2nd^{53}	2nd^{54}
Adv. ET	Internal Face	**4th**^{55}	2nd^{18}
	Internal Node	2nd^{56}	2nd^{20}

Superscript values correspond to error terms in Appendix C.

Both ET schemes were primarily designed to achieve improved face accuracies when applied to non-equispaced grids. It is therefore pleasing to observe that both schemes achieve at least a single OoA improvement over the next best κ-Upwind method (Advection-Diffusion ET scheme achieves a two OoA improvement). Disappointingly, however, the improved face accuracies do not carry over to an improved nodal accuracy.

The special case of a constantly stretched grid occurs when $f = g = h \neq 0.5$.

Table 4.2: OoA's of the ET Schemes on constantly stretched grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
Adv.-Dif. ET	Internal Face	3rd^{66}	**3rd**^{67}
	Internal Node	1st^{61}	1st^{62}
Adv. ET	Internal Face	**4th**^{68}	2nd^{18}
	Internal Node	**2nd**^{64}	1st^{65}

Superscript values correspond to error terms in Appendix C.

Both ET Schemes maintain at least a single OoA improvement over the next best κ-Upwind method at a face for constantly stretched meshes. Additionally, the Advection ET Scheme also now manages to achieve an advection OoA improvement at an internal node.

Finally, a non-equispaced grid results when $0 < f < 1$ ($f \neq 0.5$), $0 < g < 1$ ($g \neq 0.5$), and $0 < h < 1$ ($h \neq 0.5$).

Table 4.3: OoA's of the ET Schemes on non-equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
Adv.-Dif. ET	Internal Face	3rd[57]	**3rd**[58]
	Internal Node	1st[61]	1st[62]
Adv. ET	Internal Face	**4th**[63]	2nd[18]
	Internal Node	**2nd**[64]	1st[65]

Superscript values correspond to error terms in Appendix C.

The improvements shown for the constantly stretched case are similarly present in the generalised non-equispaced case. This is important as it shows that the Advection ET Scheme is formally 2nd-order advection accurate at an internal node.

4.1.1 Conclusion

The patterns observed in the achieved accuracies of both the Advection-Diffusion ET Scheme and the Advection ET Scheme are generally consistent with the design intentions of both schemes. When observed on non-equispaced grids (Tables 4.3 and 4.2) the advection accuracy of the Advection ET Scheme is consistently greater than that of the Advection-Diffusion ET Scheme, whereas, the opposite is true for the total advection-diffusion accuracy. On an equispaced grid (Table 4.1), however, the Advection-Diffusion ET Scheme performs beyond expectations by matching the advection accuracy, and greatly beating the advection-diffusion accuracy, of the Advection ET Scheme.

The true extent to which the ET Schemes achieve their respective design goals is revealed via comparison with the κ-Upwind Schemes (shown in Chapter 2). The Advection-Diffusion ET Scheme consistently improves on both advection and advection-diffusion accuracies achieved by the κ-Upwind Schemes, whereas the Advection ET Scheme greatly improves on the κ-Upwind Schemes' advection accuracy. However, as will be shown next, equispaced meshes result in odd-even decoupling, which is undesirable and needs to be avoided.

4.2 Odd-Even Decoupling

Odd-even decoupling classically results when variations in the field variable between adjacent nodes are not adequately represented by the discretization method. Proving complete avoidance of this for the general case is cumbersome. It is however instructive to study its likelihood via simple cases as decoupling is typically an underlying quality of the discretization method. We will therefore consider a 1D constant stretching case where $u_{x_L} = u_{x_R} = u_x$.

Both ET schemes are assessed for advection odd-even decoupling here, whereas, the Advection-Diffusion ET Scheme is additionally assessed for diffusion odd-even decoupling in Appendix D (due to its additional influence on the diffusion term). The simplified advection and advection-diffusion discretizations at node c, respectively, will be used as follows:

$$\begin{aligned}\phi_{f_R} - \phi_{f_L} \approx &\left[C_{f_L}(1-g)\right]\phi_{uu} + \left[-C_{f_R}(1-g) - A_{f_L} + D_{f_L}g\right]\phi_u \\ &+ \left[A_{f_R} - D_{f_R}g - B_{f_L} - C_{f_L}(1-g)\right]\phi_c \\ &+ \left[B_{f_R} + C_{f_R}(1-g) - D_{f_L}g\right]\phi_d + \left[D_{f_R}g\right]\phi_{dd}\end{aligned} \tag{4.1}$$

and

$$\begin{aligned}&u_x\left(\phi_{f_R} - \phi_{f_L}\right) - k\left(\left.\frac{d\phi}{dx}\right|_{f_R} - \left.\frac{d\phi}{dx}\right|_{f_L}\right) \\ &\approx \left[u_x\left(C_{f_L}(1-g)\right)\right]\phi_{uu} + \left[u_x\left(-C_{f_R}(1-g) - A_{f_L} + D_{f_L}g\right) - \frac{k}{\Delta x}\right]\phi_u \\ &\quad + \left[u_x\left(A_{f_R} - D_{f_R}g - B_{f_L} - C_{f_L}(1-g)\right) + \frac{k}{(1-g)\Delta x}\right]\phi_c \\ &\quad + \left[u_x\left(B_{f_R} + C_{f_R}(1-g) - D_{f_L}g\right) - \frac{gk}{(1-g)\Delta x}\right]\phi_d + \left[u_x\left(D_{f_R}g\right)\right]\phi_{dd}\end{aligned} \tag{4.2}$$

Although odd-even decoupling is strictly only concerned with the coefficient of the centre node (c), the analyses in this section will take a more comprehensive approach in order to better understand the ET Schemes. Therefore, the coefficients at all nodes will be analysed and assessed. It is important to remember that g only makes sense when $0 < g < 1$, therefore, only values within this range will be considered when observing values that reduce the coefficients to zero.

4.2.1 Advection-Diffusion ET Scheme: Advection Decoupling

The optimum Advection-Diffusion ET Scheme values from Section 3.1.1 are substitued into the coefficients of Equation (4.1) in order to acquire the final advection coefficients of the Advection-Diffusion ET Scheme. These resulting coefficients are displayed and analysed below.

Coefficient of ϕ_{uu}:

$$\frac{(g-1)^3[-2gk+u_x\Delta x(g-2)]}{8g(g^2-g+1)u_x\Delta x} \tag{4.3}$$

Coefficient of ϕ_u:

$$\frac{2gk(g-1)+u_x\Delta x(3g^3-9g^2+8g-4)}{8g(g^2-g+1)u_x\Delta x} \tag{4.4}$$

Coefficient of ϕ_c:

$$\frac{4g^2k(g-1)+u_x\Delta x(-2g^3+3g^2+3g-2)}{8g(g-1)u_x\Delta x} \tag{4.5}$$

Coefficient of ϕ_d:

$$\frac{2g^2k+u_x\Delta x(-3g^3+g-2)}{8(g-1)(g^2-g+1)u_x\Delta x} \tag{4.6}$$

Coefficient of ϕ_{dd}:

$$-\frac{g^3[2gk-u_x\Delta x(g+1)]}{8(g-1)(g^2-g+1)u_x\Delta x} \tag{4.7}$$

When considering the complete coefficients shown above, all of them are non-zero and defined for all realistic values of g. However, when $g = 0.5$ (ie. an equispaced grid), and the discretization is completed by multiplying through by u_x, the coefficient of ϕ_c simplifies to:

$$\frac{k}{4\Delta x} \tag{4.8}$$

As can be observed, this coefficient contains no "pure" advection component (ie. it is independent of u_x), so while it may not be strictly zero, it will likely still lead to odd-even decoupling.

4.2.2 Advection ET Scheme: Advection Decoupling

By setting the value of k to zero for all of the discretization coefficients calculated in Section 4.2.1 above, the resulting advection coefficients of Equation (4.1) for the Advection ET Scheme are acquired. These coefficients are displayed and analysed below.

Coefficient of ϕ_{uu}:

$$\frac{(g-1)^3(g-2)}{8g(g^2-g+1)} \tag{4.9}$$

Coefficient of ϕ_u:

$$\frac{3g^3-9g^2+8g-4}{8g(g^2-g+1)} \tag{4.10}$$

Coefficient of ϕ_c:

$$\frac{-2g^3+3g^2+3g-2}{8g(g-1)} \tag{4.11}$$

Coefficient of ϕ_d:

$$\frac{-3g^3+g-2}{8(g-1)(g^2-g+1)} \tag{4.12}$$

Coefficient of ϕ_{dd}:

$$\frac{g^3(g+1)}{8(g-1)(g^2-g+1)} \tag{4.13}$$

All of the above coefficients, with the important exception of the ϕ_c coefficient, are non-zero and defined for all realistic values of g. The ϕ_c coefficient, however, simplifies to zero when $g = 0.5$ (ie. an equispaced grid). This indicates that odd-even decoupling will be a problem for the Advection ET Scheme on equispaced grids.

Odd-even decoupling will clearly be problematic on equispaced meshes. This is unideal as numerical methods are traditionally at their most accurate when applied to equispaced meshes. As a result, blending of the ET Schemes with currently available schemes is necessary to overcome these shortfalls. This blending process is expressed next.

4.3 Blending to Counteract Non-Physical Oscillations

Odd-even decoupling is not generally an issue for κ-Upwind schemes due to their asymmetrical and upwind bias (a notable exception being CDS as it is not strictly a κ-Upwind Scheme). Additionally, during initial testing, it was observed that the ET schemes produced oscillatory solutions on equi- and near-equispaced meshes as well as at high advection-diffusion ratios. As a result, a blend of the ET schemes with κ-Upwind schemes is necessary to achieve an accurate and stable numerical scheme.

Ideally, it is desired that the blend resort fully to the selected κ-Upwind scheme on equispaced grids (ie. $f = 0.5$) while tending to the ET schemes once sufficient mesh stretching is present. Additionally, it is necessary that the blend resort to the κ-Upwind scheme once a certain advection-diffusion ratio is hit. As this work is focused primarily on the application and accuracy of the ET Schemes on non-equispaced grids, a pragmatic approach was taken as to which κ-Upwind Scheme to select and to what cutoff values for mesh stretching and advection-diffusion ratios to use.

NVSF was selected as the κ-Upwind Scheme for blending due to its suitability to stretched meshes while reverting to QUICK on equispaced meshes. Through trial and error in solving the linear advection-diffusion equation, cutoff values of 0.545 and 50 were chosen for the mesh stretching factor and advection-diffusion ratio respectively. This cutoff stretching factor is equivalent to a traditional 'growth' factor of 1.2. Written mathematically, the blending algorithm used in this work follows:

$$Blend = (\pi)NVSF + (1 - \pi)ET \tag{4.14}$$

where:

$$\pi = \pi_s + (1 - \pi_s)\pi_r \tag{4.15}$$

and π_s and π_r represent the components of the blend derived from the stretching factor and advection-diffusion ratio respectively. They are calculated as:

$$\pi_s = \begin{cases} 1 - \left| \dfrac{f - 0.5}{0.044} \right|, & \text{if } 0.456 < f < 0.544. \\ 0, & \text{otherwise.} \end{cases} \tag{4.16}$$

$$\pi_r = \begin{cases} \dfrac{u_x}{50k}, & \text{if } \left|\dfrac{u_x}{k}\right| < 50. \\ 1, & \text{otherwise.} \end{cases} \tag{4.17}$$

It is important to note that the blend is continuous and smooth. This is necessary to minimise adverse impact of the blending on the numerical solution. Additionally, it is clear to see that the scheme reverts to NVSF when $f = 0.5$ (ie. equispaced) or $\dfrac{u_x}{k} \geq 50$.

Chapter 5

Numerical Examples

For scheme evaluation purposes, both 1D and 2D cases are considered, namely the advection-diffusion and incompressible flow lid-driven cavity respectively. In both cases, the ET schemes are compared to the κ-Upwind schemes renowned for their accuracy (namely CUI, QUICK, and NVSF), and the ET schemes have been suitably blended as described in Section 4.3.

5.1 1D Advection-Diffusion Equation

The 1D Advection-Diffusion equation, being the model equation for this work, is a natural choice of test case. The domain utilised extends between $x = 0$ and $x = 1$, with boundary values of $\phi_0 = 1$ and $\phi_1 = 0$ respectively. A generic analytical solution for a 1D advection-diffusion equation exists as follows:

$$\phi_{exact}(x) = \phi_0 + (\phi_1 - \phi_0)\left(\frac{e^{xu_x/k} - 1}{e^{Lu_x/k} - 1}\right) \tag{5.1}$$

where L denotes the length of the domain ($L = 1$). Substitution of the boundary conditions and $L = 1$ into the above equation results in:

$$\phi_{exact}(x) = 1 - \left(\frac{e^{xu_x/k} - 1}{e^{u_x/k} - 1}\right) \tag{5.2}$$

In setting up the advection-diffusion problem, there are essentially three variable metrics to consider; the advection-diffusion ratio, the constant nodal mesh stretching present, and the number of nodes in the domain. Extensive results are presented for the various combinations of three advection-diffusion ratios (10, 20, and 40 - analytical solutions shown in Figure 5.1), three constant mesh stretching factors (0.524, 0.545, and 0.565, which are equivalent to traditional 'growth' factors of 1.1, 1.2, and 1.3), and three mesh sizes (10, 30, and 90), in Appendix E. To follow is a summary. It is important to note that higher solution gradients are present near the right hand boundary and, as such, stretching is done from right to left in order to better represent the physics.

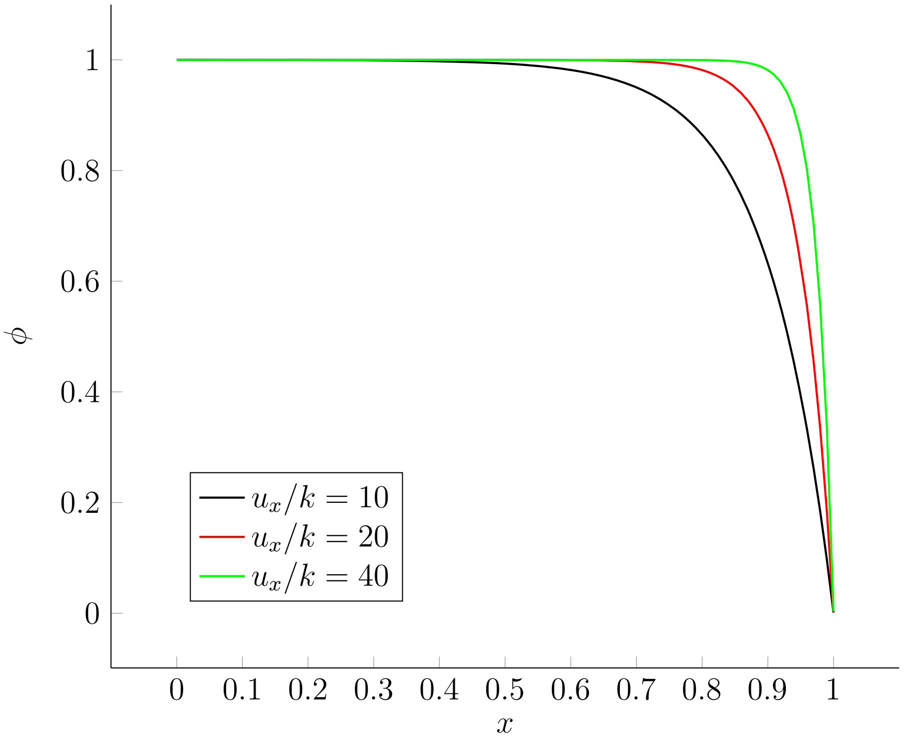

FIGURE 5.1: Analytical solutions of ϕ throughout the computational domain for three advection-diffusion ratios (10, 20, and 40).

Figures 5.2 and 5.3 show the actual plots of pre- and post-blended ET Advection-Diffusion schemes respectively, along with the κ-Upwind schemes and analytical solution to indicate how the blending has damped the oscillations. For this test case the 10 node, 1.1 growth factor mesh with an advection-diffusion ratio of 40 was by far the most oscillatory case tested.

As can be observed, blending helps the resulting ET schemes minimise oscillations to a level similar to that of the κ-Upwind schemes. Due to this success, the blending was maintained for each subsequent test case. The majority of the test cases resulted in similar patterns so, to avoid repetition, only two error plots and a summary of the data are presented here (as mentioned, extensive data appears in Appendix E).

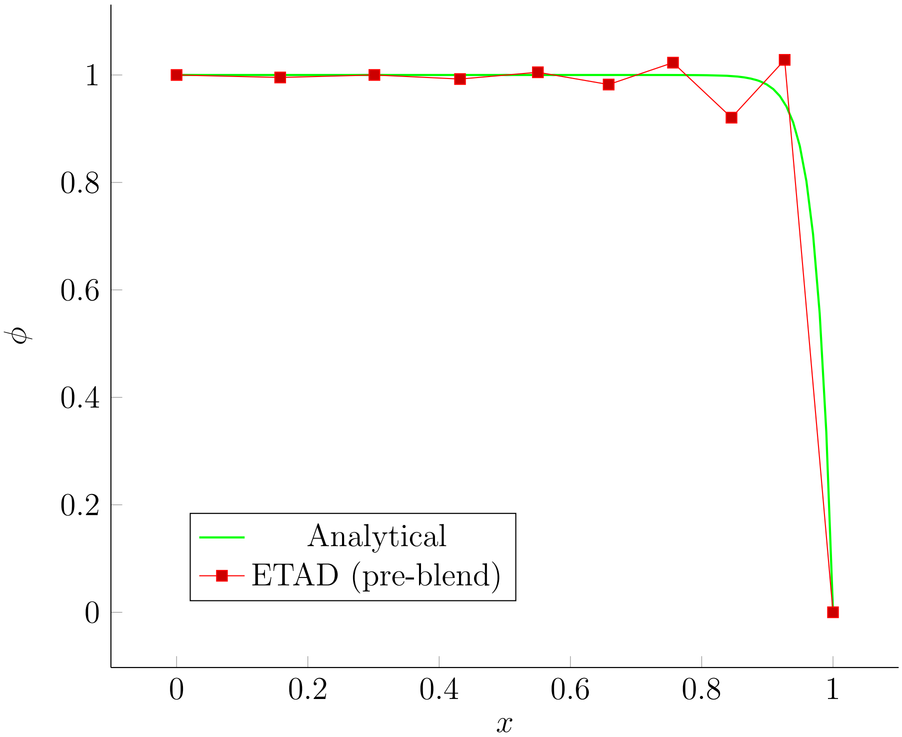

FIGURE 5.2: Un-blended ET Advection-Diffusion scheme on a 10 node, 1.1 growth factor mesh with an advection-diffusion ratio of 40.

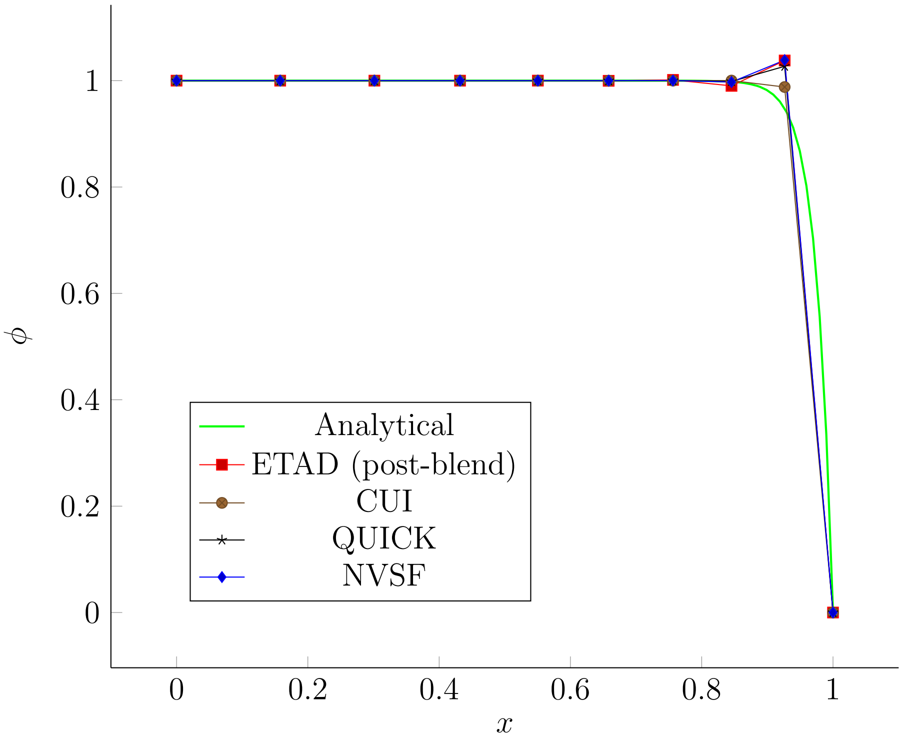

FIGURE 5.3: Effects of blending the ET Advection-Diffusion scheme with NVSF on a 10 node, 1.1 growth factor mesh with an advection-diffusion ratio of 40.

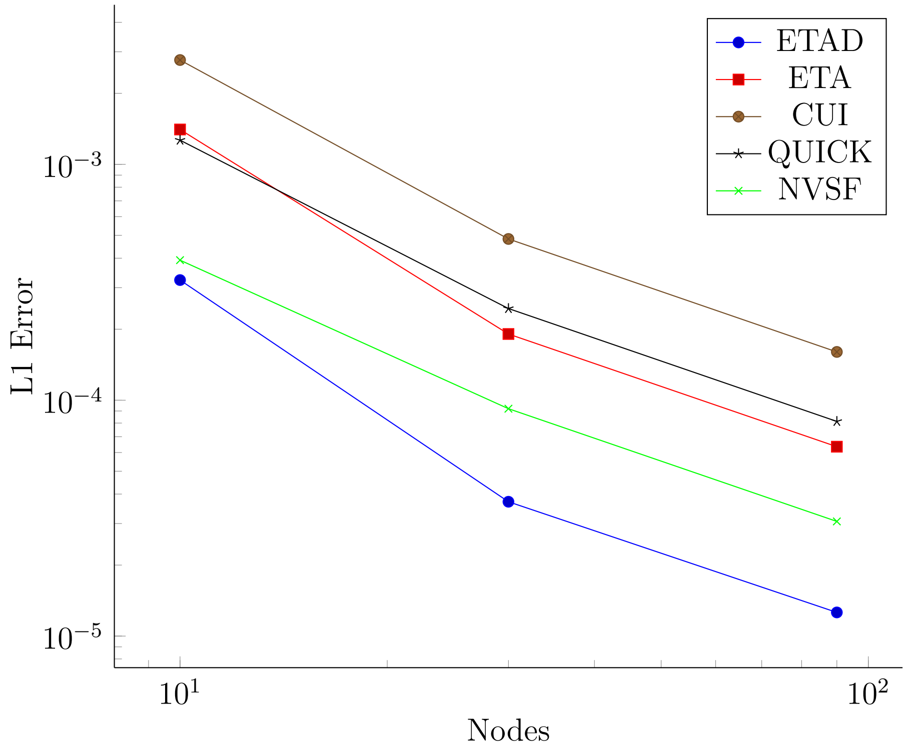

FIGURE 5.4: L1 Error vs Nodes for an advection-diffusion ratio of 10 and a constant mesh growth factor of 1.3.

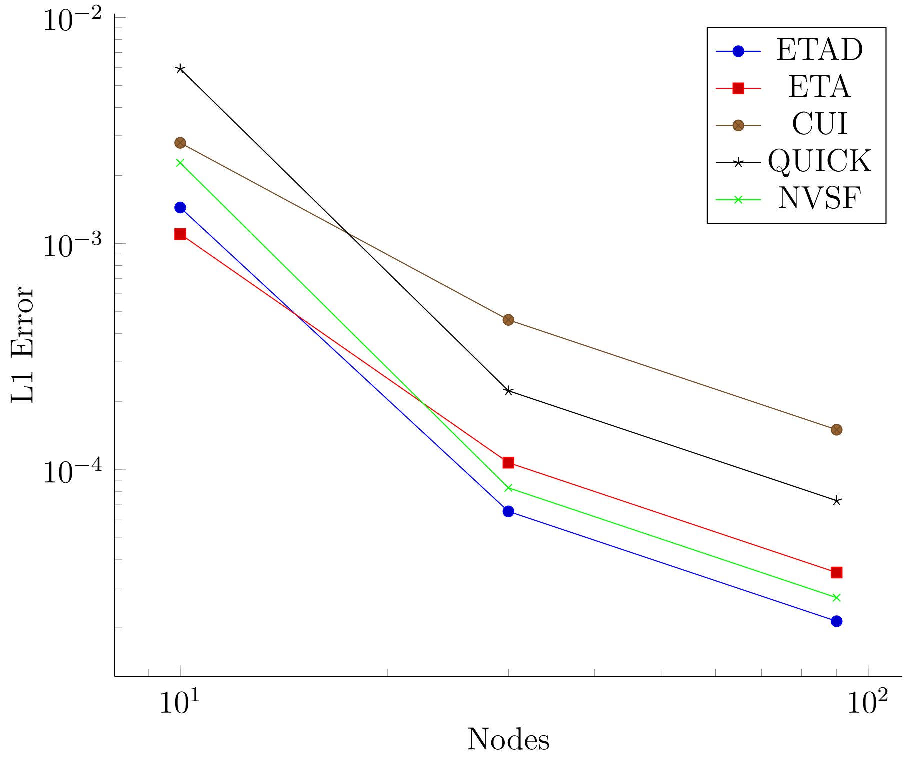

FIGURE 5.5: L1 Error vs Nodes for an advection-diffusion ratio of 40 and a constant mesh growth factor of 1.3.

Table 5.1: 1D Advection-Diffusion Equation summary of averaged error indices for each growth factor (averaged across advection-diffusion ratios), and error normalised to QUICK.

Growth Factor	**Error**	**Scheme:** **ET** (Adv. & Dif.)	 **ET** (Adv.)	 **CUI** $\left(\kappa = \frac{1}{3}\right)$	 **QUICK** $\left(\kappa = \frac{1}{2}\right)$	 **NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
1.1	Max. abs.	**69.54**	98.59	191.28	100.00	81.94
	L1-norm	**64.86**	98.10	198.28	100.00	79.86
	L2-norm	**67.56**	97.13	195.16	100.00	80.71
1.2	Max. abs.	**46.18**	110.73	206.27	100.00	66.95
	L1-norm	**52.10**	120.04	225.00	100.00	72.25
	L2-norm	**50.10**	115.03	215.93	100.00	68.93
1.3	Max. abs.	**25.32**	76.67	190.88	100.00	36.50
	L1-norm	**28.06**	78.77	194.37	100.00	36.60
	L2-norm	**25.85**	77.02	191.45	100.00	35.23
Combined	Max. abs.	**47.01**	95.33	196.14	100.00	61.80
	L1-norm	**48.34**	98.97	205.89	100.00	62.90
	L2-norm	**47.83**	96.39	200.85	100.00	61.62

As expected, the accuracy of the ET Advection-Diffusion increased relative to the other schemes as the growth factor increased. ET Advection, however, was inconsistent, achieving it's best results at the highest growth factor of 1.3. The former also competed better as diffusion was more dominant, while the ET advection scheme again behaved inconsistently.

Table 5.2: 1D Advection-Diffusion Equation summary of averaged error indices for each advection-diffusion ratio, and an overall average.

Adv.:Dif.	Error	**Scheme:** **ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
10	Max. abs.	**24.78**	97.64	217.23	100.00	47.80
	L1-norm	**28.76**	100.28	228.71	100.00	45.80
	L2-norm	**26.21**	98.38	223.22	100.00	45.97
20	Max. abs.	**50.29**	111.74	218.86	100.00	66.46
	L1-norm	**56.38**	125.27	236.83	100.00	73.87
	L2-norm	**53.96**	116.77	227.61	100.00	69.03
40	Max. abs.	**65.97**	76.62	152.34	100.00	71.13
	L1-norm	**59.88**	71.35	152.12	100.00	69.03
	L2-norm	**63.33**	74.03	151.70	100.00	69.87
Combined	Max. abs.	**47.01**	95.33	196.14	100.00	61.80
	L1-norm	**48.34**	98.97	205.89	100.00	62.90
	L2-norm	**47.83**	96.39	200.85	100.00	61.62

To summarise from the table, the ET Advection-Diffusion scheme performs well relative to all other available schemes. On average, decreases in error in excess of 75%, 50%, and 22% are observed relative to CUI, QUICK, and NVSF respectively. The ET Advection scheme however performs only marginally better than QUICK, and worse than NVSF. QUICK substantially outperforms CUI, with reductions in error up to 50%. Results for the 2D lid-driven cavity test case are presented next.

5.2 2D Lid-Driven Cavity

For the 2D incompressible test-case the Elemental™ software was employed. The incompressible flow solver incorporates a version of the UP-AC split method (Oxtoby and Malan [17]). The ET schemes were then simply applied to the discretization of the convective momentum term. In the x-direction, the following applies:

$$\nabla . \left(\rho u_x \mathbf{u}\right) \approx \frac{1}{V} \int_A \rho u_x \mathbf{u}.\mathbf{n} dA \tag{5.3}$$

where u_x and $\mathbf{u}$ are calculated using the ET scheme and face convective approximation respectively. Though a more thorough implementation is possible (via treatment of pressure at faces), the above implementation was deemed suitable for this study. Note that the developed ET schemes did not have a noticable effect on increasing solution times.

The 2D lid-driven cavity example considers steady recirculation flow in a quadralateral cavity. The recirculation flow is created by the uniform motion of the upper boundary, the "lid", while all other boundaries maintain a no-slip boundary condition. No flow is permitted either into or out of the cavity. A square cavity of length 1 was selected in this work in order to aid comparison to the results of Ghia et al. [10] and the numerical results of Erturk et al. [7]. The left bottom corner of the cavity is also placed at $(0, 0)$. Three Reynolds numbers (Re = 1000, 5000, and 10000) were chosen, both to aid comparison, as well as to reflect a broad range of advection-diffusion ratios. The Reynolds number is calculated as follows:

$$Re = \frac{UL}{\nu} \tag{5.4}$$

where U is the uniform velocity of the lid in ms^{-1}, L is the length of the lid in m, and ν is the kinematic viscosity of the fluid in the cavity in $\mathrm{m}^2\mathrm{s}^{-1}$. In this case, the latter is equivalent to k, the diffusion constant, due to the density (ρ) being set to one. Figure 5.6 graphically portrays flow in the Re = 10000 case through the use of a streamtracer diagram.

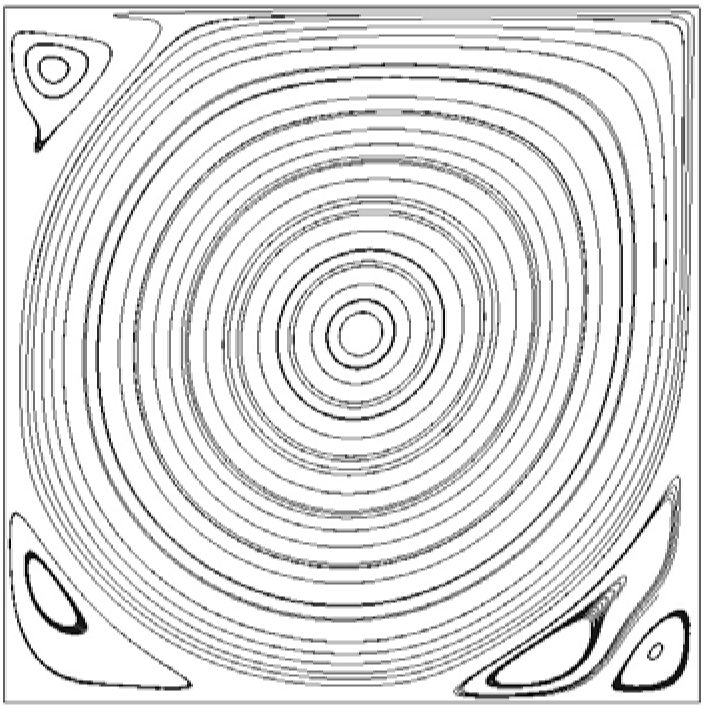

Figure 5.6: 2D Lid-Driven Cavity $Re = 10000$ streamtracer.

A study was carried out using the proprietary software of Elemental™. For the purpose of estimating error, a so-called benchmark solution was generated via a grid convergence study. This culminated in using a 251x251 mesh with 1.2 growth factor similar to shown in Figure 5.8. An odd number of nodes was purposely chosen in order to ensure that mesh nodes are present at $x = 0.5$. This central column has traditionally been the focus of error analyses, with the vast majority of journal articles observing and/or calculating the velocity on this line. As such, all error observations for the lid-driven cavity in this work will also focus on the x-direction velocity at $x = 0.5$.

The solutions acquired from the Elemental™ study showed near-perfect correlation to the experimental results of Ghia et al. [10], with the only exception being the solution of the $Re = 10000$ flow, which showed slight discrepancies near the lid boundary. When compared to the numerical results of Erturk et al. [7], however, near-perfect correlation was found for all Reynolds number solutions.

For each Reynolds number, the schemes were tested on three different mesh sizes with the resulting velocities compared to those of the benchmark solution. The meshes selected for testing were vastly coarser than the 251x251 mesh used for the benchmark study and contained roughly 10%, 15%, and 20% of the number of nodes. This related to meshes of size 81x81, 95x95 and 111x111, respectively. Each mesh was stretched with a constant stretching factor of $f = g = h = 0.45$ (growth factor of 1.2). As seen in Figure 5.7, even the coarsest meshes chosen (81x81 nodes), both structured and unstructured, were sufficient to represent the flow within the cavity. This helps add confidence to the acquired computational solutions. Examples of the anisotropic structured and unstructured meshes used are shown in Figures 5.8 and 5.9. Note that the mesh in Figure 5.9 is defined as being unstructured not for having arbitrarily shaped cell elements, but for possessing cell faces that have noncollinear upwind and downwind nodes. This ensures that discretizations and stretching factors are calculated in a generic and unstructured fashion.

Solution data of the individual test cases are very similar to one another, so only summaries are included here. See Appendix F for detailed data.

As expected, the accuracies of the ET schemes relative to the QUICK benchmark decreased as the Reynolds number increased (i.e. increasing advection-diffusion

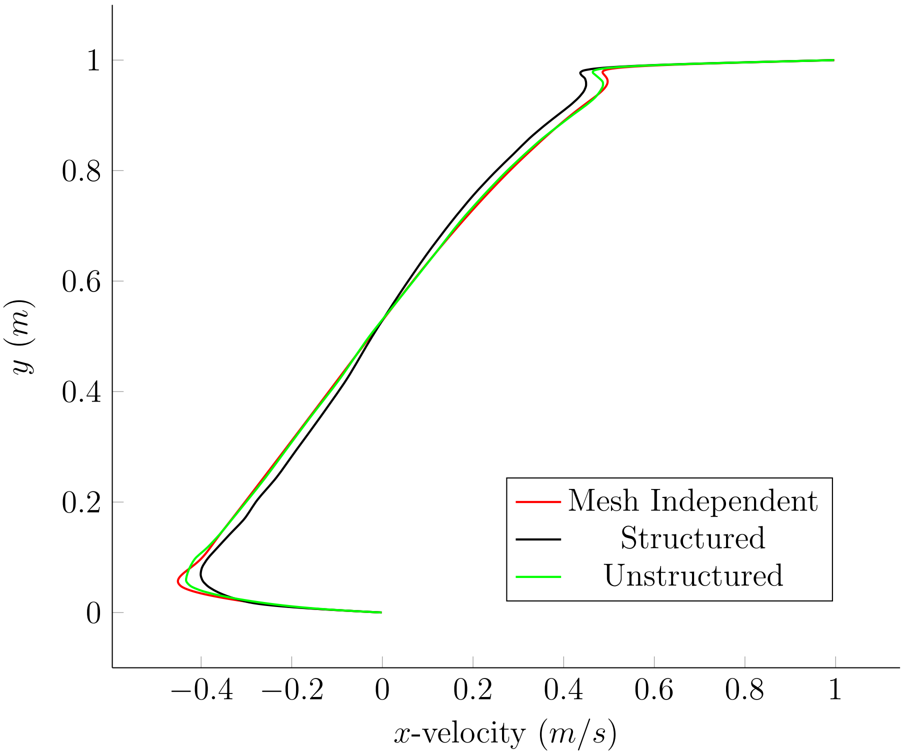

FIGURE 5.7: x-velocity at the $x = 0.5$ line for the Re=10000 mesh independent, ETAD 81x81 structured grid, and ETAD 81x81 unstructured grid solution.

ratio). Despite this, the ET schemes still maintained a reduction in error in excess of 27% relative to QUICK at the highest Reynolds number test case of $Re = 10000$.

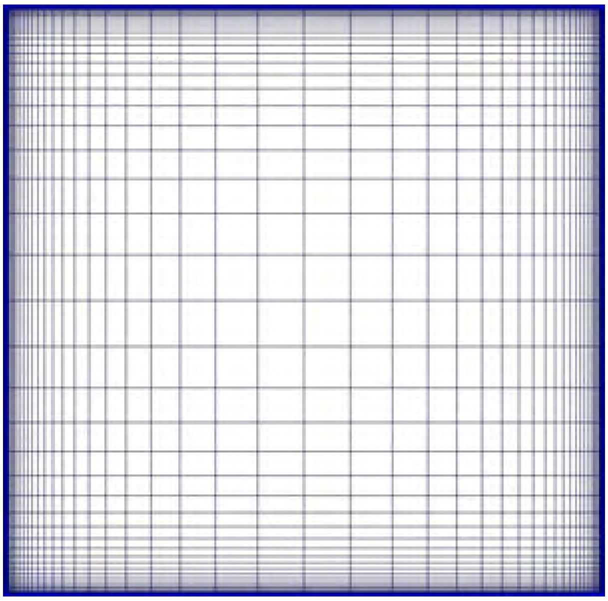

FIGURE 5.8: 81x81 node structured grid.

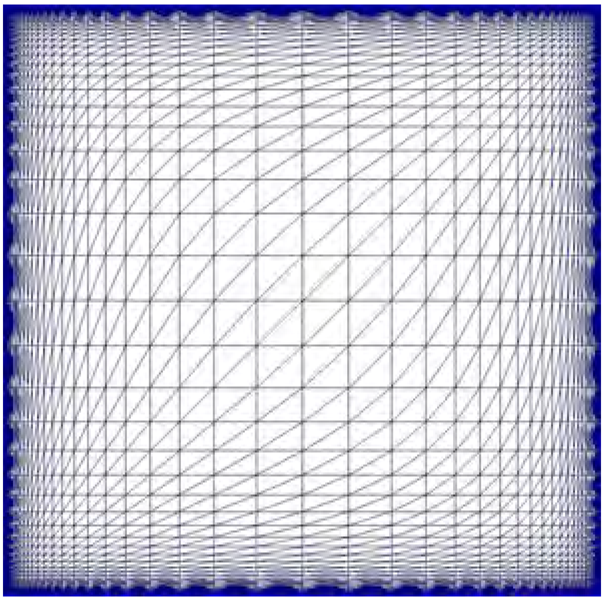

FIGURE 5.9: 81x81 node unstructured grid.

TABLE 5.3: 2D Lid-Driven Cavity error comparison summary on non-equispaced structured grids with a constant growth factor of 1.2.

Re. No.	**Error**	**Scheme:**				
		ET (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
1000	Max. abs.	58.72	**58.59**	98.60	100.00	93.99
	L1-norm	**55.80**	56.06	95.46	100.00	82.80
	L2-norm	**55.73**	55.97	96.09	100.00	84.09
5000	Max. abs.	64.64	**63.49**	100.80	100.00	98.02
	L1-norm	54.92	**53.35**	84.76	100.00	89.15
	L2-norm	55.35	**54.04**	93.31	100.00	92.78
10000	Max. abs.	73.27	**72.88**	101.16	100.00	100.22
	L1-norm	69.29	**64.88**	96.77	100.00	100.04
	L2-norm	71.02	**67.23**	98.17	100.00	99.38
Averaged	Max. abs.	65.54	**64.99**	100.19	100.00	97.41
Indices	L1-norm	60.00	**58.10**	92.33	100.00	90.66
(Base QUICK)	L2-norm	60.70	**59.08**	95.86	100.00	92.08

TABLE 5.4: 2D Lid-Driven Cavity error comparison summary on non-equispaced unstructured grids with a constant growth factor of 1.2.

Re. No.	Error	Scheme:				
		ET	**ET**	**CUI**	**QUICK**	**NVSF**
		(Adv. & Dif.)	(Adv.)	$\left(\kappa = \frac{1}{3}\right)$	$\left(\kappa = \frac{1}{2}\right)$	$\left(\kappa = \frac{3g-1}{2g}\right)$
1000	Max. abs.	57.30	**57.11**	107.05	100.00	92.62
	L1-norm	**53.90**	54.43	103.30	100.00	83.48
	L2-norm	**51.17**	51.41	106.76	100.00	88.33
5000	Max. abs.	41.60	**40.58**	110.87	100.00	101.69
	L1-norm	47.42	**45.15**	115.13	100.00	101.53
	L2-norm	41.05	**39.46**	111.83	100.00	98.30
10000	Max. abs.	**48.81**	48.88	110.96	100.00	105.22
	L1-norm	**46.14**	46.24	112.72	100.00	99.14
	L2-norm	**44.68**	44.71	111.48	100.00	99.41
Averaged	Max. abs.	49.23	**48.86**	109.63	100.00	99.84
Indices	L1-norm	49.15	**48.60**	110.38	100.00	94.72
(Base QUICK)	L2-norm	45.63	**45.19**	110.02	100.00	95.35

The accuracies of the ET schemes relative to the QUICK benchmark improved from the structured mesh to the unstructured mesh. Unlike the structured grids, the accuracies of the ET schemes did not show clear correlation to the Reynolds number. This is probably due to the varying flow present in this problem.

TABLE 5.5: 2D Lid-Driven Cavity error comparison combined summary of both non-equispaced structured and unstructured grids with a constant growth factor of 1.2.

Mesh Type	Error	Scheme:				
		ET	**ET**	**CUI**	**QUICK**	**NVSF**
		(Adv. & Dif.)	(Adv.)	$\left(\kappa = \frac{1}{3}\right)$	$\left(\kappa = \frac{1}{2}\right)$	$\left(\kappa = \frac{3g-1}{2g}\right)$
Struc.	Max. abs.	65.54	**64.99**	100.19	100.00	97.41
	L1-norm	60.00	**58.10**	92.33	100.00	90.66
	L2-norm	60.70	**59.08**	95.86	100.00	92.08
Unstruc.	Max. abs.	49.23	**48.86**	109.63	100.00	99.84
	L1-norm	49.15	**48.60**	110.38	100.00	94.72
	L2-norm	45.63	**45.19**	110.02	100.00	95.35
Averaged	Max. abs.	57.39	**56.92**	104.91	100.00	98.63
Indices	L1-norm	54.58	**53.35**	101.36	100.00	92.69
(Base QUICK)	L2-norm	53.17	**52.14**	102.94	100.00	93.71

Overall, the ET schemes out-performed all the other tested schemes and, on average, decreased error in excess of 45%, 42%, and 40% relative to CUI, QUICK, and NVSF respectively. It is interesting to note that the ET schemes achieve very similar errors in this 2D test case to one another, but in the 1D test case presented above, significant differences are observable. This is perhaps partly due to only one term being altered in the UP-AC algorithm as well as the cavity problem containing a range of flow features. NVSF is marginally superior to both QUICK and CUI on both structured and unstructured grids. CUI is superior to QUICK on structured grids, but the reverse is true for unstructured grids. On average, QUICK is marginally superior to CUI.

Chapter 6

Conclusion

A critical analysis of current advection discretization methodologies, namely the κ-Upwind schemes, was carried out in this work. This analysis revealed certain inadequecies and areas for improvement. All of the schemes, CUI and QUICK included (considered to be the best advection discretization options in terms of pure accuracy), were shown to reduce to 1st-order nodal accuracy as soon as generic mesh stretching was considered. Additionally, all schemes were shown to decrease in total accuracy, regardless of mesh stretching present, as soon as total advection-diffusion accuracy was considered. The NVSF scheme, which was originally expressed in the NVD format, was converted to the κ-Upwind format in this work and shown to be additionally applicable to unstructured grids through the use of the stretching factor developed. It was also shown to be equivalent to QUICK on equispaced grids. This scheme was the only currently available scheme found to adapt and optimise accuracy on non-equispaced grids. As such, it was found to maintain its internal face advection accuracy on generic non-equispaced grids. It too, however, experienced reductions in nodal advection and total advection-diffusion accuracy as soon as stretching was applied.

The main focus of this work, however, was the design and testing of a novel advection discretization methodology for systems containing both advection and diffusion flow. In keeping with the stated philosophy of industrial relevance, the developed discretization methodology was designed for application on both structured and unstructured anisotropic meshes. Two related schemes, the ET Advection-Diffusion scheme and the ET Advection scheme were developed to achieve the stated aims. Through analytical error analyses, both schemes were shown to improve the OoA at a face in all mesh conditions. Further, the ET Advection scheme was shown to be formally 2nd-order-accurate at an internal node. It did however suffer from non-physical oscillations as a function of mesh stretching, as well as advection-diffusion ratio. To rectify this, a blending with NVSF was proposed which was based on trial and error with the advection-diffusion equation. The developed ET schemes were next evaluated via application to 1D and 2D test cases. These involved a range of mesh stretching and advection-diffusion ratios. The 1D numerical example showed the ET Advection-Diffusion scheme to be superior to all other schemes, whereas the ET Advection scheme was superior only to CUI (and marginally superior to QUICK). The 2D numerical example showed both schemes to be equivalent and superior to the other schemes, but it is suspected that optimisation of multi-dimensional application would result in improvements for the ET Advection-Diffusion scheme. Average reductions in error up to 50% for the ET schemes relative to the base case of QUICK were found and, in certain individual test cases, this reduction in error was found to be upwards of 80%. These are clearly significant improvements in accuracy, and it is safe to say that an 'improved' and promising advection discretization methodology has been developed.

6.1 Areas for Future Research

A more comprehensive treatment of the incompressible flow solver discretization is proposed. Also, in developing the ET schemes, application was limited to incompressible flow regimes. Extension to compressible flow, however, would result in a greater range of industrial application and is therefore an area of potential future research.

Optimising the application, usage, and blending of the various ET schemes is an area of future potential. Improved blending and switching between the two ET schemes, and/or κ-Upwind schemes in various flow and mesh conditions could potentially result in further accuracy gains.

Appendix A

NVSF: Conversion to κ-Upwind Format

Darwish and Moukalled [5][4] designed the NVSF Scheme to maximise advection face accuracy of the NVD on non-equispaced grids. With reference to Figure 2.2, this was achieved through the use of a linear scheme with an NVD slope defined as follows:

$$Slope = \frac{\widetilde{x}_f(\widetilde{x}_f - 1)}{\widetilde{x}_u(\widetilde{x}_u - 1)} \tag{A.1}$$

where, in the context of Figure 2.2, $\widetilde{x}$ is defined as:

- $\widetilde{x} = \dfrac{x - x_{uu}}{x_c - x_{uu}}$

Therefore, $\widetilde{x}_f$ and $\widetilde{x}_u$ can be expressed as:

$$\widetilde{x}_f = \frac{x_f - x_{uu}}{x_c - x_{uu}} \tag{A.2}$$

and

$$\widetilde{x}_u = \frac{x_u - x_{uu}}{x_c - x_{uu}} \tag{A.3}$$

Expressed in the format of the stretching factor (f - defined in Chapter 2) and maintaining the assumption of vertex-centered grids (used throughout this work), Equations (A.2) and (A.3) can be shown to be equivalent to:

$$\widetilde{x}_f = \frac{x_f - x_{uu}}{x_c - x_{uu}} = \frac{1 + f}{2} \tag{A.4}$$

and

$$\widetilde{x}_u = \frac{x_u - x_{uu}}{x_c - x_{uu}} = f \tag{A.5}$$

Substitution of Equations (A.4) and (A.5) into Equation (A.1) results in the NVSF's NVD slope expressed in the stretching factor format of this work:

$$Slope = \frac{\widetilde{x}_f(\widetilde{x}_f - 1)}{\widetilde{x}_u(\widetilde{x}_u - 1)} = \frac{1+f}{4f} \tag{A.6}$$

While this form is already far simpler than originally expressed, it is not yet suitably written to aid implementation and comparison with κ-Upwind Schemes. Ideally, the NVSF expressed in κ-Upwind format as a function of stretching factor (f) would be the simplest of forms. With this goal in mind, we look to the work of Leonard [16]. In this work he stated that the slope of any κ-Upwind Scheme in the NVD is equivalent to:

$$Slope = \frac{2-\kappa}{2} \tag{A.7}$$

Equating Equations (A.6) and (A.7) to one another, and re-arranging to isolate the κ-value as a function of mesh stretching, results in the following κ-value:

$$\begin{aligned} \frac{2-\kappa}{2} &= \frac{1+f}{4f} \\ \kappa &= \frac{3f-1}{2f} \end{aligned} \tag{A.8}$$

Equation (A.8) represents the NVSF Scheme in the κ-Upwind format, expressed as a function of the mesh stretching factor, which was defined in this work.

Appendix B

Boundary-Adjacent-Face Analysis

An assessment of prominent κ-Upwind Schemes and the ET Schemes are shown here at boundary-adjacent faces for completeness. Where a scheme achieved an OoA greater than that of all other schemes, the result was made bold.

B.1 Current Methods

Table B.1: OoA's of various advection discretization schemes at upwind-boundary-adjacent faces on equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
CDS ($\kappa = 1$)	Up.-Boun.-Adj. Face	2nd^1	2nd^2
CUI $\left(\kappa = \frac{1}{3}\right)$	Up.-Boun.-Adj. Face	2nd^1	2nd^2
Fromm's ($\kappa = 0$)	Up.-Boun.-Adj. Face	2nd^1	2nd^2
LUI ($\kappa = -1$)	Up.-Boun.-Adj. Face	2nd^1	2nd^2
NVSF $\left(\kappa_L = \frac{1}{2}\right)$	Up.-Boun.-Adj. Face	2nd^1	2nd^2
QUICK $\left(\kappa = \frac{1}{2}\right)$	Up.-Boun.-Adj. Face	2nd^1	2nd^2

Superscript values correspond to error terms in Appendix C.

Table B.2: OoA's of various advection discretization schemes at upwind-boundary-adjacent faces on non-equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
CDS $(\kappa = 1)$	Up.-Boun.-Adj. Face	2nd[1]	2nd[2]
CUI $\left(\kappa = \frac{1}{3}\right)$	Up.-Boun.-Adj. Face	2nd[1]	2nd[2]
Fromm's $(\kappa = 0)$	Up.-Boun.-Adj. Face	2nd[1]	2nd[2]
LUI $(\kappa = -1)$	Up.-Boun.-Adj. Face	2nd[1]	2nd[2]
NVSF $\left(\kappa_L = \frac{3f-1}{2f}\right)$	Up.-Boun.-Adj. Face	2nd[1]	2nd[2]
QUICK $\left(\kappa = \frac{1}{2}\right)$	Up.-Boun.-Adj. Face	2nd[1]	2nd[2]

Superscript values correspond to error terms in Appendix C.

B.2 ET Schemes

Table B.3: OoA's of the ET Schemes at boundary-adjacent faces on equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
Adv.-Dif. ET	Up.-Boun.-Adj. Face	**3rd**[52]	2nd[18]
	Down.-Boun.-Adj. Face	**3rd**[17]	2nd[18]
Adv. ET	Up.-Boun.-Adj. Face	**3rd**[52]	2nd[18]
	Down.-Boun.-Adj. Face	**3rd**[17]	2nd[18]

Superscript values correspond to error terms in Appendix C.

Table B.4: OoA's of the ET Schemes at boundary-adjacent faces on non-equispaced grids.

Scheme	**Mesh Location**	**OoA of:**	
		Adv.	**Adv.-Dif.**
Adv.-Dif. ET	Up.-Boun.-Adj. Face	**3rd**[59]	2nd[18]
	Down.-Boun.-Adj. Face	**3rd**[60]	2nd[18]
Adv. ET	Up.-Boun.-Adj. Face	**3rd**[59]	2nd[18]
	Down.-Boun.-Adj. Face	**3rd**[60]	2nd[18]

Superscript values correspond to error terms in Appendix C.

Appendix C

Leading Error Terms

1. $-\left(\frac{1}{8}\right)\frac{d^2\phi}{dx^2}\bigg|_f \Delta x^2$
2. $\left[-\left(\frac{1}{8}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_f + \left(\frac{1}{24}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_f\right]\Delta x^2$
3. $-\left(\frac{1}{6}\right)u_x\,\frac{d^3\phi}{dx^3}\bigg|_c \Delta x^2$
4. $\left[-\left(\frac{1}{6}\right)u_x\,\frac{d^3\phi}{dx^3}\bigg|_c + \left(\frac{1}{12}\right)k\,\frac{d^4\phi}{dx^4}\bigg|_c\right]\Delta x^2$
5. $\left(\frac{1}{24}\right)\frac{d^2\phi}{dx^2}\bigg|_f \Delta x^2$
6. $\left[\left(\frac{1}{24}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_f + \left(\frac{1}{24}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_f\right]\Delta x^2$
7. $-\left(\frac{1}{12}\right)\frac{d^4\phi}{dx^4}\bigg|_c \Delta x^3$
8. $\left(\frac{1}{12}\right)k\,\frac{d^4\phi}{dx^4}\bigg|_c \Delta x^2$
9. $\left(\frac{1}{8}\right)\frac{d^2\phi}{dx^2}\bigg|_f \Delta x^2$
10. $\left[\left(\frac{1}{8}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_f + \left(\frac{1}{24}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_f\right]\Delta x^2$
11. $\left(\frac{1}{12}\right)u_x\,\frac{d^3\phi}{dx^3}\bigg|_c \Delta x^2$

12. $\left[\left(\frac{1}{12}\right) u_x \left.\frac{d^3\phi}{dx^3}\right|_c + \left(\frac{1}{12}\right) k \left.\frac{d^4\phi}{dx^4}\right|_c\right] \Delta x^2$

13. $\left(\frac{3}{8}\right) \left.\frac{d^2\phi}{dx^2}\right|_f \Delta x^2$

14. $\left[\left(\frac{3}{8}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_f + \left(\frac{1}{24}\right) k \left.\frac{d^3\phi}{dx^3}\right|_f\right] \Delta x^2$

15. $\left(\frac{1}{3}\right) u_x \left.\frac{d^3\phi}{dx^3}\right|_c \Delta x^2$

16. $\left[\left(\frac{1}{3}\right) u_x \left.\frac{d^3\phi}{dx^3}\right|_c + \left(\frac{1}{12}\right) k \left.\frac{d^4\phi}{dx^4}\right|_c\right] \Delta x^2$

17. $-\left(\frac{1}{16}\right) \left.\frac{d^3\phi}{dx^3}\right|_f \Delta x^3$

18. $\left(\frac{1}{24}\right) k \left.\frac{d^3\phi}{dx^3}\right|_f \Delta x^2$

19. $-\left(\frac{1}{24}\right) u_x \left.\frac{d^3\phi}{dx^3}\right|_c \Delta x^2$

20. $\left[-\left(\frac{1}{24}\right) u_x \left.\frac{d^3\phi}{dx^3}\right|_c + \left(\frac{1}{12}\right) k \left.\frac{d^4\phi}{dx^4}\right|_c\right] \Delta x^2$

21. $-\left(\frac{1-2g}{2g}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

22. $\left[-\left(\frac{1-2g}{2g}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c + \left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c\right] \Delta x$

23. $-\left(\frac{\left(\frac{7}{3}\right) f - 1}{8(f-1)}\right) \left.\frac{d^2\phi}{dx^2}\right|_f \Delta x^2$

24. $\left[-\left(\frac{\left(\frac{7}{3}\right) f - 1}{8(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_f + \left(\frac{1}{24}\right) k \left.\frac{d^3\phi}{dx^3}\right|_f\right] \Delta x^2$

25. $-\left(\frac{-8fg+3f-2g^2+8g-3}{6g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

26. $\left[-\left(\frac{-8fg+3f-2g^2+8g-3}{6g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c + \left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c\right] \Delta x$

27. $\left(-\frac{3f-1}{8(f-1)}\right) \left.\frac{d^2\phi}{dx^2}\right|_f \Delta x^2$

28. $\left[-\left(\frac{3f-1}{8(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_f + \left(\frac{1}{24}\right) k \left.\frac{d^3\phi}{dx^3}\right|_f\right] \Delta x^2$

29. $-\left(\frac{-3fg+f-g^2+3g-1}{2g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

30. $\left[-\left(\frac{-3fg+f-g^2+3g-1}{2g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c + \left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c\right] \Delta x$

31. $-\left(\frac{5f-1}{8(f-1)}\right) \left.\frac{d^2\phi}{dx^2}\right|_f \Delta x^2$

32. $\left[-\left(\frac{5f-1}{8(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_f + \left(\frac{1}{24}\right) k \left.\frac{d^3\phi}{dx^3}\right|_f\right] \Delta x^2$

33. $-\left(\frac{-4fg+f-2g^2+4g-1}{2g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

34. $\left[-\left(\frac{-4fg+f-2g^2+4g-1}{2g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c + \left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c\right] \Delta x$

35. $-\left(\frac{f+1}{48\,(1-f)}\right) \left.\frac{d^3\phi}{dx^3}\right|_f \Delta x^3$

36. $-\left(\frac{1-2g}{4g}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

37. $-\left[\left(\frac{1-2g}{4g}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c + \left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c\right] \Delta x$

38. $-\left(\frac{2f-1}{8(f-1)}\right) \left.\frac{d^2\phi}{dx^2}\right|_f \Delta x^2$

39. $\left[-\left(\frac{2f-1}{8(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_f + \left(\frac{1}{24}\right) k \left.\frac{d^3\phi}{dx^3}\right|_f\right] \Delta x^2$

40. $-\left(\frac{-5fg+2f-g^2+5g-2}{4g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

41. $\left[-\left(\frac{-5fg+2f-g^2+5g-2}{4g(f-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c + \left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c\right] \Delta x$

42. $-\left(\frac{-10g^2+11g-3}{6g(g-1)}\right) u_x \left.\frac{d^2\phi}{dx^2}\right|_c \Delta x$

43. $$\left[-\left(\frac{-10g^2+11g-3}{6g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c+\left(\frac{2g-1}{3g}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_c\right]\Delta x$$

44. $$-\left(\frac{-4g^2+4g-1}{2g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c\,\Delta x$$

45. $$\left[-\left(\frac{-4g^2+4g-1}{2g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c+\left(\frac{2g-1}{3g}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_c\right]\Delta x$$

46. $$-\left(\frac{-6g^2+5g-1}{2g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c\,\Delta x$$

47. $$\left[-\left(\frac{-6g^2+5g-1}{2g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c+\left(\frac{2g-1}{3g}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_c\right]\Delta x$$

48. $$-\left(\frac{-6g^2+7g-2}{4g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c\,\Delta x$$

49. $$\left[-\left(\frac{-6g^2+7g-2}{4g(g-1)}\right)u_x\,\frac{d^2\phi}{dx^2}\bigg|_c+\left(\frac{2g-1}{3g}\right)k\,\frac{d^3\phi}{dx^3}\bigg|_c\right]\Delta x$$

50. $$\left[-\left(\frac{k}{192u_x}\right)\frac{d^5\phi}{dx^5}\bigg|_f+\left(\frac{3}{128}\right)\frac{d^4\phi}{dx^4}\bigg|_f\right]\Delta x^4$$

51. $$\left[-\left(\frac{3}{640}\right)k\,\frac{d^5\phi}{dx^5}\bigg|_f+\left(\frac{3}{128}\right)u_x\,\frac{d^4\phi}{dx^4}\bigg|_f\right]\Delta x^4$$

52. $$\left(\frac{1}{16}\right)\frac{d^3\phi}{dx^3}\bigg|_f\,\Delta x^3$$

53. $$-\left(\frac{1}{24}\right)\left[\frac{d^3\phi}{dx^3}\bigg|_c+\left(\frac{k}{u_x}\right)\frac{d^4\phi}{dx^4}\bigg|_c\right]\Delta x^2$$

54. $$-\left(\frac{1}{24}\right)\left[u_x\,\frac{d^3\phi}{dx^3}\bigg|_c-k\,\frac{d^4\phi}{dx^4}\bigg|_c\right]\Delta x^2$$

55. $$\left(\frac{3}{128}\right)\frac{d^4\phi}{dx^4}\bigg|_f\,\Delta x^4$$

56. $$-\left(\frac{1}{24}\right)\frac{d^3\phi}{dx^3}\bigg|_c\,\Delta x^2$$

57. $$-\left(\frac{f+g-1}{96g(f-1)}\right)\left(\frac{k}{u_x}\right)\frac{d^4\phi}{dx^4}\bigg|_f\,\Delta x^3$$

58. $$-\left(\frac{f+g-1}{96g(f-1)}\right)k\,\frac{d^4\phi}{dx^4}\bigg|_f\,\Delta x^3$$

59. $$-\left(\frac{g-2}{48g}\right)\frac{d^3\phi}{dx^3}\bigg|_f\,\Delta x^3$$

60. $-\left(\frac{f+1}{48(1-f)}\right)\left.\frac{d^3\phi}{dx^3}\right|_f \Delta x^3$

61. $-\left(\frac{g(1-2g)}{12(g-1)}\right)\left(\frac{k}{u_x}\right)\left.\frac{d^3\phi}{dx^3}\right|_c \Delta x$

62. $-\left(\frac{(1-2g)(g-2)^2}{12g(g-1)}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c \Delta x$

63. $-\left(\frac{(f+1)(2-g)}{384g(f-1)}\right)\left.\frac{d^4\phi}{dx^4}\right|_f \Delta x^4$

64. $-\left(\frac{g}{12}\right)\left.\frac{d^3\phi}{dx^3}\right|_c \Delta x^2$

65. $-\left(\frac{2g-1}{3g}\right) k \left.\frac{d^3\phi}{dx^3}\right|_c \Delta x$

66. $-\left(\frac{2g-1}{96g(g-1)}\right)\left(\frac{k}{u_x}\right)\left.\frac{d^4\phi}{dx^4}\right|_f \Delta x^3$

67. $-\left(\frac{2g-1}{96g(g-1)}\right) k \left.\frac{d^4\phi}{dx^4}\right|_f \Delta x^3$

68. $-\left(\frac{(g+1)(2-g)}{384g(g-1)}\right)\left.\frac{d^4\phi}{dx^4}\right|_f \Delta x^4$

Appendix D

Advection-Diffusion ET Scheme: Diffusion Decoupling

The optimum values from section 3.1.1 are substituted into Equation (4.2) in order to analyse the diffusion decoupling as a result of the Advection-Diffusion ET Scheme. Only terms in the coefficients containing the diffusion constant (k) are analysed, however, as these are the diffusion components. In addition, only the coefficients of ϕ_u, ϕ_c, and ϕ_d are shown below, as these are the only coefficients that vary from the advection coefficients shown in Section 4.2.1.

Coefficient of ϕ_u:

$$\frac{(-4g^2+5g-5)k}{4(g^2-g+1)\Delta x} \tag{D.1}$$

Coefficient of ϕ_c:

$$\frac{(g-2)(g+1)k}{2(g-1)\Delta x} \tag{D.2}$$

Coefficient of ϕ_d:

$$\frac{g(4g^2-3g+4)k}{4(g-1)(g^2-g+1)\Delta x} \tag{D.3}$$

All of the above coefficients are non-zero and defined for all realistic values of g. This indicates that odd-even decoupling will never be a problem for the diffusive component of the Advection-Diffusion ET Scheme.

Appendix E

1D Advection-Diffusion Test Case Data

Data from the 1D Advection-Diffusion test case is presented here for completeness.

TABLE E.1: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 10 and a constant mesh growth factor of 1.1.

Mesh Size	**Error**	**Scheme:**				
		ET	**ET**	**CUI**	**QUICK**	**NVSF**
		(Adv. & Dif.)	(Adv.)	$\left(\kappa = \frac{1}{3}\right)$	$\left(\kappa = \frac{1}{2}\right)$	$\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	1.14E-03	3.37E-03	8.42E-03	2.46E-03	**7.94E-04**
	L1-norm	3.74E-04	1.07E-03	2.82E-03	6.98E-04	**1.76E-04**
	L2-norm	1.75E-04	5.03E-04	1.37E-03	3.64E-04	**9.71E-05**
30 nodes	Max. abs.	**1.33E-04**	2.71E-04	6.37E-04	3.18E-04	2.31E-04
	L1-norm	**6.29E-05**	1.28E-04	3.00E-04	1.49E-04	1.07E-04
	L2-norm	**1.43E-05**	2.90E-05	6.81E-05	3.39E-05	2.46E-05
90 nodes	Max. abs.	**6.87E-05**	1.40E-04	3.29E-04	1.66E-04	1.22E-04
	L1-norm	**1.18E-05**	2.39E-05	5.68E-05	2.89E-05	2.12E-05
	L2-norm	**2.51E-06**	5.10E-06	1.20E-05	6.09E-06	4.46E-06
Averaged	Max. abs.	**43.17**	101.99	246.77	100.00	59.42
Indices	L1-norm	**45.58**	107.31	267.59	100.00	57.05
(Base QUICK)	L2-norm	**43.75**	102.48	258.18	100.00	57.50

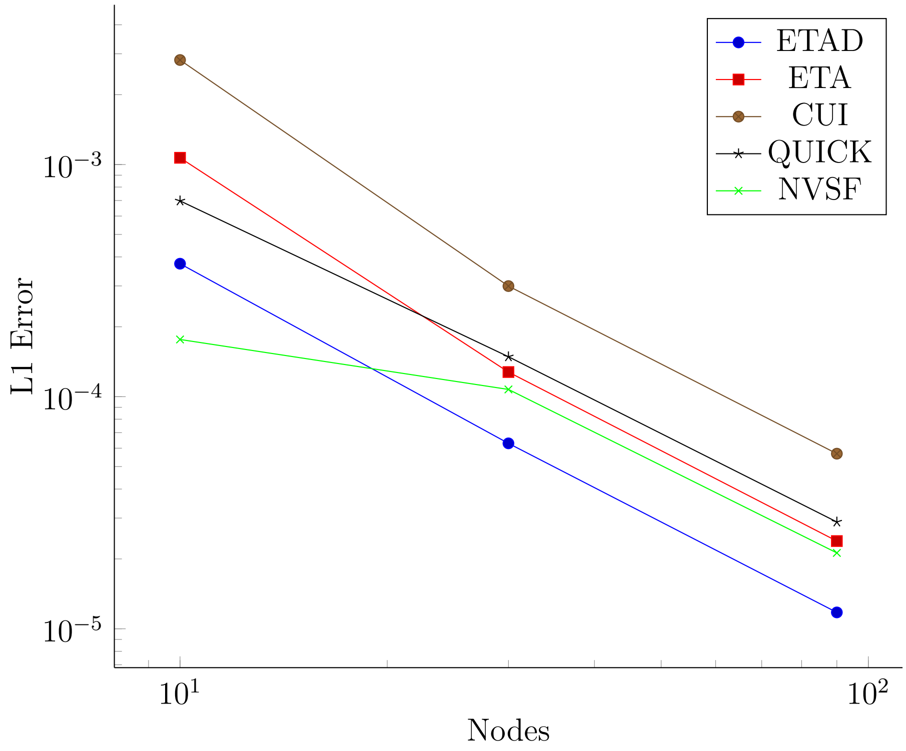

FIGURE E.1: L1 Error vs Nodes for an advection-diffusion ratio of 10 and a constant mesh growth factor of 1.1.

TABLE E.2: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 10 and a constant mesh growth factor of 1.2.

Mesh Size	**Error**	**Scheme:**				
		ET	**ET**	**CUI**	**QUICK**	**NVSF**
		(Adv. & Dif.)	(Adv.)	$\left(\kappa = \frac{1}{3}\right)$	$\left(\kappa = \frac{1}{2}\right)$	$\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	**1.01E-03**	4.03E-03	7.02E-03	3.13E-03	1.15E-03
	L1-norm	5.15E-04	1.69E-03	2.90E-03	1.16E-03	**3.65E-04**
	L2-norm	2.08E-04	7.14E-04	1.25E-03	5.28E-04	**1.76E-04**
30 nodes	Max. abs.	**6.78E-05**	5.64E-04	1.25E-03	6.38E-04	3.38E-04
	L1-norm	**1.84E-05**	1.48E-04	3.46E-04	1.76E-04	9.05E-05
	L2-norm	**5.12E-06**	4.48E-05	1.01E-04	5.10E-05	2.66E-05
90 nodes	Max. abs.	**6.31E-05**	5.33E-04	1.19E-03	6.06E-04	3.22E-04
	L1-norm	**6.22E-06**	4.70E-05	1.10E-04	5.61E-05	2.91E-05
	L2-norm	**1.60E-06**	1.41E-05	3.19E-05	1.62E-05	8.45E-06
Averaged	Max. abs.	**17.76**	101.71	205.95	100.00	47.67
Indices	L1-norm	**21.96**	104.49	214.21	100.00	44.93
(Base QUICK)	L2-norm	**19.79**	103.42	210.66	100.00	45.83

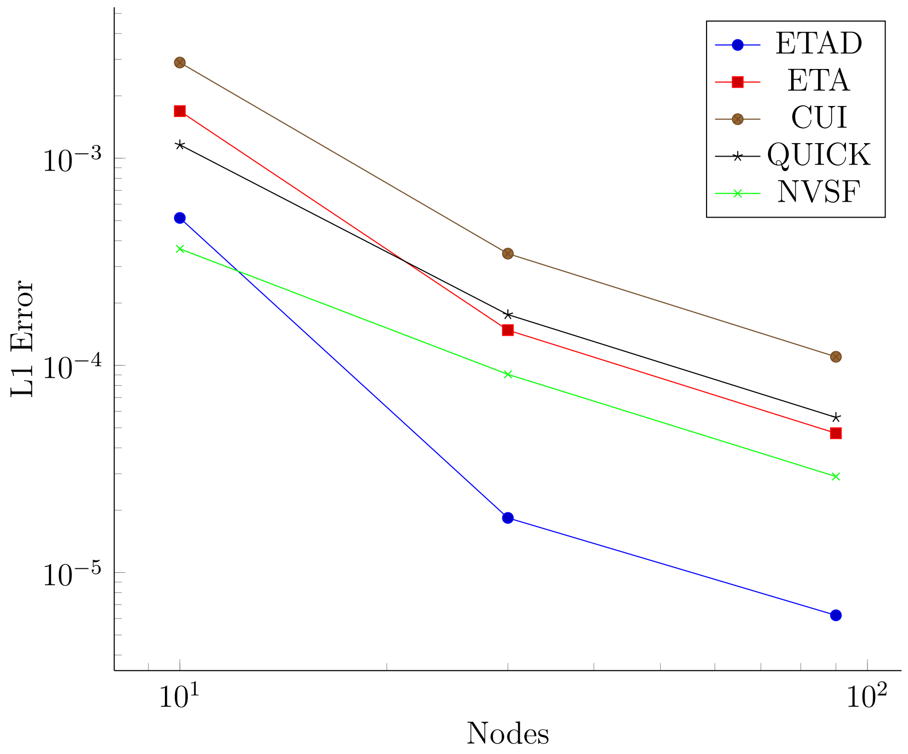

FIGURE E.2: L1 Error vs Nodes for an advection-diffusion ratio of 10 and a constant mesh growth factor of 1.2.

TABLE E.3: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 10 and a constant mesh growth factor of 1.3.

Mesh Size	Error	**Scheme:**				
		ET (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	**6.04E-04**	3.02E-03	6.14E-03	3.02E-03	9.70E-04
	L1-norm	**3.23E-04**	1.41E-03	2.77E-03	1.27E-03	3.92E-04
	L2-norm	**1.24E-04**	5.74E-04	1.14E-03	5.43E-04	1.63E-04
30 nodes	Max. abs.	**1.27E-04**	1.05E-03	2.46E-03	1.25E-03	4.80E-04
	L1-norm	**3.71E-05**	1.91E-04	4.83E-04	2.45E-04	9.20E-05
	L2-norm	**9.48E-06**	6.86E-05	1.66E-04	8.46E-05	3.12E-05
90 nodes	Max. abs.	**1.26E-04**	1.04E-03	2.45E-03	1.24E-03	4.78E-04
	L1-norm	**1.26E-05**	6.35E-05	1.60E-04	8.13E-05	3.06E-05
	L2-norm	**3.16E-06**	2.27E-05	5.51E-05	2.81E-05	1.03E-05
Averaged Indices (Base QUICK)	Max. abs.	**13.42**	89.21	198.97	100.00	36.31
	L1-norm	**18.73**	89.04	204.32	100.00	35.43
	L2-norm	**15.10**	89.25	200.83	100.00	34.60

TABLE E.4: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 20 and a constant mesh growth factor of 1.1.

Mesh Size	**Error**	**Scheme:**				
		ET (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	1.39E-02	1.36E-02	9.66E-03	**9.02E-03**	1.44E-02
	L1-norm	1.52E-03	1.84E-03	**1.42E-03**	1.60E-03	2.44E-03
	L2-norm	1.40E-03	1.40E-03	**1.04E-03**	1.04E-03	1.63E-03
30 nodes	Max. abs.	**2.79E-04**	4.64E-04	1.13E-03	5.53E-04	3.96E-04
	L1-norm	**1.05E-04**	1.76E-04	4.30E-04	2.05E-04	1.44E-04
	L2-norm	**2.72E-05**	4.52E-05	1.10E-04	5.33E-05	3.78E-05
90 nodes	Max. abs.	**8.23E-05**	1.36E-04	3.32E-04	1.65E-04	1.20E-04
	L1-norm	**1.42E-05**	2.38E-05	5.76E-05	2.84E-05	2.05E-05
	L2-norm	**3.01E-06**	5.02E-06	1.22E-05	6.04E-06	4.36E-06
Averaged	Max. abs.	**85.00**	105.91	170.89	100.00	101.35
Indices	L1-norm	**65.42**	94.73	166.88	100.00	98.30
(Base QUICK)	L2-norm	**78.50**	100.92	169.44	100.00	99.96

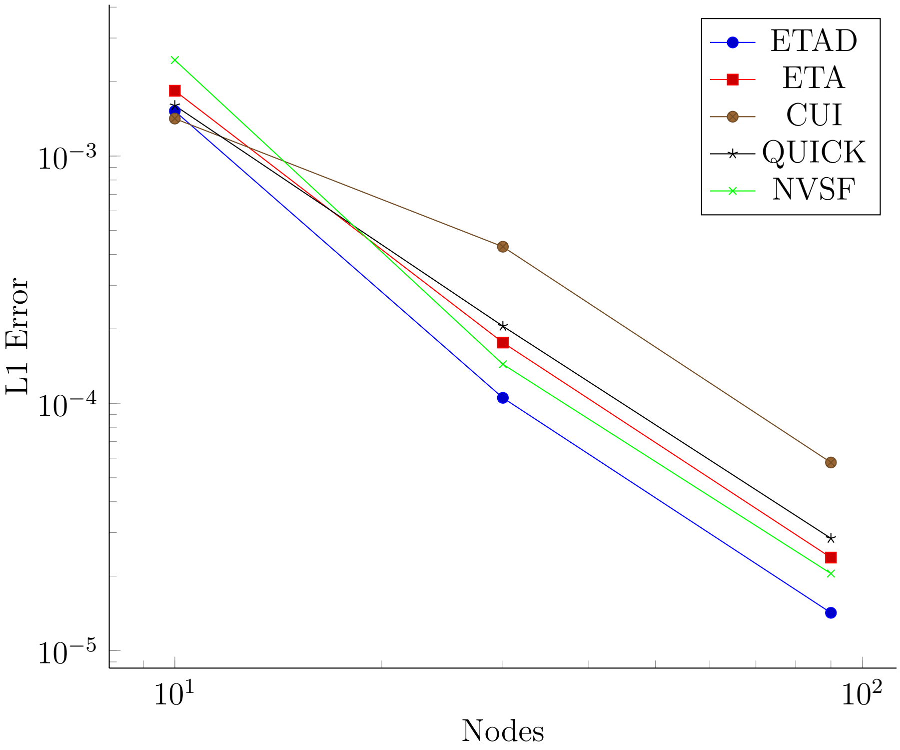

FIGURE E.3: L1 Error vs Nodes for an advection-diffusion ratio of 20 and a constant mesh growth factor of 1.1.

Table E.5: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 20 and a constant mesh growth factor of 1.2.

Mesh Size	**Error**	**Scheme:**				
		ET	**ET**	**CUI**	**QUICK**	**NVSF**
		(Adv. & Dif.)	(Adv.)	$\left(\kappa = \frac{1}{3}\right)$	$\left(\kappa = \frac{1}{2}\right)$	$\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	**2.15E-03**	7.71E-03	1.36E-02	3.30E-03	2.95E-03
	L1-norm	7.72E-04	1.82E-03	2.97E-03	**5.47E-04**	8.81E-04
	L2-norm	3.68E-04	9.86E-04	1.76E-03	**3.66E-04**	4.42E-04
30 nodes	Max. abs.	**1.65E-04**	5.62E-04	1.31E-03	6.45E-04	3.32E-04
	L1-norm	**4.90E-05**	1.59E-04	3.55E-04	1.73E-04	8.53E-05
	L2-norm	**1.37E-05**	4.58E-05	1.04E-04	5.13E-05	2.56E-05
90 nodes	Max. abs.	**1.49E-04**	5.07E-04	1.18E-03	5.83E-04	3.00E-04
	L1-norm	**1.47E-05**	4.81E-05	1.07E-04	5.22E-05	2.57E-05
	L2-norm	**4.12E-06**	1.38E-05	3.13E-05	1.54E-05	7.70E-06
Averaged	Max. abs.	**38.71**	135.83	272.08	100.00	64.11
Indices	L1-norm	**65.91**	172.10	317.71	100.00	86.57
(Base QUICK)	L2-norm	**51.29**	149.36	295.60	100.00	73.47

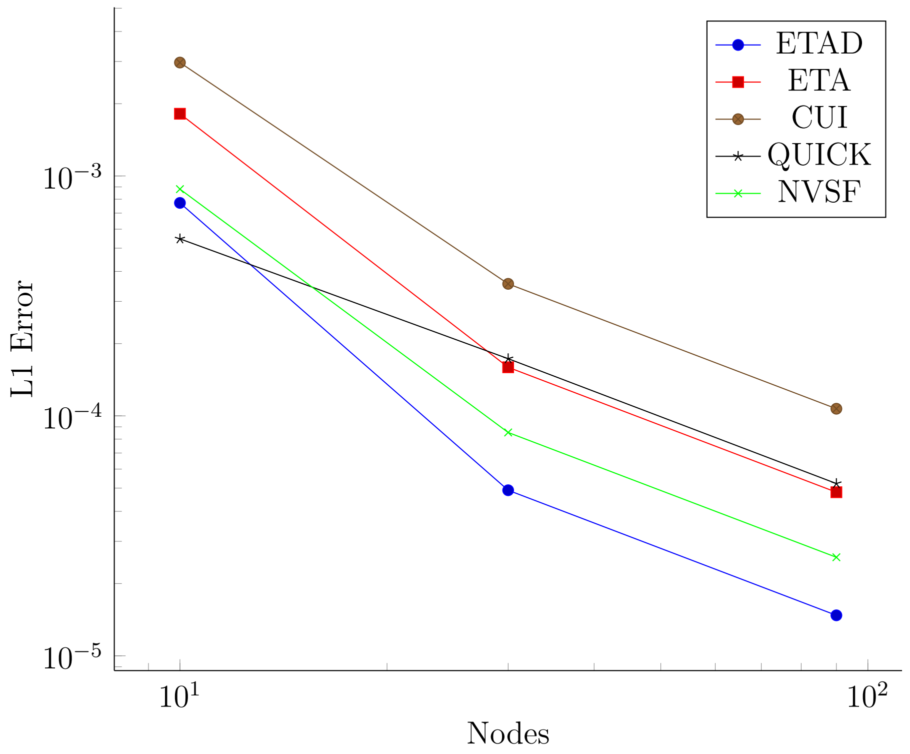

Figure E.4: L1 Error vs Nodes for an advection-diffusion ratio of 20 and a constant mesh growth factor of 1.2.

TABLE E.6: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 20 and a constant mesh growth factor of 1.3.

Mesh Size	**Error**	**Scheme:**				
		ET (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	1.68E-03	5.20E-03	1.12E-02	4.65E-03	**1.44E-03**
	L1-norm	6.79E-04	1.73E-03	3.32E-03	1.25E-03	**4.42E-04**
	L2-norm	2.96E-04	8.31E-04	1.70E-03	6.83E-04	**2.15E-04**
30 nodes	Max. abs.	**2.74E-04**	1.02E-03	2.41E-03	1.21E-03	4.27E-04
	L1-norm	**6.53E-05**	2.08E-04	4.56E-04	2.21E-04	8.27E-05
	L2-norm	**2.09E-05**	7.05E-05	1.60E-04	7.91E-05	2.75E-05
90 nodes	Max. abs.	**2.71E-04**	1.01E-03	2.39E-03	1.20E-03	4.23E-04
	L1-norm	**2.16E-05**	6.88E-05	1.50E-04	7.29E-05	2.73E-05
	L2-norm	**6.91E-06**	2.33E-05	5.28E-05	2.61E-05	9.07E-06
Averaged	Max. abs.	**27.15**	93.47	213.61	100.00	33.93
Indices	L1-norm	37.82	108.99	225.89	100.00	**36.75**
(Base QUICK)	L2-norm	**32.10**	100.02	217.79	100.00	33.67

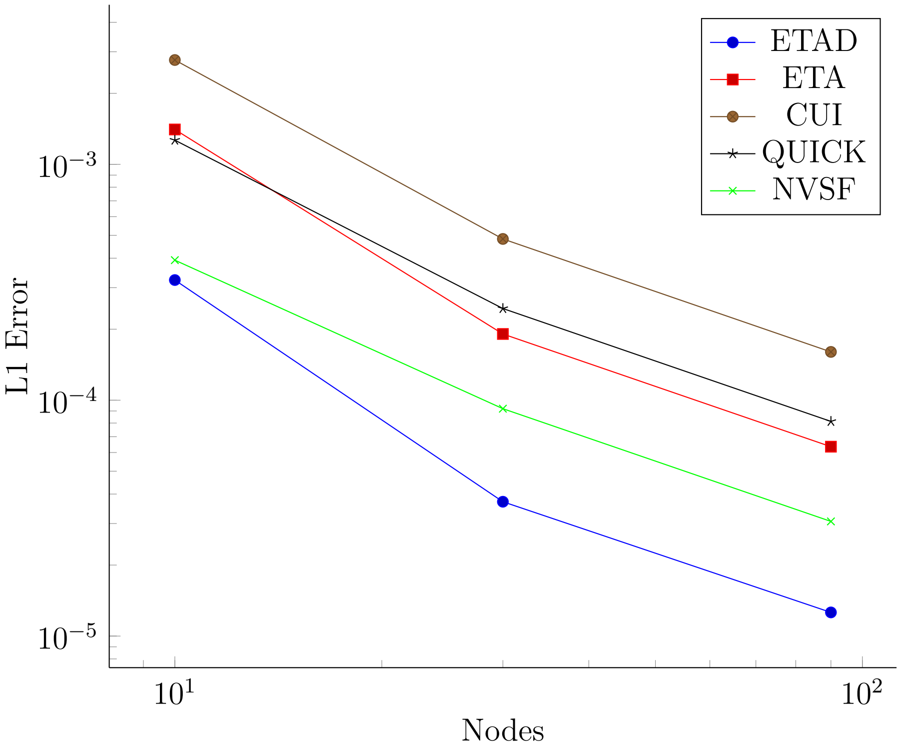

FIGURE E.5: L1 Error vs Nodes for an advection-diffusion ratio of 20 and a constant mesh growth factor of 1.3.

Table E.7: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 40 and a constant mesh growth factor of 1.1.

		Scheme:				
Mesh Size	**Error**	**ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	9.04E-02	9.03E-02	**4.06E-02**	7.95E-02	9.12E-02
	L1-norm	1.00E-02	1.02E-02	**4.29E-03**	8.03E-03	9.21E-03
	L2-norm	9.08E-03	9.08E-03	**4.06E-03**	7.95E-03	9.12E-03
30 nodes	Max. abs.	**7.54E-04**	8.91E-04	2.59E-03	1.20E-03	8.18E-04
	L1-norm	**1.77E-04**	2.12E-04	6.47E-04	2.87E-04	1.89E-04
	L2-norm	**5.78E-05**	6.86E-05	2.05E-04	9.27E-05	6.22E-05
90 nodes	Max. abs.	**1.07E-04**	1.26E-04	3.34E-04	1.66E-04	1.19E-04
	L1-norm	**1.85E-05**	2.17E-05	5.79E-05	2.86E-05	2.06E-05
	L2-norm	**3.93E-06**	4.60E-06	1.22E-05	6.06E-06	4.38E-06
Averaged	Max. abs.	**80.45**	87.87	156.19	100.00	85.06
Indices	L1-norm	**83.58**	92.26	160.37	100.00	84.21
(Base QUICK)	L2-norm	**80.43**	88.00	157.87	100.00	84.67

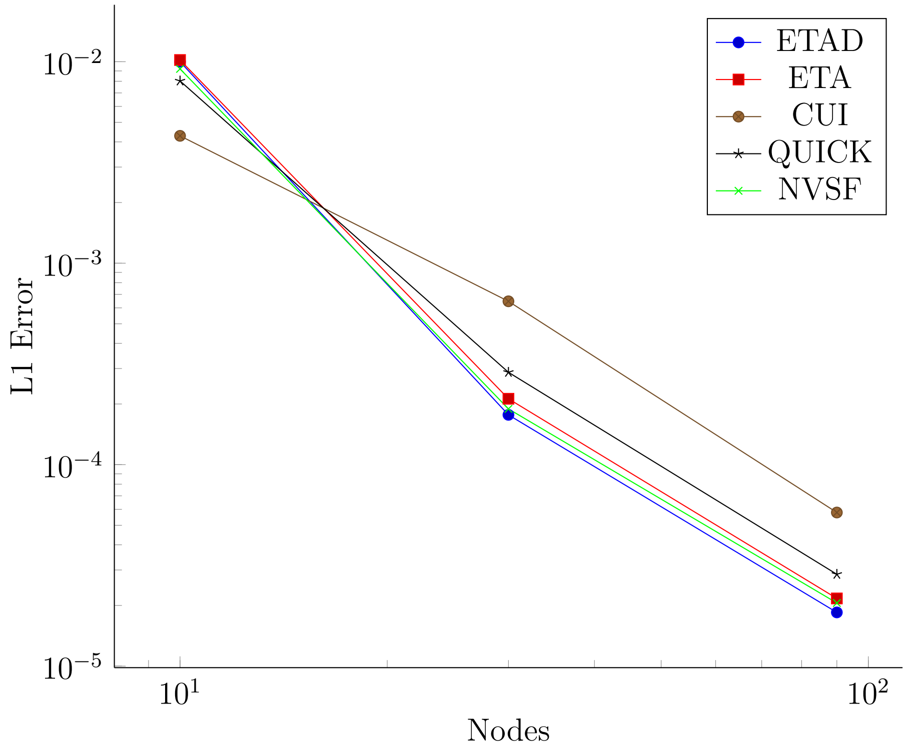

Figure E.6: L1 Error vs Nodes for an advection-diffusion ratio of 40 and a constant mesh growth factor of 1.1.

Table E.8: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 40 and a constant mesh growth factor of 1.2.

Mesh Size	**Error**	**Scheme:**				
		ET	**ET**	**CUI**	**QUICK**	**NVSF**
		(Adv. & Dif.)	(Adv.)	$\left(\kappa=\frac{1}{3}\right)$	$\left(\kappa=\frac{1}{2}\right)$	$\left(\kappa=\frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	3.63E-02	3.61E-02	**4.51E-03**	2.24E-02	3.70E-02
	L1-norm	3.76E-03	3.86E-03	**5.75E-04**	3.06E-03	4.81E-03
	L2-norm	3.63E-03	3.61E-03	**4.59E-04**	2.36E-03	3.84E-03
30 nodes	Max. abs.	**3.02E-04**	4.42E-04	1.45E-03	7.19E-04	3.66E-04
	L1-norm	**7.83E-05**	1.18E-04	3.90E-04	1.90E-04	9.34E-05
	L2-norm	**2.36E-05**	3.50E-05	1.15E-04	5.67E-05	2.82E-05
90 nodes	Max. abs.	**2.47E-04**	3.61E-04	1.18E-03	5.88E-04	2.99E-04
	L1-norm	**2.15E-05**	3.25E-05	1.07E-04	5.22E-05	2.58E-05
	L2-norm	**6.45E-06**	9.55E-06	3.13E-05	1.55E-05	7.70E-06
Averaged Indices (Base QUICK)	Max. abs.	**82.07**	94.65	140.78	100.00	89.06
	L1-norm	**68.43**	83.53	143.08	100.00	85.26
	L2-norm	**79.21**	92.30	141.53	100.00	87.50

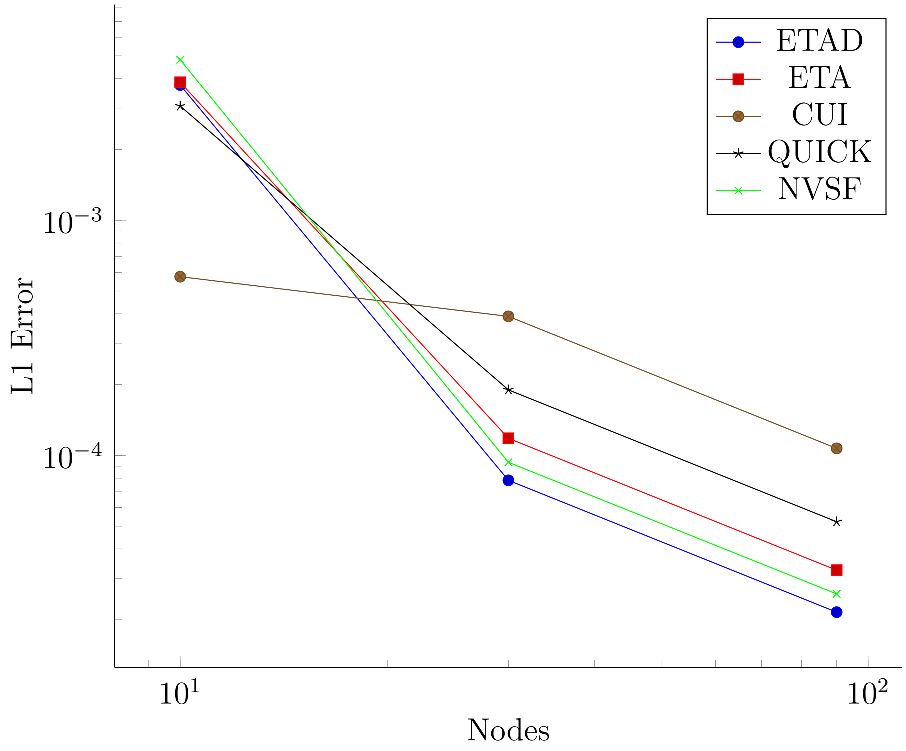

Figure E.7: L1 Error vs Nodes for an advection-diffusion ratio of 40 and a constant mesh growth factor of 1.2.

TABLE E.9: 1D Advection-Diffusion Equation error comparisons with an advection-diffusion ratio of 40 and a constant mesh growth factor of 1.3.

Mesh Size	**Error**	**Scheme:** **ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
10 nodes	Max. abs.	9.67E-03	**9.53E-03**	1.83E-02	2.05E-02	9.95E-03
	L1-norm	1.44E-03	**1.10E-03**	2.79E-03	5.93E-03	2.28E-03
	L2-norm	1.04E-03	**9.61E-04**	2.01E-03	3.21E-03	1.37E-03
30 nodes	Max. abs.	**3.60E-04**	5.84E-04	2.39E-03	1.22E-03	4.23E-04
	L1-norm	**6.55E-05**	1.08E-04	4.60E-04	2.24E-04	8.32E-05
	L2-norm	**2.34E-05**	3.82E-05	1.62E-04	7.99E-05	2.78E-05
90 nodes	Max. abs.	**3.53E-04**	5.72E-04	2.34E-03	1.20E-03	4.15E-04
	L1-norm	**2.14E-05**	3.51E-05	1.50E-04	7.31E-05	2.72E-05
	L2-norm	**7.64E-06**	1.25E-05	5.28E-05	2.61E-05	9.07E-06
Averaged	Max. abs.	**35.39**	47.33	160.06	100.00	39.27
Indices	L1-norm	**27.65**	38.27	152.91	100.00	37.62
(Base QUICK)	L2-norm	**30.34**	41.81	155.72	100.00	37.42

Appendix F

2D Lid-Driven Cavity Test Case Data

Data from the 2D Lid-Driven Cavity test case is presented here for completeness.

Table F.1: 2D Lid-Driven Cavity $Re = 1000$ error comparisons on non-equispaced structured grids with a constant growth factor of 1.2.

Mesh Size	**Error**	**Scheme:**				
		ET (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
∼ 10%	Max. abs.	1.46E-02	**1.46E-02**	2.42E-02	2.49E-02	2.30E-02
(81x81)	L1-norm	3.85E-05	**3.85E-05**	6.57E-05	6.88E-05	5.66E-05
(6561 nodes)	L2-norm	6.03E-06	**6.03E-06**	1.04E-05	1.08E-05	8.96E-06
∼ 15%	Max. abs.	1.19E-02	**1.18E-02**	2.42E-02	2.49E-02	2.30E-02
(95x95)	L1-norm	**2.19E-05**	2.20E-05	3.75E-05	3.94E-05	3.30E-05
(9025 nodes)	L2-norm	**3.42E-06**	3.42E-06	5.85E-06	6.10E-06	5.27E-06
∼ 20%	Max. abs.	1.46E-02	**1.45E-02**	2.11E-02	2.08E-02	2.03E-02
(111x111)	L1-norm	**2.09E-05**	2.12E-05	3.59E-05	3.76E-05	3.09E-05
(12321 nodes)	L2-norm	**3.22E-06**	3.26E-06	5.60E-06	5.82E-06	4.83E-06
Averaged	Max. abs.	58.72	**58.59**	98.60	100.00	93.99
Indices	L1-norm	**55.80**	56.06	95.46	100.00	82.80
(Base QUICK)	L2-norm	**55.73**	55.97	96.09	100.00	84.09

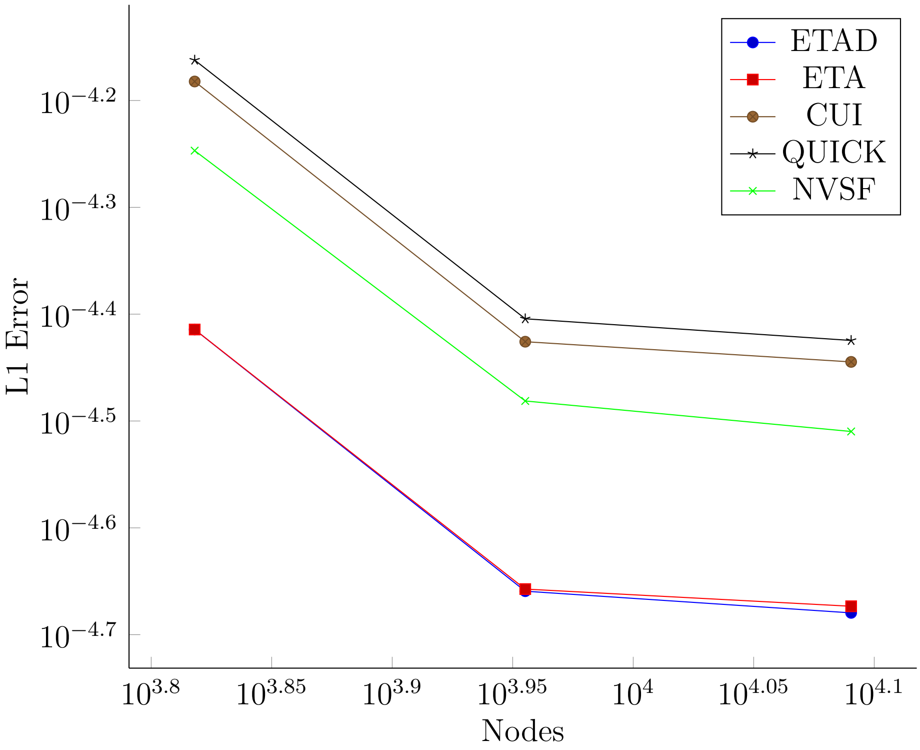

Figure F.1: L1 Error vs Nodes for $Re = 1000$ on structured grids with constant mesh growth factor of 1.2.

TABLE F.2: 2D Lid-Driven Cavity $Re = 5000$ error comparisons on non-equispaced structured grids with a constant growth factor of 1.2.

Mesh Size	**Error**	**Scheme:**				
		ET (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa=\frac{1}{3}\right)$	**QUICK** $\left(\kappa=\frac{1}{2}\right)$	**NVSF** $\left(\kappa=\frac{3g-1}{2g}\right)$
~ 10%	Max. abs.	**3.59E-02**	3.61E-02	5.89E-02	5.85E-02	5.74E-02
(81x81)	L1-norm	9.43E-05	**9.38E-05**	1.40E-04	1.71E-04	1.50E-04
(6561 nodes)	L2-norm	1.53E-05	**1.53E-05**	2.61E-05	2.84E-05	2.60E-05
~ 15%	Max. abs.	3.81E-02	**3.55E-02**	5.58E-02	5.49E-02	5.39E-02
(95x95)	L1-norm	**6.04E-05**	6.13E-05	1.04E-04	1.15E-04	1.06E-04
(9025 nodes)	L2-norm	1.07E-05	**1.03E-05**	1.82E-05	1.89E-05	1.80E-05
~ 20%	Max. abs.	**3.69E-02**	3.74E-02	5.85E-02	5.85E-02	5.71E-02
(111x111)	L1-norm	5.31E-05	**4.82E-05**	7.64E-05	9.27E-05	8.11E-05
(12321 nodes)	L2-norm	8.45E-06	**8.16E-06**	1.39E-05	1.52E-05	1.39E-05
Averaged	Max. abs.	64.64	**63.49**	100.80	100.00	98.02
Indices	L1-norm	54.92	**53.35**	84.76	100.00	89.15
(Base QUICK)	L2-norm	55.35	**54.04**	93.31	100.00	92.78

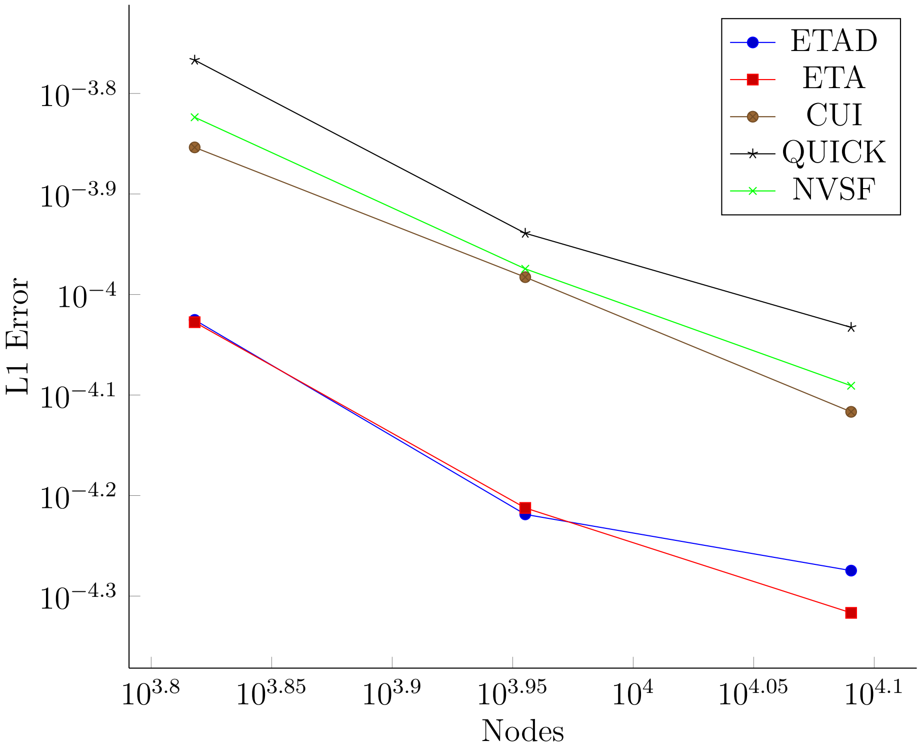

FIGURE F.2: L1 Error vs Nodes for $Re = 5000$ on structured grids with constant mesh growth factor of 1.2.

TABLE F.3: 2D Lid-Driven Cavity $Re = 10000$ error comparisons on non-equispaced structured grids with a constant growth factor of 1.2.

		Scheme:				
Mesh Size	**Error**	**ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
~ 10%	Max. abs.	**6.40E-02**	6.42E-02	9.03E-02	8.94E-02	8.96E-02
(81x81)	L1-norm	**2.19E-04**	2.20E-04	3.30E-04	3.40E-04	3.39E-04
(6561 nodes)	L2-norm	**3.44E-05**	3.46E-05	5.05E-05	5.13E-05	5.09E-05
~ 15%	Max. abs.	6.11E-02	**6.05E-02**	8.22E-02	8.09E-02	8.13E-02
(95x95)	L1-norm	1.81E-04	**1.52E-04**	2.28E-04	2.37E-04	2.40E-04
(9025 nodes)	L2-norm	2.68E-05	**2.32E-05**	3.42E-05	3.50E-05	3.51E-05
~ 20%	Max. abs.	6.50E-02	**6.44E-02**	9.01E-02	8.93E-02	8.94E-02
(111x111)	L1-norm	1.24E-04	**1.22E-04**	1.80E-04	1.85E-04	1.84E-04
(12321 nodes)	L2-norm	1.91E-05	**1.87E-05**	2.70E-05	2.75E-05	2.72E-05
Averaged	Max. abs.	73.27	**72.88**	101.16	100.00	100.22
Indices	L1-norm	69.29	**64.88**	96.77	100.00	100.04
(Base QUICK)	L2-norm	71.02	**67.23**	98.17	100.00	99.38

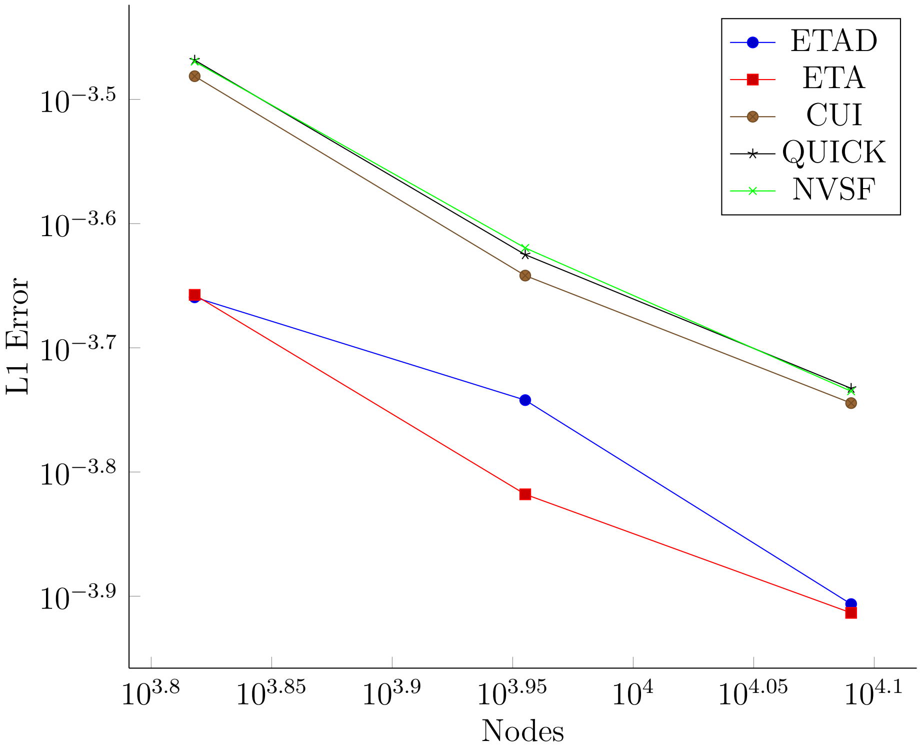

FIGURE F.3: L1 Error vs Nodes for $Re = 10000$ on structured grids with constant mesh growth factor of 1.2.

Table F.4: 2D Lid-Driven Cavity Re = 1000 error comparisons on non-equispaced unstructured grids with a constant growth factor of 1.2.

		Scheme:				
Mesh Size	**Error**	**ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa=\frac{1}{3}\right)$	**QUICK** $\left(\kappa=\frac{1}{2}\right)$	**NVSF** $\left(\kappa=\frac{3g-1}{2g}\right)$
~ 10%	Max. abs.	5.83E-03	**5.83E-03**	1.12E-02	1.05E-02	9.72E-03
(81x81)	L1-norm	1.75E-05	**1.75E-05**	3.26E-05	3.23E-05	2.65E-05
(6561 nodes)	L2-norm	2.80E-06	**2.80E-06**	5.67E-06	5.41E-06	4.71E-06
~ 15%	Max. abs.	4.76E-03	**4.70E-03**	1.12E-02	1.04E-02	9.70E-03
(95x95)	L1-norm	**8.64E-06**	8.91E-06	1.77E-05	1.70E-05	1.43E-05
(9025 nodes)	L2-norm	**1.43E-06**	1.45E-06	3.20E-06	2.97E-06	2.64E-06
~ 20%	Max. abs.	**5.78E-03**	5.78E-03	8.78E-03	8.20E-03	7.53E-03
(111x111)	L1-norm	**9.58E-06**	9.58E-06	1.77E-05	1.69E-05	1.42E-05
(12321 nodes)	L2-norm	**1.51E-06**	1.51E-06	3.04E-06	2.82E-06	2.51E-06
Averaged	Max. abs.	57.30	**57.11**	107.05	100.00	92.62
Indices	L1-norm	**53.90**	54.43	103.30	100.00	83.48
(Base QUICK)	L2-norm	**51.17**	51.41	106.76	100.00	88.33

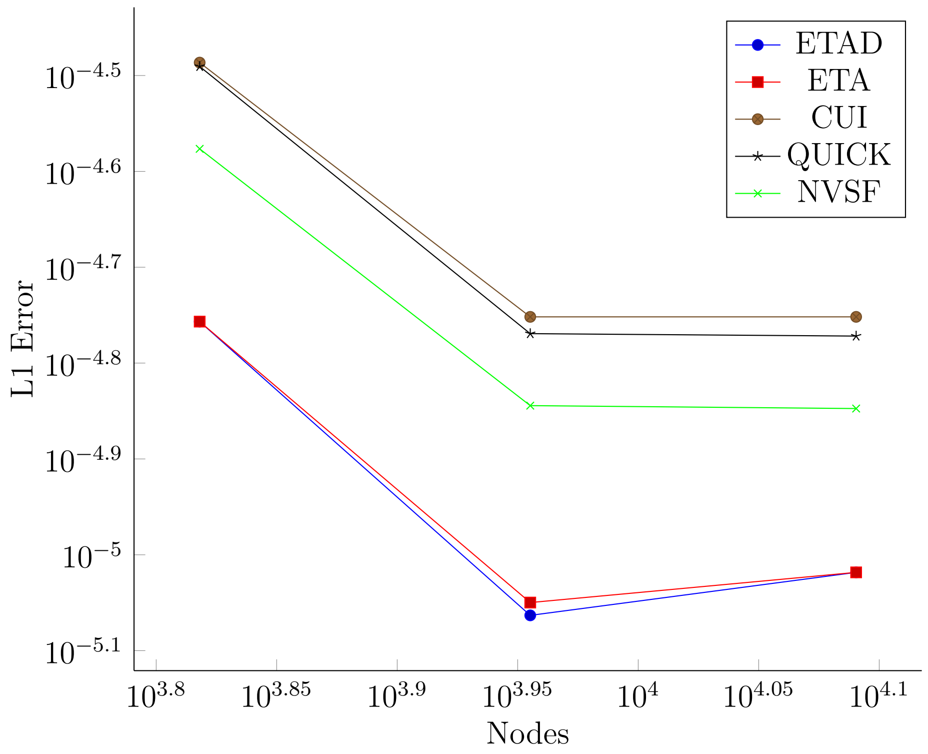

Figure F.4: L1 Error vs Nodes for Re = 1000 on unstructured grids with constant mesh growth factor of 1.2.

TABLE F.5: 2D Lid-Driven Cavity Re = 5000 error comparisons on non-equispaced ustructured grids with a constant growth factor of 1.2.

		Scheme:				
Mesh Size	**Error**	**ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
∼ 10%	Max. abs.	1.50E-02	**1.41E-02**	3.74E-02	3.41E-02	3.39E-02
(81x81)	L1-norm	5.22E-05	**4.56E-05**	1.05E-04	9.20E-05	9.12E-05
(6561 nodes)	L2-norm	8.29E-06	**7.44E-06**	1.98E-05	1.78E-05	1.73E-05
∼ 15%	Max. abs.	1.28E-02	**1.27E-02**	3.62E-02	3.20E-02	3.38E-02
(95x95)	L1-norm	**2.96E-05**	2.99E-05	8.13E-05	6.95E-05	7.36E-05
(9025 nodes)	L2-norm	**4.70E-06**	4.71E-06	1.35E-05	1.19E-05	1.19E-05
∼ 20%	Max. abs.	1.38E-02	**1.38E-02**	3.73E-02	3.40E-02	3.40E-02
(111x111)	L1-norm	2.14E-05	**2.14E-05**	5.73E-05	4.99E-05	4.98E-05
(12321 nodes)	L2-norm	3.51E-06	**3.51E-06**	1.05E-05	9.49E-06	9.25E-06
Averaged	Max. abs.	41.60	**40.58**	110.87	100.00	101.69
Indices	L1-norm	47.42	**45.15**	115.13	100.00	101.53
(Base QUICK)	L2-norm	41.05	**39.46**	111.83	100.00	98.30

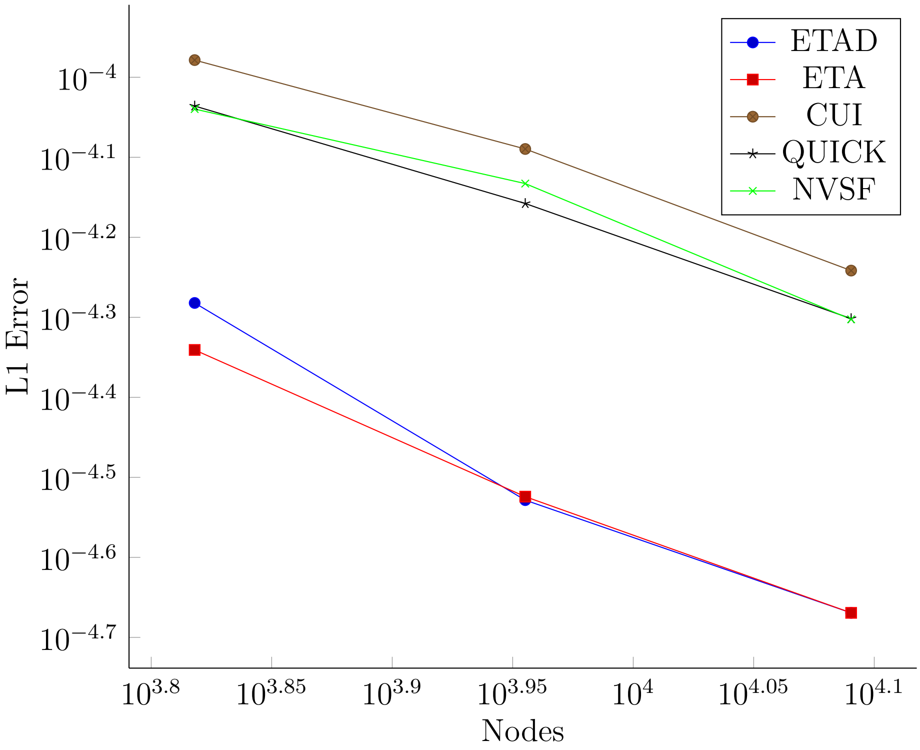

FIGURE F.5: L1 Error vs Nodes for Re = 5000 on unstructured grids with constant mesh growth factor of 1.2.

TABLE F.6: 2D Lid-Driven Cavity $Re = 10000$ error comparisons on non-equispaced unstructured grids with a constant growth factor of 1.2.

		Scheme:				
Mesh Size	**Error**	**ET** (Adv. & Dif.)	**ET** (Adv.)	**CUI** $\left(\kappa = \frac{1}{3}\right)$	**QUICK** $\left(\kappa = \frac{1}{2}\right)$	**NVSF** $\left(\kappa = \frac{3g-1}{2g}\right)$
∼ 10%	Max. abs.	**3.14E-02**	3.15E-02	7.20E-02	6.49E-02	6.75E-02
(81x81)	L1-norm	**1.01E-04**	1.01E-04	2.46E-04	2.19E-04	2.16E-04
(6561 nodes)	L2-norm	**1.69E-05**	1.69E-05	4.15E-05	3.72E-05	3.69E-05
∼ 15%	Max. abs.	**2.73E-02**	2.73E-02	6.46E-02	5.82E-02	6.26E-02
(95x95)	L1-norm	**6.68E-05**	6.70E-05	1.75E-04	1.56E-04	1.55E-04
(9025 nodes)	L2-norm	**1.06E-05**	1.06E-05	2.84E-05	2.56E-05	2.54E-05
∼ 20%	Max. abs.	**3.32E-02**	3.32E-02	7.21E-02	6.48E-02	6.76E-02
(111x111)	L1-norm	5.95E-05	**5.95E-05**	1.36E-04	1.20E-04	1.19E-04
(12321 nodes)	L2-norm	**9.40E-06**	9.40E-06	2.23E-05	1.99E-05	1.99E-05
Averaged	Max. abs.	**48.81**	48.88	110.96	100.00	105.22
Indices	L1-norm	**46.14**	46.24	112.72	100.00	99.14
(Base QUICK)	L2-norm	**44.68**	44.71	111.48	100.00	99.41

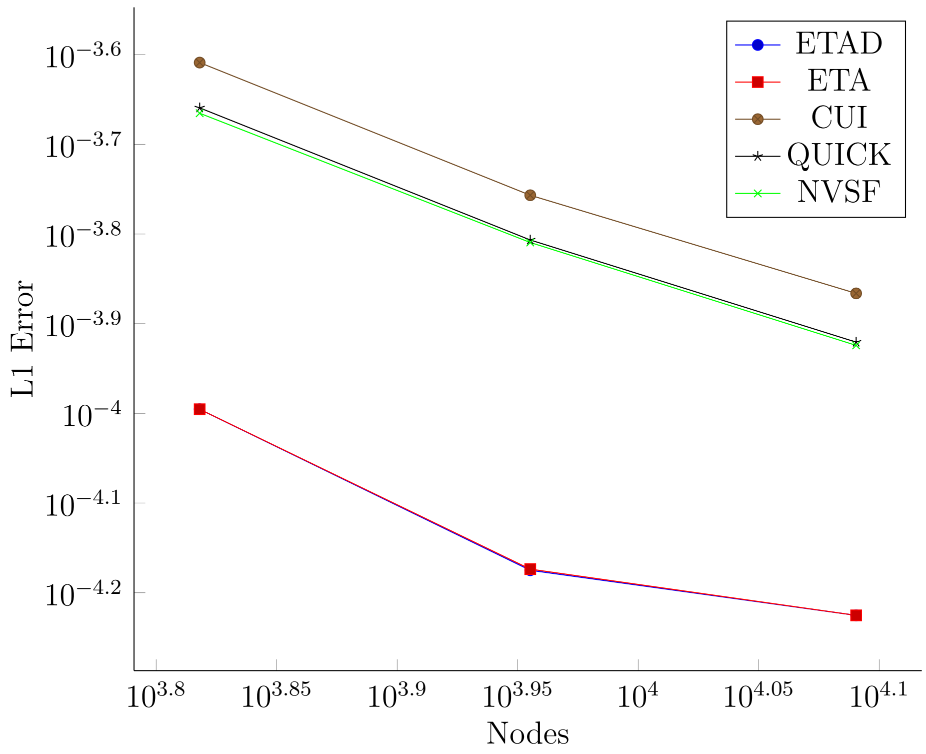

FIGURE F.6: L1 Error vs Nodes for $Re = 10000$ on unstructured grids with constant mesh growth factor of 1.2.

Bibliography

[1] R.K. Agarwal. A third-order-accurate upwind scheme for Navier-Stokes solutions at high reynolds numbers, AIAA paper 1981-112. In *19th AIAA Aerospace Sciences Meeting, St. Louis, MO, USA*, 1981.

[2] T.J. Barth and B. Jespersen. The design and application of upwind schemes on unstructured meshes, AIAA paper 89-0366. In *27th AIAA Aerospace Sciences Meeting, Reno, NV, USA*, 1989.

[3] M. Berger, M.J. Aftosmis, and S.M. Murman. Analysis of slope limiters on irregular grids, AIAA paper 2005-0490. In *43rd AIAA Aerospace Sciences Meeting, Reno, NV, USA*, 2005.

[4] M.S. Darwish. A new high-resolution scheme based on the normalized variable formulation. *Numerical Heat Transfer, Part B*, 24:353–371, 1993.

[5] M.S. Darwish and F.H. Moukalled. Normalized variable and space formulation methodology for high-resolution schemes. *Numerical Heat Transfer, Part B*, 26:79–96, 1994.

[6] M.S. Darwish and F.H. Moukalled. TVD schemes for unstructured grids. *International Journal of Heat and Mass Transfer*, 46:599–611, 2003.

[7] E. Erturk, T.C. Corke, and C. Gokcol. Numerical solutions of 2-D steady incompressible driven cavity flow at high reynolds numbers. *International Journal for Numerical Methods in Fluids*, 48:747–774, 2005.

[8] J.E. Fromm. A method for reducing dispersion in convective difference schemes. *Journal of Computational Physics*, 3:176, 1968.

[9] P.H. Gaskell and A.K.C. Lau. Curvature compensated convective transport: SMART, a new boundedness preserving transport algorithm. *International Journal for Numerical Methods in Fluids*, 8:617–641, 1988.

[10] U. Ghia, K.N. Ghia, and C.T. Shin. High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. *Journal of Computational Physics*, 48:387–411, 1982.

[11] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 49:357–393, 1983.

[12] A. Harten. On a class of high resolution total-variation stable finite difference schemes. *SIAM Journal on Numerical Analysis*, 21:1, 1984.

[13] M.E. Hubbard. Multidimensional slope limiters for MUSCL-type finite volume schemes on unstructured grids. *Journal of Computational Physics*, 155: 54–74, 1999.

[14] H. Jasak, H.G. Weller, and A.D. Gosman. High resolution NVD differencing scheme for arbitrarily unstructured meshes. *International Journal for Numerical Methods in Fluids*, 31:431–449, 1999.

[15] B.P. Leonard. The QUICK algorithm: a uniformly third-order finite difference method for highly convective flows. *Computer Methods in Applied Mechanics and Engineering*, 19:59, 1979.

[16] B.P. Leonard. Simple high-accuracy resolution program for convective modelling of discontinuities. *International Journal for Numerical Methods in Fluids*, 8:1291–1318, 1988.

[17] O.F. Oxtoby and A.G. Malan. A matrix-free, implicit, incompressible fractional-step algorithm for fluid-structure interaction applications. *Journal of Computational Physics*, 231:5389–5405, 2012.

[18] P.K. Sweby. High resolution schemes using flux-limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 21:995, 1984.

[19] B. van Leer. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *Journal of Computational Physics*, 32:101, 1979.

[20] B. van Leer. Upwind-difference methods for aerodynamics problems governed by the euler equations. In *Lectures in Applied Mathematics*. 1985.

[21] R.F. Warming and R.M. Beam. Upwind second-order difference schemes and application in aerodynamics flows. *AIAA Journal*, 14 (9):1241, 1976.

[22] N.P. Waterson and H. Deconinck. Design principles for bounded higher-order convection schemes - a unified approach. *Journal of Computational Physics*, 224:182–207, 2007.

[23] P.L. Woodfield, K. Suzuki, and K. Nakabe. A simple strategy for constructing bounded convection schemes for unstructured grids. *International Journal for Numerical Methods in Fluids*, 46:1007–1024, 2004.