A-Posteriori Error Estimation Using Higher Moments in Computational Fluid Dynamics

A thesis submitted to the University of Manchester for the degree of Doctor of Engineering in the Faculty of School of Mechanical, Aerospace and Civil Engineering

2015

Stuart Russant

Faculty of Engineering and Physical Sciences
Abstract

A-Posteriori Error Estimation Using Higher Moments in Computational Fluid Dynamics

Stuart Russant
University of Manchester
2015

In industrial situations time is expensive and simulation accuracy is not always investigated because it requires grid refinement studies or other time consuming methods. With this in mind the goal of this research is to develop a method to assess the errors and uncertainties on computational fluid dynamics (CFD) simulations that can be adopted by industry to meet their requirements and time constraints.

In a CFD calculation there are a number of sources of errors and uncertainties. An uncertainty is a potential deficiency that is due to a lack of knowledge of an activity of the modelling process, for example turbulence modelling. An error is defined as a recognisable deficiency that is not due to a lack of knowledge, for example numerical discretisation error. The process of determining the level of errors and uncertainties is termed verification and validation. The work aims to define an error estimation method for verification of numerical errors that can be produced during one simulation on a single grid.

The second moment solution error estimate for scalar and vector quantities was proposed to meet these requirements. Where the governing equations of CFD, termed the first moments, represent the transport of primary variables such as the velocity, the second moments represents the transport of the primary variables squared such as the total kinetic energy. The second moments are formed by a rearrangement of the first moments. Based on a mathematical justification, an error estimate for vector or scalar quantities was defined from combinations of the solutions to the first and second moments.

The error estimate was highly successful when applied to six test cases using laminar flow and scalar transport. These test cases used either central differencing with Gaussian elimination, or the finite volume method with the CFD solver Code_Saturne to conduct the simulations, demonstrating the applicability of the error estimate across solution methods. Comparisons were made to the numerical simulation errors, which were found using either the analytical or refined solutions. The comparisons were aided by the normalised cross correlation coefficient, which compared the similarity of the
shape prediction, and the averaged summation coefficients, which compared the scale prediction.

When using the first order upwind scheme the method consistently produced good predictions of the locations of error. When using the second order centred or second order linear upwind schemes there was similar success, but limited by influences from solution unboundedness, non-resolution of the boundary layer, the near-wall gradient approximation, and numerical pressure error. At high Reynolds numbers these caused the prediction of the location of error to degrade. This effect was made worse when using the second order schemes in conjunction with the constant value boundary condition. This was the case for the scalar or velocity simulations, and is caused by the unavoidable drop to first order accuracy during the near-wall gradient approximation that is required for the second moment source term approximation.

The prediction of the scale demonstrated a dependence on the cell Peclet number. Below cell Peclet number 4 the increase of the estimate scale was linearly related to the increase of the error scale. The estimate scale consistently over-predicts by up to a factor of 3. This allows confidence that the true error level is below that which is predicted by the error estimate. At cell Peclet numbers greater than 4 the relationship between the scales remained linear, however, the estimate begins to under-predict the estimate. The exact relation becomes case dependent, and the highest under-prediction was by a factor of 10. In such circumstances a computationally inexpensive calibration can be done.
Acknowledgments

I would like to express my sincere gratitude to Prof D. Laurence and Dr M. Rabbitt for the creation of this EngD project, and the sponsorship provided by EDF Energy.

I also wish to thank and express my appreciation to my supervisors, Prof H. Iacovides and Dr S. Utyuzhnikov, for their continued input, support and dedication to the completion of this work.

Finally, I would like to thank D. R. Balfour for his continued support throughout the course of this EngD project.
Declarations

No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the “Copyright”) and he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made only in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the thesis, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://documents.manchester.ac.uk/DocInfo.aspx?DocID=487), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see http://www.manchester.ac.uk/library/aboutus/regulations) and in The University’s policy on Presentation of Theses.
## Contents

1 Mathematical Background and Theory ............................................... 29
   1.1 Governing Equations of Fluid Dynamics ....................................... 30
      1.1.1 Mass .............................................................................. 30
      1.1.2 Momentum ..................................................................... 31
      1.1.3 Advection of a Scalar ...................................................... 32
      1.1.4 Dimensionless Quantities .................................................. 32
         1.1.4.1 Reynolds Number ....................................................... 32
         1.1.4.2 Peclet Number .......................................................... 33
   1.2 CFD Analysis Process ................................................................. 34
      1.2.1 Pre-Processing Stage ........................................................ 34
         1.2.1.1 Formulate the Flow Problem and Modelling of the Flow Domain Geometry ................................................................................................................................. 34
         1.2.1.2 Generate the Grid ............................................................ 34
         1.2.1.3 Establish the Boundary and Initial Conditions ................. 35
         1.2.1.4 Modelling of Phenomena .............................................. 35
      1.2.2 Perform the Simulation .......................................................... 35
      1.2.3 Post-Processing the Simulation to Get the Results .................. 36
   1.3 Code_Saturne .............................................................................. 37
      1.3.1 Outline of the Code_Saturne Solver Process .......................... 37
      1.3.2 Discretisation of the Governing Equations ............................ 37
         1.3.2.1 Discretisation in Time .................................................... 38
         1.3.2.2 Discretisation in Space .................................................. 39
      1.3.3 Discretisation Schemes in Code_Saturne ............................... 41
         1.3.3.1 First Order Upwind Scheme (FO) .................................. 41
         1.3.3.2 Second Order Centred Scheme (SOC) ............................ 41
         1.3.3.3 Second Order Linear Upwind Scheme (SOLU): .............. 42
         1.3.3.4 Slope Test .................................................................... 42
      1.3.4 User Subroutines ................................................................. 42
      1.3.5 Convergence Criteria ........................................................... 43
   1.4 Chapter 1 Summary .................................................................... 44
2 Error Estimation Background and Theory

2.1 Error and Uncertainty in CFD Calculations
   2.1.1 Definition of Uncertainty
   2.1.2 Definition of Error
   2.1.3 Sources of Error and their Reduction
      2.1.3.1 Numerical Errors
      2.1.3.2 Modelling Uncertainties
      2.1.3.3 Application Uncertainties
      2.1.3.4 User Errors
      2.1.3.5 Code Errors
   2.1.4 Numerical Errors

2.2 Verification and Validation
   2.2.1 Verification
      2.2.1.1 Verification of a Code
      2.2.1.2 Verification of a Calculation
      2.2.1.3 Verification Assessment Process
   2.2.2 Validation
      2.2.2.1 Validation Assessment Process

2.3 Current Error Determination Methods
   2.3.1 Grid Refinement using Multiple Grids
      2.3.1.1 Richardson Extrapolation
   2.3.2 Higher Order Accuracy Solutions on the Same Grid
      2.3.2.1 Example Use of Higher Order solutions
   2.3.3 Auxiliary PDE solutions on the Same Grid
      2.3.3.1 Estimating Errors Using Higher Order Schemes
   2.3.4 Auxiliary Algebraic Evaluations on the Same Grid
      2.3.4.1 Estimating the Leading Truncated Term from the Taylor Series
      2.3.4.2 Second Moment Residual Method

2.4 Chapter 2 Summary

3 Proposed Estimation Method: Second Moment Solution Estimate Method

3.1 Creation and Solution of the Second Moment Equation
   3.1.1 Governing Equations of Motion in Fluid (First Moments)
   3.1.2 Manipulation of the Scalar Transport Equation into the Second Moment Equation
   3.1.3 Manipulation of the Momentum Equation into the Second Moment
   3.1.4 Solving the Second Moment Equations
   3.1.5 Combining the Solutions of the First and Second Moments

3.2 Creation of the Second Moment Solution Error Estimate
   3.2.1 Mathematical Analysis for Scalar Transport
      3.2.1.1 Shape Estimate
      3.2.1.2 Scale Estimate
3.2.2 Mathematical Analysis for Momentum Transport
3.2.2.1 Shape Estimate
3.2.2.2 Scale Estimate
3.2.3 Error Estimate Definitions
3.2.3.1 Scalar Error and Estimate
3.2.3.2 Momentum Error and Estimate
3.2.4 Summation Coefficients: Averaged Quantities to Compare the Scale Estimate
3.2.5 Cross Correlation Coefficient to Compare the Shape Estimate
3.3 Implementation in Code_Saturne
3.3.1 Simulation of the First Moment Equations
3.3.2 Simulation of the Second Moment Equations
3.4 Chapter 3 Summary

4 Test Cases: Preliminary Investigations
4.1 Descriptions of Preliminary Test Cases
4.1.1 Investigation of Convection-Diffusion Transport of \( f \) and \( q \) in 1D
4.1.1.1 Solving for \( f \) and \( q \)
4.1.2 Investigation of Convection-Diffusion of a Passive Scalar from a Point Source in a Crossflow
4.1.2.1 Solving for \( f \) and \( q \)
4.1.3 2D Navier-Stokes Solver Applied to Burggraf Flow
4.1.3.1 Solving for the Velocity \( u \) and the Total Kinetic Energy \( K \)
4.2 Errors and Error Estimates for the Preliminary Test Cases
4.2.1 1D Convection Diffusion
4.2.1.1 \( f \) Error and Error Estimate Distributions
4.2.1.2 Correlation Coefficient
4.2.1.3 Summation Error
4.2.2 Point Source in a Crossflow
4.2.2.1 \( f \) Error and Error Estimate
4.2.2.2 Correlation Coefficient
4.2.2.3 Summation Error
4.2.3 Burggraf Flow
4.2.3.1 Laminar Flow \( u \) Error and Error Estimate
4.2.4 Discussion of Preliminary Test Case Results
4.3 Chapter 4 Summary

5 Main Test Case Descriptions
5.1 Ribbed Channel
5.1.1 Laminar Velocity Solution
5.1.2 Scalar Solution in Laminar Flow
5.2 Impinging Flow
5.2.1 Laminar Velocity Solution
List of Figures

1.3.1 A diagram of a control volume. ......................................................... 40

4.1.1 Solution of the 1D convection diffusion equation for the five Peclet numbers with 50 grid points. ................................................................. 79

4.1.2 Solution of the second moment form of the 1D convection diffusion equation for the five Peclet numbers with 50 grid points. ............................ 79

4.1.3 A diagram of the flow setup and the geometry of the mesh for the point source in a cross flow test case. ................................................................. 81

4.1.4 Solution of the scalar transport equation for the point source in a cross flow test case in Code_Saturne. ................................................................. 82

4.1.5 Solution of the second moment equation for the point source in a cross flow test case in Code_Saturne. ................................................................. 82

4.1.6 The geometry of the Burggraf flow test case. ........................................... 83

4.1.7 The velocity stream lines for the Code_Saturne simulation of the Burggraf flow with coarse mesh. ................................................................. 84

4.1.8 The total kinetic energy for the Code_Saturne simulation of the Burggraf flow with coarse mesh. ................................................................. 85

4.2.1 The 1D convection-diffusion equation solution errors (left) and corresponding estimates (right), using the second order centred (top) and first order (bottom) discretisation schemes, for the five Peclet numbers. They are expressed as a percentage of the volume average of $f$. ................................................................. 86

4.2.2 The 1D convection-diffusion equation correlation coefficients for the two meshes (coarse and very coarse), the two discretisation schemes (first order upwind and second order centred), and the five Peclet numbers. ................................................................. 87

4.2.3 The summation error and summation estimate for the two meshes (coarse and very coarse) and the two discretisation schemes (first order upwind and second order centred). They are expressed as a percentage of the maximum value of $f$ found in each of the simulations. ................................................................. 87

4.2.4 Examples of the exact errors for the point source in a cross flow Code_Saturne solution with $u = 1$. The error distributions include a contour outlining the values at 5% of the maximum value of $f$ for each. ................................................................. 88
4.2.5 Point source in a cross flow test case correlation coefficients against $Pe_L$ for the four simulations with two meshes and two discretisation schemes.

4.2.6 The summation error and estimate for the point source in a cross flow test case. Displayed as error against estimate for the two discretisation schemes: FO and SOC. They are expressed as a percentage of the volume average of $f$.

4.2.7 Examples of the Burggraf flow error and estimate distributions with $Re = 10$.

5.1.1 A diagram of the ribbed channel geometry indicating the direction of flow and the non-zero scalar boundaries.

5.1.2 The two coarse meshes used in ribbed channel test case.

5.1.3 Examples of the velocity streamlines for the ribbed channel test case. The scale was normalised by $U_b$.

5.1.4 The $U$ and $K$ solutions for the ribbed channel test case found using the three mesh densities at $Re = 674$. The velocity scale was normalised by $U$, and the total kinetic energy solution by $\frac{U^2}{2}$.

5.1.5 The convergence of the ribbed channel simulation with $Re = 674$ on the coarse mesh, displayed as the volume averaged velocity, $\langle U \rangle$, and total kinetic energy, $\langle K \rangle$, divided by their respective final values, against iteration number.

5.1.6 The $f$ solutions for $Re = 674$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the bottom walls.

5.1.7 The $q$ solutions for $Re = 674$ found using the three mesh densities and with constant flux (left) and constant value (right) boundary conditions along the bottom walls.

5.1.8 The convergence of the ribbed channel simulation with $Re = 674$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number.

5.2.1 The geometry of the 2D impinging flow test case, indicating the relative dimensions and the fluid directions.

5.2.2 The two coarse meshes used in the impinging flow test case.

5.2.3 Examples of the velocity streamlines for the impinging flow test case. The scale was normalised by $U_b$.

5.2.4 The $U$ and $K$ solutions for the impinging flow test case found using the three mesh densities at $Re = 1000$. The velocity scale was normalised by $U$, and the total kinetic energy solution by $\frac{U^2}{2}$.

5.2.5 The convergence of the impinging flow simulation with $Re = 1000$ on the coarse mesh, displayed as the volume averaged velocity, $\langle U \rangle$, and total kinetic energy, $\langle K \rangle$, divided by their respective final values, against iteration number.

5.2.6 The $f$ solutions for $Re = 1000$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the bottom wall.

5.2.7 The $q$ solutions for $Re = 1000$ found using the three mesh densities and with constant flux (left) and constant value (right) boundary conditions along the bottom wall.
5.2.8 The convergence of the impinging flow simulation with $Re = 1000$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number. .......................... 109

5.3.1 A diagram to demonstrate the stacked tube bundle set up and highlight the flow geometry that was simulated. ................................................................. 110

5.3.2 The geometry of the meshes used to simulate flow through stacked tube bundles. Red lines are heated, blue lines are not. .......................................................... 111

5.3.3 The two coarse meshes used in the tube bundle test case. On the left side of the very coarse mesh five sections are labeled and outlined. ........................................ 111

5.3.4 Examples of the velocity streamlines for the tube bundles test case. The scale was normalised by $U_b$. ................................................................. 113

5.3.5 Example of the total kinetic energy solution for the tube bundles test case. The scale was normalised by $\frac{U_b^2}{2}$. ................................................................. 114

5.3.6 The convergence of the tube bundle simulation with $Re = 144$ on the coarse mesh, displayed as the volume averaged velocity, $\langle U \rangle$, and total kinetic energy, $\langle K \rangle$, divided by their respective final values, against iteration number. ........................................ 114

5.3.7 The $f$ solutions for $Re = 144$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the central cylinder wall. 116

5.3.8 The $q$ solutions for $Re = 144$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the central cylinder wall. 117

5.3.9 The convergence of the tube bundle simulation with $Re = 144$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number. ........................................ 117

6.1.1 An example of the $f$ error and estimate from the ribbed channel with constant flux boundary simulations. They were found with $Re = 175$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.82. ......................................................... 120

6.1.2 An example of the $f$ error and estimate from the ribbed channel with constant flux boundary simulations. They were found with $Re = 175$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.89. ......................................................... 121

6.1.3 An example of the $f$ error and estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.44. ......................................................... 121

6.1.4 An example of the $f$ error and estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.53. ......................................................... 122
6.1.5 An example of the $f$ error and estimate from the tube bundles with constant flux boundary simulations. They were found with $Re = 79.9$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.91.

6.1.6 An example of the $f$ error and estimate from the tube bundles with constant flux boundary simulations. They were found with $Re = 79.9$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.51.

6.1.7 Correlation coefficients plotted against $Re$ for the ribbed channel with constant flux boundary simulations and using three discretisation schemes; FO, SOC and SOLU.

6.1.8 Two examples of the error and estimate distributions for the ribbed channel with constant flux boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.1.9 Correlation coefficients plotted against $Re$ for the impinging flow with constant flux boundary simulations and using three discretisation schemes; FO, SOC and SOLU.

6.1.10 Two examples of the error and estimate distributions for the impinging flow with constant flux boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.1.11 Correlation coefficients plotted against $Re$ for the tube bundles with constant flux boundary simulations and using three discretisation schemes; FO, SOC and SOLU.

6.1.12 Two examples of the error and estimate distributions for the tube bundles with constant flux boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.2.1 An example of the $f$ error and estimate from the ribbed channel with constant value boundary simulations. They were found with $Re = 175$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.62.
6.2.2 An example of the $f$ error and estimate from the ribbed channel with constant value boundary simulations. They were found with $Re = 175$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.58.

6.2.3 An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.74.

6.2.4 An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.42.

6.2.5 An example of the $f$ error and estimate from the tube bundles with constant value boundary simulations. They were found with $Re = 79.9$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.65.

6.2.6 An example of the $f$ error and estimate from the tube bundles with constant value boundary simulations. They were found with $Re = 79.9$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.15.

6.2.7 Correlation coefficients plotted against $Re$ for the ribbed channel with constant value boundary simulations and using three discretisation schemes; FO, SOC and SOLU.

6.2.8 Two examples of the error and estimate distributions for the ribbed channel with constant value boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.2.9 Correlation coefficients plotted against $Re$ for the impinging flow with constant value boundary simulations and using three discretisation schemes; FO, SOC and SOLU.

6.2.10 Two examples of the error and estimate distributions for the impinging flow with constant value boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.2.11 Correlation coefficients plotted against $Re$ for the tube bundles with constant value boundary simulations and using three discretisation schemes; FO, SOC and SOLU.

6.2.12 Two examples of the error and estimate distributions for the tube bundles with constant value boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.2.13 Plots of the summation coefficients for the ribbed channel with constant value boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. 

18
6.2.14 Plots of the summation coefficients for the impinging flow with constant value boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. 144

6.2.15 Plots of the summation coefficients for the tube bundles with constant value boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. 145

6.3.1 A profile of the error from the second order centred ribbed channel simulation using the very coarse mesh, constant flux boundary and $Re = 1640$. The profile ran from the top of the upstream rib to the top of the downstream rib. 148

6.3.2 A diagram of a near-wall cell with a superimposed graph of a solved for variable (red) against wall normal distance. 149

6.3.3 Examples of summation error expressed as a percentage of the volume average of $f$ against Reynolds number for the three test cases. 152

6.3.4 Examples of summation estimate expressed as a percentage of the volume average of $f$ against Reynolds number. 153

6.4.1 A diagram showing the very coarse mesh cells $a - f$ (bold lines), and the coarse mesh cells (thin lines) created from their regular refinement. 5 coarse mesh cell centres are labelled, as well as the very coarse mesh boundary wall face centres. 156

6.4.2 An example of the $f$ error and RE estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.96. 157

6.4.3 An example of the $f$ error and estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.69. 157

6.4.4 An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.97. 157

6.4.5 An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.57. 158

6.4.6 Correlation coefficients produced by the RE and SMSE plotted against $Re$ for the impinging flow simulations and using two discretisation schemes; FO and SOC. 159

6.4.7 Plots of the summation coefficients produced by the RE and SMSE for the ribbed channel simulations, using two discretisation schemes; FO and SOC. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. 160
7.1.1 An example of the $u$ error and estimate from the ribbed channel simulations. They were found with $Re = 175$, using the first order upwind scheme on the coarse mesh, expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.81.

7.1.2 An example of the $u$ error and estimate from the ribbed channel simulations. They were found with $Re = 175$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.12.

7.1.3 An example of the $u$ error and estimate from the impinging flow simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.54.

7.1.4 An example of the $u$ error and estimate from the impinging flow simulations. They were found with $Re = 375$, using the second order centred scheme and on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.17.

7.1.5 An example of the $u$ error and estimate from the tube bundles simulations. They were found with $Re = 79.9$, using the first order upwind scheme and on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.60.

7.1.6 An example of the $u$ error and estimate from the tube bundles simulations. They were found with $Re = 79.9$, using the second order centred scheme and on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.81.

7.1.7 Correlation coefficients plotted against $Re$ for the ribbed channel velocity simulations using three discretisation schemes; FO, SOC and SOLU.

7.1.8 Two examples of the error and estimate distributions for the ribbed channel velocity simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $u$.

7.1.9 Correlation coefficients plotted against $Re$ for the impinging flow velocity simulations using three discretisation schemes; FO, SOC and SOLU.

7.1.10 Two examples of the error and estimate distributions for the impinging flow velocity simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $u$.

7.1.11 Correlation coefficients plotted against $Re$ for the tube bundles velocity simulations using three discretisation schemes; FO, SOC and SOLU.

7.1.12 Two examples of the error and estimate distributions for the tube bundles velocity simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $u$. 

20
7.1.13 Plots of the summation coefficients for the ribbed channel velocity simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. .......................... 173

7.1.14 Plots of the summation coefficients for the impinging flow velocity simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. .......................... 174

7.1.15 Plots of the summation coefficients for the tube bundles velocity simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. .......................... 175
List of Tables

4.1.1 Summary of $Pe_L$ tested and the corresponding velocities on both of the meshes for the point source in a cross flow test case. ........................................ 81
4.2.1 Summary of the Burggraf flow summation and correlation coefficients with $Re = 10$. . . 91
5.1.1 Table showing the Reynolds numbers tested in the ribbed channel test case and the corresponding cell Peclet numbers. ........................................ 97
5.2.1 Table showing the Reynolds numbers tested in the impinging flow test case and the corresponding cell Peclet numbers. ........................................ 104
5.3.1 Table showing the Reynolds numbers tested in the tube bundle test case and the correspond- ing cell Peclet numbers. ........................................ 112
6.3.1 Summary of breakdown $Pe_L$ for both mesh densities and boundary conditions. ....... 154
Glossary of Terms

Latin

A - Discretisation matrix
A - Surface area of a discretised cell
\(A_{CM}\) - Surface area of a control mass
\(A_{CV}\) - Surface area of a control volume
a - Acceleration
b - Body forces acting on fluid
\(b_i\) - Source term to the discretisation matrix equation
C - Modelling constants identified with subscripts
D - Strain rate tensor
dc - Characteristic length of a discretisation cell
dL - Characteristic length
F - Force
\(F_{ij}\) - Control volume face fluxes
f - The scalar solution of the first moment scalar equation
\(f_e\) - The exact solution of the first moment scalar equation
I - Identity matrix
K - The total kinetic energy simulation solution
\(K_e\) - The exact kinetic energy
k - Turbulent kinetic energy
L - Length
\(l_m\) - Mixing length
\(N_{cc}\) - Normalised cross correlation coefficient
n - Wall normal vector
n - The time step
P - Pressure
Pe - Peclet number
\(Pe_L\) - Cell Peclet number
\(P_k\) - Production of \(k\)
q - The solution of the second moment scalar equation
\(q_e\) - The exact solution of the second moment scalar equation
Re - Reynolds number
r - The distance of a point from the origin
S - Area vector
S - Source term to the scalar first moment equation
\(S_e\) - Source term
\(S_i\) - Source term
\(S_k\) - Source of \(k\)
$S_{\varepsilon}$ - Source of $\varepsilon$
$U$ - Velocity magnitude
$U_B$ - Bulk velocity
$\mathbf{u}$ - Velocity vector simulation solution
$u_e$ - Exact velocity vector distribution
$u$ - The $x$ component of the velocity vector
$V_o$ - Set of control volumes
$v$ - The $y$ component of the velocity vector
$w$ - The $z$ component of the velocity vector
Greek

\( \beta \) - Diffusivity of the scalars \( f \) and \( q \)
\( \beta^* \) - Turbulent modelling constant
\( \Gamma \) - Source term to the rate of change of an extensive property
\( \Delta t \) - Time step
\( \delta \) - Error distribution on the \( f \) solution
\( \varepsilon \) - Total error distribution on the \( q \) solution
\( \varepsilon_d \) - Error distribution on the \( q \) solution caused by solving discretised equations
\( \varepsilon_f \) - Error distribution on the \( q \) solution caused by inaccurate second moment source terms
\( \epsilon \) - Turbulent dissipation rate
\( \eta \) - The total error distribution on the total kinetic energy solution
\( \eta_d \) - Error distribution on the \( K \) solution caused by solving discretised equations
\( \eta_a \) - Error distribution on the \( K \) solution caused by inaccurate second moment source terms
\( \theta \) - A variable denoting the time step used
\( \theta_F \) - The time step used for mass
\( \theta_S \) - The time step used for source terms
\( \theta_P \) - The time step used for physical variables
\( \mu \) - Molecular viscosity
\( \mu_t \) - Turbulent viscosity
\( \nu \) - Fluid viscosity
\( \xi_s \) - Estimate of the errors on the scalar solution
\( \xi_v \) - Estimate of the errors on the velocity solution
\( \rho \) - Fluid density
\( \sigma \) - Stress tensor
\( \sigma_x \) - Standard deviation of variable \( x \)
\( \sigma_k, \sigma_v, \sigma_\omega \) - Turbulent modelling constant
\( \varsigma \) - Velocity error scalar distribution on the velocity magnitude solution
\( \tau \) - Viscous stress tensor
\( \Phi \) - Extensive property
\( \phi \) - Intensive property
\( \Omega \) - Volume of discretised cells
\( \Omega_{CV} \) - Volume of a control volume
\( \Omega_{CM} \) - Volume of a control mass
\( \chi \) - Velocity error vector distribution on the vector velocity solution
\( \omega \) - Turbulent specific dissipation rate
Abbreviations

AIAA American Institute of Aeronautics and Astronautics
BPG Best Practice Guidelines
CAD Computer aided design
CFD Computational fluid dynamics
CM Control mass
CV Control volume
DNS Direct numerical simulation
EDF Électricité de France
EngD Engineering Doctorate
ERCOFTAC European Research Community on Flow, Turbulence and Combustion
FD Finite difference method
FE Finite element method
FO First order upwind discretisation scheme
FV Finite volume method
GUI Graphical user interface
KE Kinetic energy
LES Large eddy simulation
LHS Left hand side
PDE Partial differential equation
PIRT The Phenomena Identification and Ranking Table
PSC The point source in a cross flow
RANS Reynolds averaged Navier-Stokes equations
RE Richardson extrapolation
RHS Right hand side
ROI Regions of interest
SME Second moment equation
SMSE Second moment solution estimate method
SOC Second order centred discretisation scheme
SOLU Second order linear upwind discretisation scheme
SST Shear stress transport
Introduction

Fluid flows and related phenomena can be described by partial differential equations, termed the governing equations, that cannot be solved analytically except in special cases and so require the use of numerical methods to provide solutions. To obtain an approximate solution numerically a discretisation method must be applied to small domains in space and/or time. These approximate the differential equations by a system of algebraic equations so that the numerical solution provides results at discrete locations in space and time. However, to find a solution to a complex fluid dynamics problem would generally require the repetitive manipulation of thousands, or even millions, of numbers, which could not be achieved until creation of the high-speed digital computer [14][45][56].

Once the power of computers had been recognised interest in numerical techniques increased dramatically, and the increasing ability to simulate problems of more and more detail and sophistication was intimately related to advances in computer hardware. Particularly in regard to storage and execution speed which has been increasing in power according to Moore’s Law since the 1950s. Solution of the equations of fluid mechanics on computers has become so important that it now occupies up to a third of all research into fluid mechanics and is still increasing. This field is known as Computational Fluid Dynamics (CFD) and one of the strongest forces driving the development of new supercomputers is coming from the CFD community [45][46][56].

The problem caused by the rapid development of computers was that the universe of possible problems is so extensive, and the power of simulations so great, that CFD practitioners often focused on qualitative simulation of the next more difficult problem class that the computer capabilities just barely allowed, rather than on achieving quantitative accuracy on the previous problem class. The result of this situation sometimes has caused a decrease in the quality of simulations published. At the 1980-81 Air Force Office of Scientific Research Heat Transfer and Turbulent Mechanics conference on complex turbulent flows the conclusion of the evaluation committee was that, in most of the submissions to the conference, it was impossible to distinguish physical modelling errors from numerical errors related to the algorithm and grid. As an improvement to standards, the Journal of Fluids and Engineering stopped accepting papers reporting numerical fluid engineering solutions that fail to address the task of systematic truncation error testing and accuracy estimation, making it clear that it is not possible to infer an accuracy estimate from a single calculation on a fixed grid. The editors will also not consider a reasonable agreement with experimental data to be sufficient proof of accuracy, especially if any adjustable parameters are involved such as in turbulence modelling [27][45].

Blottner [9] commented on these issues distinguishing the two distinct sources of error from either inappropriate governing equations or inaccurate numerical solution procedure. Blottner also introduced the term credibility and noted the necessary requirement that computational fluid dynamicists perform grid refinement studies whenever they present numerical studies like experimentalists are expected to put error bounds on their results. Only once this is done can the needed credibility of CFD be established. Today there is a world-wide and general interest for quantifying errors and uncertainties in CFD studies across industry sectors, in industrial design, but particularly safety applications where demand for “error bars” around CFD results is now urgent.

Errors and uncertainties in CFD, which have a number of sources, can be identified and reduced by verification and validation methods or through spending more time running simulations. All of
these methods require time to be spent on further investigation, time which in industrial applications is monetarily expensive and so such in-depth error analysis may be overlooked leading to unreliable results. This EngD thesis sets out to meet both the time-saving and reliability goals of today’s industry. This is to be done through research into methods that can give information on the numerical errors in CFD simulations that use only one simulation and on a single grid, allowing quick, computationally inexpensive error analysis.

A recent effort to address this issue by Jasak and Gosman [26] used the property that analytical solutions of transport equations (for scalar and vector quantities) satisfy all higher moments to create an error estimate. Higher moments are rearrangements of the first moments, which are the equations that govern the transport of the primary variables such as velocity. The second moments are transport equations for the variables squared such as the total kinetic energy. In [26] the method calculated the residual of the second moment equation using the numerical solution of the first moment equation as input. The work here expands on this and investigates the use of the second moment equation to provide additional information that can reveal the scale and location of the numerical simulation errors not previously seen.

The proposed method, the second moment solution estimate (SMSE) method, uses the second moment equation to solve directly for the squared variable, using the first moment solution to approximate the additional source terms that appear. It is expected that the two solutions would be suitably comparable with slight differences between them appearing that will represent the first moment solution errors.

From the two solutions a full error estimate was defined that is mathematically related to the real errors allowing reliable error prediction. The estimate can be found on a single grid from one simulation at the expense of solving one additional transport equation. It is more computationally expensive than certain recent efforts such as that by Jasak and Gosman, but computationally cheaper than a full error analysis that requires multiple grid generation or higher order discretisation. Therefore this estimate meets the requirements this work sets out to address.

The thesis is separated into 8 chapters that build up the theory required for the definition of the proposed error estimation method before the estimate’s subsequent testing. The first three chapters present the theory. Chapter 1 describes the mathematical background and theory of computational fluid dynamics which allow numerical solvers to be created to find solutions to fluid dynamic problems. Chapter 2 introduces the concepts of errors and uncertainties, how they can be reduced and identified, and current methods available for error determination. Chapter 3 outlines the steps required to form the proposed error estimation method using the first and second moment equations.

The remaining chapters present the investigations conducted using the method. Chapter 4 outlines the preliminary test cases and the initial investigations conducted. Chapter 5 gives a description of the three main test cases. Chapter 6 presents and analyses the results found when using the scalar error estimate with the three main test cases using either a constant flux or constant value boundary condition. Chapter 7 presents and analyses the results found when using the velocity error estimate with the three main test cases. Finally, chapter 8 presents the conclusions of the work and suggests areas of further research.
Chapter 1

Mathematical Background and Theory

Chapter 1 defines the basic mathematics of fluid dynamics and the methods employed to simulate fluid flow problems. First a description of the mathematics is given, defining the governing equations of fluid dynamics. Mass, momentum and other scalars that are transported by fluids obey a set of equations that are typically non-linear, especially once the phenomenon of turbulence is included. The next section gives a brief description of turbulence and how it is included in the fluid dynamics equations. Finding solutions to such equations poses a problem, and turbulence modelling is required.

Typically, analytical solutions cannot be found to these equations and this leads on to the need for numerical methods, which again leads onto the use of computers. This has opened up the field of computational fluid dynamics (CFD), a continually developing area of study. A description of the CFD process is given, exploring how it is possible to simulate large fluid flow problems efficiently. Finally in this chapter, a description of the open source CFD software Code_Saturne is given. This software is used extensively in the work presented here.
1.1 Governing Equations of Fluid Dynamics

The governing equations of CFD derive from the basic conservation laws of mass, momentum and energy:

1) The mass of a fluid is conserved because mass cannot be created or destroyed.

2) The rate of change of momentum equals the sum of the forces on a fluid particle (Newton’s Second Law).

3) The rate of change of energy is equal to the sum of the rate of heat addition and the rate of work done on a fluid particle (first law of thermodynamics).

These fundamental principles can be expressed as mathematical statements, typically partial differential equations. A brief derivation of these governing equations follows, treating the fluid as a continuum, and considering the flow of a control mass (CM) with extensive properties such as mass, momentum and energy, etc. within a certain spatial region called a control volume (CV) with intensive properties mass, momentum etc. per unit volume [14][56].

For the conserved intensive property \( \phi \) of the CM with volume \( \Omega_{CM} \), density \( \rho \), and velocity field \( \mathbf{u} \), the corresponding extensive property \( \Phi \) is found from the control volume equation for the CV

\[
\Phi = \int_{\Omega_{CM}} \rho \phi d\Omega
\]

(1.1.1)

A conservation equation governs the rate of change of the extensive property, which is

\[
\frac{d\Phi}{dt} = \Gamma
\]

where \( \Gamma \) represents all contributions to the rate of change of \( \Phi \). The left hand side (LHS) of the conservation equation therefore yields:

\[
\frac{d}{dt} \int_{\Omega_{CM}} \rho \phi d\Omega = \frac{\partial}{\partial t} \int_{\Omega_{CV}} \rho \phi d\Omega + \int_{A_{CV}} \rho \phi \mathbf{u} \cdot \mathbf{n} dA
\]

(1.1.2)

with CV volume \( \Omega_{CV} \) and surface \( A_{CV} \). Equation 1.1.2 states that the rate of change of the amount of the property in the control mass, \( \Phi \), is the rate of change of the property within the control volume plus the net flux of it through the CV boundary due to fluid motion relative to the CV boundary. The first term is time dependent and is named the transient. The second term is called the convective term and represents the convective flux of \( \phi \) through the CV boundary. A more detailed derivation of this equation can be found in references [8, 14, 19 & 52], and the remaining analysis in this section draws upon them.

1.1.1 Mass

For the conservation of mass in the system, \( \phi = 1 \), and without the mass source term \( \Gamma \) the right hand side of the conservation equation is zero, leading to

\[
\frac{d}{dt} \int_{\Omega} \rho d\Omega + \int_{A} \rho \mathbf{u} \cdot \mathbf{n} dA = 0
\]

(1.1.3)
where \( \Omega \) and \( A \) are now the volume and surface area of the CV. Using the divergence theorem and allowing the control volume to become infinitesimally small leads to

\[
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0
\]  
(1.1.4)

For incompressible flow this becomes

\[
\nabla \cdot (\rho \mathbf{u}) = 0
\]  
(1.1.5)

Equation 1.1.4 can also be used to re-write the term \( \frac{\partial \rho \phi}{\partial t} \) as

\[
\rho \frac{\partial \phi}{\partial t} - \phi \nabla \cdot (\rho \mathbf{u})
\]  
(1.1.6)

This is the termed the continuity equation.

### 1.1.2 Momentum

For the motion of the fluid, \( \phi \) represents the momentum, \( \phi = \rho \mathbf{u} \). The conservation equation is essentially Newton’s second law per unit volume, \( \mathbf{F} = \rho \mathbf{a} \), applied to a continuum. The right hand side of the conservation equation is equal to the sum of the forces acting on the fluid particle, leading to

\[
\frac{d}{dt} \int_{\Omega} \rho \mathbf{u} d\Omega + \int_{A} \rho \mathbf{u} \mathbf{u} \cdot \mathbf{n} d\mathbf{A} = \sum b
\]  
(1.1.7)

With no other forces acting other than viscous effects, forces resulting from the turbulent stresses \( \sigma \), and body forces \( b \), and again using the divergence theorem and allowing the control volume to become infinitesimally small, in non-conservative form this becomes

\[
\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \cdot \mathbf{u}) = \nabla \cdot \sigma + \rho b
\]  
(1.1.8)

with the \( i^{th} \) Cartesian component

\[
\frac{\partial (\rho u_i)}{\partial t} + \nabla \cdot (\rho u_i u) = \nabla_j \sigma_{ij} + \rho b_i
\]  
(1.1.9)

The stress tensor \( \sigma \) for Newtonian fluids and laminar flow is

\[
\sigma = \tau - P I
\]  
(1.1.10)

where \( P \) is the pressure, \( I \) is the identity matrix and \( \tau \) is the viscous stress tensor [16]. \( \tau \) is defined from the dynamic molecular viscosity, \( \mu = \mu_t \), and the strain rate tensor, \( \mathbf{D} \), as

\[
\tau = 2\mu \mathbf{D} - \frac{2}{3} \nabla \cdot \mathbf{u} I
\]  
(1.1.11)

with

\[
\mathbf{D} = \frac{1}{2} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)
\]  
(1.1.12)
where \((\nabla \mathbf{u})^T\) is the transpose of \(\nabla \mathbf{u}\). With the stress tensor known, and in conservative form, using Equation 1.1.5 the momentum conservation equation becomes

\[
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{\mu}{\rho} \nabla^2 \mathbf{u} - \frac{\nabla P}{\rho} + \mathbf{b} \tag{1.1.13}
\]

with the \(i^{th}\) Cartesian component

\[
\frac{\partial u_i}{\partial t} + \mathbf{u} \cdot \nabla u_i = \frac{\mu}{\rho} \nabla^2 u_i - \frac{\nabla_i P}{\rho} + b_i \tag{1.1.14}
\]

making three equations, one for each Cartesian coordinate. The material derivative, \(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}\), represents the acceleration term in Newton’s second law. The other terms are the forces related to the pressure, \(\nabla P\), the diffusion, \(\nu \nabla^2 \mathbf{u}\), and \(\mathbf{b}\) represents the other sources force term. These equations, along with the continuity equation, are called the Navier-Stokes equations.

### 1.1.3 Advection of a Scalar

From the conservation equation with \(\phi\) representing a scalar quantity, \(f\), using Equation 1.1.6, and using Fick’s law with diffusivity, \(\beta\) to separate the diffusive source term from other additional source terms, \(S\), the advection of a scalar (such as enthalpy, temperature, concentrations, mass fractions etc.) becomes

\[
\frac{d}{dt} \int_{\Omega} \rho f \, d\Omega + \int_{A} \rho f \mathbf{u} \cdot \mathbf{n} dA = \int_{A} \beta \nabla f \cdot \mathbf{n} dA + \int_{\Omega} S \tag{1.1.15}
\]

Using the divergence theorem, Equation 1.1.5, and allowing the control volume to become infinitesimally small, this becomes

\[
\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla f = \beta \nabla^2 f + S \tag{1.1.16}
\]

where \(\beta\) and \(S\) have absorbed the division by \(\rho\) [16]. This equation is referred to as the scalar transport equation or the scalar first moment.

### 1.1.4 Dimensionless Quantities

Often it is convenient to present fluid dynamics results in a dimensionless form. This is done through the use of a number of dimensionless quantities. As long as the dimensionless quantities are the same even flows that have large variation of scales and fluid properties are comparable. The work presented here will make reference to two of these frequently and so this section will give a definition for the Reynolds number and the Peclet number.

#### 1.1.4.1 Reynolds Number

The Reynolds number, \(Re\), is a measure of the relative effect of inertial forces to viscous forces [6]. Considering Equation 1.1.13, the momentum equation, the Reynolds number can be defined informally
as the ratio of the convective term to the diffusion term
\[
Re = \frac{u \cdot \nabla u}{\frac{\rho}{\mu} \nabla^2 u}
\] (1.1.17)

If the terms are separated into their length scales it becomes
\[
Re = \frac{\rho [u][u]/[L]}{\mu [u]/[L]^2}
\] (1.1.18)

and rearrangement gives
\[
Re = \frac{\rho [u][L]}{\mu}
\] (1.1.19)

There is a choice of values to use for the velocity and length scales. Normally they are chosen as the bulk velocity of the considered flow, \(U_B\), and a characteristic length of the geometry, \(d_L\). This gives
\[
Re = \frac{\rho U_B d_L}{\mu}
\] (1.1.20)

### 1.1.4.2 Peclet Number

The Peclet number, \(Pe\), is a measure of the rate of convection of a physical quantity by the flow compared to the rate of diffusion of the same quantity \([37]\). A similar analysis can be followed, this time starting from Equation 1.1.16, the scalar transport equation. Defining the Peclet number as the ratio of the convection term with the diffusion term
\[
Pe = \frac{u \cdot \nabla f}{\beta \nabla^2 f}
\] (1.1.21)

then separating the terms into their length scales finds
\[
Pe = \frac{[u][f]/[L]}{\beta [f]/[L]^2}
\] (1.1.22)

which after rearranging is
\[
Pe = \frac{[u][L]}{\beta}
\] (1.1.23)

The same velocity and length scales can be used as for the Reynolds number and so the Peclet number is defined as
\[
Pe = \frac{U_B d_L}{\beta}
\] (1.1.24)

From this the cell Peclet number, \(Pe_L\), can be defined as
\[
Pe_L = \frac{U_B d_c}{\beta}
\]

which uses the discretisation cell size, \(d_c\), as the characteristic length.
1.2 CFD Analysis Process

Solving the governing equations of a fluid dynamics problem numerically involves first their discretisation into many algebraic equations, the discretisation of the fluid domain, and the solution of these equations by repeatedly improving upon calculations of their approximate solutions. The general process with the use of computers is outlined in this section drawing on references [14, 48 & 56].

Before starting a CFD simulation consideration should be given to the overall goal, i.e. what “numbers” are sought. The Phenomena Identification and Ranking Table (PIRT) [36] methodology is recommended for deciding this, particularly in the nuclear engineering sector. The solutions obtained through the CFD analysis process is only approximate because the steps involved in preparing a simulation and the numerical solution of the problem introduces errors from a number of sources. Here the process is briefly described and the points where errors are introduced are mentioned.

1.2.1 Pre-Processing Stage

Pre-processing consists of the input of a flow problem to a CFD program and its transformation into a form suitable for use by the solver. This will often lead to certain parts of the flow being simplified so that it fits with the models or limitations of the CFD software. The final simulation may therefore not match the true physical flow exactly. The main parts of the pre-processing stage are summarised in the following sections.

1.2.1.1 Formulate the Flow Problem and Modelling of the Flow Domain Geometry

The definition of the flow problem should consider the geometry of the flow domain. This includes the dimensionality required and identification of which parts of the geometry should and should not be included as regions of interest (ROI). A choice is then made to determine the computational domain while still being able to achieve the objectives of the analysis. The geometry and flow domain need to be modelled using a computer-aided design software (CAD) package to provide input for the grid generation with discretisation conforming to the boundaries of the ROI so that boundary conditions can be accurately represented. The extent of the finite flow domain in which the flow is to be simulated should be considered with approximations and simplifications of the geometry to reduce the need for computational resources. While care is taken so that these approximations will only have a negligible effect, it will introduce errors because the simulated flow is not the same as the real physical one.

1.2.1.2 Generate the Grid

To solve partial differential equations numerically, discretisation of the field into a collection of points or elemental volumes is required [53]. Grid generation is the sub-division of the domain into a finite number of smaller sub-domains creating a discretised grid of cells. These grids may be

- Structured – for simple geometries regular or structured grids consisting of groups of grid lines that do not cross each other and cross the members of other groups once only.

- Block-structured – for more complicated geometries the domain is divided into relatively large blocks which at the coarse resolution may or may not be structured, whereas at fine resolutions...
each block is structured.

- Unstructured – For complex geometries the geometry can be divided into fine cells of any shape allowing refined discretisation in any local area and, unlike structured grids, therefore avoids propagating extra grid-lines elsewhere.

The discretisation of the geometry for grid generation is the cause of a major source of error in CFD because information about the gradients of the solution is lost when it comes to solving the equations. This will be discussed in later sections. Another issue is that a complex geometry may have curves and other shapes that may not be captured by an unrefined mesh.

1.2.1.3 Establish the Boundary and Initial Conditions

Since the flow domain is finite, the domain boundaries must be defined with appropriate physical conditions to represent the whole flow problem, named the boundary conditions. Typical definitions would be walls, inlets, outlets, symmetry and periodic surfaces. These boundaries attempt to simulate the true physical flow, however they may still only be simplifications and so the CFD simulation will deviate from the physical solution.

1.2.1.4 Modelling of Phenomena

The selection of physical or chemical phenomena that are present in the flow to be modelled with or without simplifications must be decided, e.g. the nature of the viscous flow, or the choice of turbulence or chemistry model [15]. Modelling these phenomena introduces errors because the behaviour is only estimated rather than fully simulated. For example modelling turbulent flows with a range of models can produce quite different CFD solutions.

1.2.2 Perform the Simulation

As described in earlier sections, the flow of a fluid can be represented by a set of governing partial differential equations (PDE) that derive from simple conservation principles. There are three common choices of discretisation methods to solve these PDE by numerical methods: finite differences (FD), finite elements (FE) and finite volumes (FV).

A simulation using one of these methods generally starts from an initial solution and uses an iterative method to reach a final flow field solution that satisfies the PDEs. In the finite volume method, the method mainly considered in this report, the steps that a solver would take are:

1. Integration of the governing equations of fluid flow over all the (finite) control volumes of the domain.

2. Discretisation of the resulting integral equations converting them into a system of algebraic equations.

3. Solution of the algebraic equations by an iterative method. As the simulation proceeds, the solution is monitored to determine if a “converged” solution has been obtained where the difference between the solutions from one iteration to the next, the residual, is below a desired limit.

The iteration method reduces the error from the incompletely solved governing equations, however the error caused by the discretisation remains.
1.2.3 Post-Processing the Simulation to Get the Results

Post-Processing involves extracting the desired flow properties (thrust, lift, drag, heat transfer etc.) from the computed flow field. Care needs to be taken over how the raw data can be reduced from millions of numbers to present in the most meaningful way to fellow engineers and decision makers involved in the overall project. The PIRT can provide valuable guidelines for this.
1.3 **Code_Saturne**

In this work CFD simulations are performed using *Code_Saturne* version 2.0.1, an open source CFD code developed by EDF energy. It is a system designed to solve the Navier-Stokes equations in the cases of 2D, 2D axisymmetric or 3D flows. Its main module is designed for the simulation of flows which may be steady or unsteady, laminar or turbulent, incompressible or potentially dilatable, isothermal or not. The code includes specific modules, referred to as "specific physics", for the treatment of Lagrangian particle tracking, semi-transparent radiative transfer, gas combustion, pulverised coal combustion, electric effects (Joule effect and electric arcs) and compressible flows. This section gives a description of the solver process and discretisation schemes used in *Code_Saturne*, drawing upon references [5, 15 & 16].

*Code_Saturne* is composed of two main elements:

1. The Kernel module, which is the numerical solver
2. The pre-processor module which is in charge of mesh data reading, mesh pasting, domain decomposition for parallel computing and definition of periodicity boundary conditions.

*Code_Saturne* relies on the finite volume discretisation method and allows the use of various mesh types which may be hybrid (containing several kinds of elements) and may have structural non-conformities (hanging nodes).

1.3.1 **Outline of the Code_Saturne Solver Process**

At the beginning of a time step, the physical properties of the flow, \( \Phi \), are computed. A fractional step scheme is then used to solve the mass and momentum equations, first using the momentum equation to predict the velocity components, and then the mass equation is taken into account to solve a pressure Poisson equation before the mass fluxes at the cell faces are updated. If required, the equations for the turbulent variables are solved (turbulent kinetic energy and dissipation or Reynolds stresses and dissipation). For the \( k-\varepsilon \) model an additional step is carried out to couple the source terms. Next, the equations for the scalars (enthalpy, temperature, tracers, concentrations, mass fractions) are solved. Finally, all the variables are updated and another time step may start.

1.3.2 **Discretisation of the Governing Equations**

The general equation for the advection of a variable, \( f \) (e.g. a scalar such as temperature, or a velocity component), is re-written, after using Equation 1.1.6 to simplify the transient term, as

\[
\rho \frac{\partial f}{\partial t} + \nabla \cdot (\rho u f) - \nabla \cdot (\beta \nabla f) = S_i (\Phi, \varphi) f + S_\varepsilon (\Phi, \varphi) + \nabla \cdot (\rho u) f \tag{1.3.1}
\]

where \( \Phi \) represents the physical properties (\( \rho, \beta, \mu_t \), etc.); \( \varphi \) represents the variables of the problem (\( u, k, \varepsilon \), etc.); \( S_i (\Phi, \varphi) f \) represents the linear part of the source terms; \( S_\varepsilon (\Phi, \varphi) \) includes all other source terms; \( \nabla (\rho u) f \) is the term associated with “mass accumulation.”
1.3.2.1 Discretisation in Time

The time scheme used by the solver to solve Equation 1.3.1 is a $\theta$-scheme [32]. This allows the different variables to be evaluated at different times by selecting the value of $\theta$. The common values of $\theta$ used by Code_Saturne, although $\theta$ can vary away from these, are

\[
\begin{align*}
\theta & = 1 \quad \text{for an implicit first order Euler scheme evaluated at time step } n + 1 \\
\theta & = \frac{1}{2} \quad \text{for a second order Crank-Nicolson scheme evaluated at time step } n + \frac{1}{2} \\
\theta & = 0 \quad \text{for a standard explicit formulation evaluated at time step } n
\end{align*}
\]

If $\theta = \frac{1}{2}$ the time step $\Delta t$ is constant in time and uniform in space. The definition of these schemes is beyond the scope of this work and more information about the implicit, explicit and Crank-Nicolson schemes can be found in [13][14].

With the $\theta$-scheme applied to the variable $f$, the evaluation becomes

\[
f^{n+\theta} \equiv \theta f^{n+1} + (1 - \theta) f^n
\]

where superscripts indicate the time step at which it is evaluated. Calculation of the other terms is also carried out via a $\theta$-scheme.

**Physical Properties**

The physical properties of the flow are either defined at the time step $n$, which is explicit with $\theta_\Phi = 0$, or extrapolated at $n + \theta_\Phi$ using the Adam-Bashforth time scheme [1] where $\theta_\Phi = \frac{1}{2}$ or $\theta_\Phi = 1$. Generalised, the physical properties become

\[
\Phi \equiv \Phi^{n+\theta_\Phi} = (1 + \theta_\Phi) \Phi^n - \theta_\Phi \Phi^{n-1} \quad (1.3.2)
\]

**Mass Flux**

For the mass flux, three time schemes are available. The first two are explicit, and taken at time step $n$ for the momentum equations. They are also either taken at time step $n$ for the equations for turbulence and scalars or updated with their value at time step $n + 1$ (standard scheme). The third is implicit and taken at $n + \theta_F$ (again it is second order if $\theta_F = \frac{1}{2}$). To solve the momentum equations, $(\rho u)^{n-2+\theta_F}$ and $(\rho u)^{n-1+\theta_F}$ are known. Hence, the value at $n + \theta_F$ is obtained as a result of the extrapolation

\[
(\rho u)^{n+\theta_F} = 2 (\rho u)^{n-1+\theta_F} - (\rho u)^{n-2+\theta_F} \quad (1.3.3)
\]

At the end of this phase, $(\rho u)^{n+1}$ is known, and determining the mass flux at $n + \theta_F$ that will be adopted for the equations for turbulence and scalars uses the interpolation

\[
(\rho u)^{n+\theta_F} = \frac{1}{2 - \theta_F} (\rho u)^{n+1} - \frac{1 - \theta_F}{2 - \theta_F} (\rho u)^{n-1+\theta_F} \quad (1.3.4)
\]
Source Terms

Similar to the physical properties, the source terms are found explicitly using

\[ [S_e(\phi, \varphi)]^n = S_e(\phi^{n+\theta_S}, \varphi^n) \]  

(1.3.5)

or implicitly extrapolated at \( n + \theta_S \) using the Adams-Bashforth scheme as

\[ [S_e(\phi, \varphi)]^{n+\theta_S} = (1 + \theta_S) S_e(\phi^n, \varphi^n) - \theta_S S_e(\phi^{n-1}, \varphi^{n-1}) \]  

(1.3.6)

To be consistent and preserve the order of convergence in time, the implicit source terms are discretised with the same scheme as that used for convection-diffusion of the unknown being considered, i.e. taken at \( n + \theta \):

\[ [S_i(\phi, \varphi)f]^{n+\theta} = S_i(\phi^{n+\theta_S}, \varphi^n) \left[ \theta f^{n+1} + (1 - \theta) f^n \right] \]  

(1.3.7)

Discretisation of the Transient Term

The transient term becomes

\[ \frac{df^{n+\theta_i}}{dt} = \frac{f^{n+1} - f^n}{\Delta t} \]  

(1.3.8)

so it is second order accurate if the right hand side is at \( n + \frac{1}{2} \), i.e. \( \theta = \frac{1}{2} \).

1.3.2.2 Discretisation in Space

The discretisation in space uses the finite volume method [30][55]. The method integrates the transport equations over each cell of the mesh, which are referred to as control volumes. To illustrate the method, consider the \( i^{th} \) cell of the domain with volume \( \Omega_i \) and cell centre \( I \) as an example of a control volume, which is shown in Figure 1.3.1. The neighbouring cells to the \( i^{th} \) cell belong to the set \( V_o(i) \). One cell out of this set, cell \( j \), is also shown in Figure 1.3.1 with cell centre \( J \). The internal face that connects cells \( i \) and \( j \), \( ij \), is indicated and has area \( S_{ij} \). Boundary faces to cell \( i \) belong to the set \( \gamma_{b}(i) \). One face out of this set is labelled as \( ik \) and has area \( S_{bik} \).

Decomposition of the vector between the cell centres, \( \vec{IJ} \), into the basis formed by the face \( ij \) and the normal to the face, allows definition of the secondary points, \( I' \) and \( J' \). Their position is defined by the face centre and the perpendicular distance of \( I \) or \( J \) to the face. Similarly, a basis can be formed from the face \( ik \). The normal to the face \( ik \) that runs through \( I \) defines the point \( K \) on the face.

It is possible to transform the integration of the partial derivatives in the integrated equations into a sum of the numerical fluxes calculated at the internal and boundary faces, \( F_{ij} \) and \( F_{bik} \), respectively, using the divergence theorem. For the quantity \( \nabla \cdot \mathbf{Y} \) this gives

\[ \int_{\Omega_{i}} \nabla \cdot \mathbf{Y} d\Omega = \sum_{j \in V_{o}(i)} F_{ij}(\mathbf{Y}) + \sum_{k \in \gamma_{b}(i)} F_{bik}(\mathbf{Y}) \]  

(1.3.9)

A description of the spatial discretisations used in Code_Saturne for the convection and diffusion terms
follows. The subscripts represent locations where the values are taken. For example $Y_{ij}$ indicates the value of $Y$ at the centre of face $ij$, and $Y_I$ is the value of $Y$ at the point $I$. Similarly, $Y_{I'}$ indicates the value of $Y$ found at the point $I'$, however, this value is not known. Finding the values at $I'$ and $J'$ requires an interpolation, which is carried out according to

$$Y_{I'} = Y_I + \overrightarrow{II'} \cdot \nabla_i Y$$

where $\overrightarrow{II'}$ is the vector between the points $I$ and $I'$, and $\nabla_i Y$ is the gradient of $Y$. The reconstruction of the gradient of $Y$ is beyond the scope of this work and a more detailed description can be found in [16].

![Diagram of a control volume](image)

**Figure 1.3.1:** A diagram of a control volume.

**Discretisation of the Convective Term**

For the integration of the $i^{th}$ cell, the convective term in Equation 1.3.1, $-\nabla \cdot (\rho u f)$, can be written as

$$\int_{\Omega_i} \nabla \cdot (\rho u f) d\Omega = \sum_{j \in \mathcal{V}_o(i)} F_{ij}(\rho u, f) + \sum_{k \in \mathcal{\Gamma}_b(i)} F_{bk}(\rho u, f) \quad (1.3.10)$$

with

$$F_{ij}(\rho u, f) = [\frac{\rho u_{ij} \cdot S_{ij}}{f_{ij}}] f_{ij} \quad (1.3.11)$$

$$F_{bk}(\rho u, f) = [\frac{\rho u_{bk} \cdot S_{bk}}{f_{bk}}] f_{bk} \quad (1.3.12)$$

where $f_{ij}$ and $f_{bk}$ represent the values of $f$ at the internal and boundary faces of $\Omega_i$ respectively, and $S$ is the area vector. However the value of $f_{ij}$ and hence the convective flux $F_{ij}$ depends on the numerical convection scheme, which will be defined later in this section.

**Discretisation of the Diffusive Term**

Similarly, the diffusive part can be written as
\[
\int_{\Omega_i} \nabla \cdot (\beta \nabla f) \, d\Omega = \sum_{j \in V_{\alpha(i)}} F_{ij}(\beta, f) + \sum_{k \in \gamma_{\alpha(i)}} F_{bk}(\beta, f) 
\tag{1.3.13}
\]

with

\[
F_{ij}(\beta, f) = \beta_{ij} \frac{f_{IJ} - f_{I'J'}}{TT'} S_{ij} \tag{1.3.14}
\]

and

\[
F_{bk}(\beta, f) = \beta_{bk} \frac{f_{I'J'} - f_{II}}{IK} S_{bk} \tag{1.3.15}
\]

using the same notation as before, and \(TT'\) is the distance between points \(I'\) and \(J'\) etc. This is a second order approximation.

### 1.3.3 Discretisation Schemes in Code_Saturne

Three different types of convection schemes are available in Code_Saturne.

#### 1.3.3.1 First Order Upwind Scheme (FO)

\(f_{ij}\) is approximated by its value at the node upstream of the middle of the vector \(IJ\) using a backward- or forward-difference approximation for the first derivative depending on the flow direction.

\[
f_{ij} = \begin{cases} 
    f_I (\rho u)_{ij} \cdot S_{ij} \geq 0 \\
    f_I (\rho u)_{ij} \cdot S_{ij} < 0 
\end{cases} \tag{1.3.16}
\]

This unconditionally satisfies the boundedness criterion, but achieves this by being numerically diffusive [37]. Rapid variations and small features in the variables will be smeared out and, since the rate of error reduction is only first order, very fine grids are required to obtain accurate solutions [14].

#### 1.3.3.2 Second Order Centred Scheme (SOC)

A second order approximation for the value of \(f_{ij}\) at the control volume-face centre uses linear interpolation between the two nearest nodes using

\[
f_{ij} = \alpha_{ij} f_{II} + (1 - \alpha_{ij}) f_{IJ} \tag{1.3.17}
\]

where \(\alpha_{ij} = \frac{|x_{I'J'}|}{|IJ|}\), and \(x_{I'J'}\) is the intersection of the wall and the line \(I'J'\) defined at the internal faces only. This scheme is not as universally stable as the first order scheme. It has requirements on the size of the cell Peclet number, \(Pe_L = \frac{d u}{\mu}\); above a certain limit, typically \(Pe_L = 2\), solutions can become unbounded. This type of second order scheme can generate numerical oscillations that can cause the calculation to diverge. It can also lead to physical scalars taking unphysical values [14].
1.3.3.3 Second Order Linear Upwind Scheme (SOLU):

This is of second order accuracy but is more complex than the centred scheme. Second order approximations for $f_{ij}$ are found by upwind interpolation for the cell values using

$$
\begin{align*}
    f_{ij} &= \begin{cases} 
        f_I + \overrightarrow{F} \cdot (\nabla f)_I & (\rho u)_{ij} \cdot S_{ij} \geq 0 \\
        f_J + \overrightarrow{F} \cdot (\nabla f)_J & (\rho u)_{ij} \cdot S_{ij} < 0
    \end{cases}
\end{align*}$$

(1.3.18)

The value of $f_{bik}$ is calculated as

$$
\begin{align*}
    f_{bik} &= \begin{cases} 
        f_I (\rho u)_{bik} \cdot S_{bik} \geq 0 \\
        f'_{bik} (\rho u)_{bik} \cdot S_{bik} < 0
    \end{cases}
\end{align*}$$

(1.3.19)

where $f'_{bik}$ is the boundary value directly computed from the prescribed boundary conditions [14].

1.3.3.4 Slope Test

A slope test allows switching from the SOC or SOLU scheme to the first order upwind scheme. The slope test checks the gradient of scalars and in regions where it is large it switches to the more robust first order linear upwind scheme. Additionally, in standard mode, $f_{ij}$ is computed as a weighted average between the upstream value and the centred value. In this work the three discretisation schemes will be used with the slope test deactivated. Therefore the two second order schemes will be operating without limiters. This risks unphysical solutions, such as those produced through unboundedness [37], which causes a limitation on the maximum cell size.

1.3.4 User Subroutines

The user subroutines are FORTRAN f90 files that may be altered to define the flow problem inputs, overwrite components of Code_Saturne, define new components to be completed during a calculation and format output. Code_Saturne can run calculations with or without the user subroutines but parameters read in the interface are overwritten by those in the user subroutine. The name and function of the user subroutines referred to in this report are

- **usclim** - When the graphical user interface is used, usclim is used to define complex boundary conditions (input profiles, conditions varying in time, etc.) which could not be specified by means of the interface, and only these need to be defined. In the case of a calculation launched without the interface, all the boundary conditions must appear in usclim.

- **usinit** - This subroutine is used to indicate the value of different calculation basic parameters: constant and uniform physical values, parameters of numerical schemes, input-output management, etc. In the case of a calculation launched using the graphical interface, it is only used to modify high-level parameters which cannot be managed by the interface. In the case of a code utilisation without interface, this subroutine is compulsory and all the headings must be completed.

- **usproj** - This is used to print or modify any variable at the end of every time step.
• *ustsse* - In this subroutine additional sources are introduced to the scalar transport equations. It is here that the additional terms in the second moment equations or additional forces are included.

• *ustsns* - This subroutine allows additional sources to be added to the Navier-Stokes equations. It can be used to apply the forcing terms necessary to drive the fluid flow in periodic cases.

### 1.3.5 Convergence Criteria

To arrive at the simulation results for steady flows *Code_Saturne* uses an iterative scheme in time. Starting with a uniform flow field, or a fabricated flow field which is one predefined close to the expected result, iterations are performed until the steady-state flow field is obtained. This is termed iterative convergence, and requires convergence criteria to be set so that the code knows to stop when the solution has reached the desired accuracy.
1.4 Chapter 1 Summary

Chapter 1 gave a brief overview of computational fluid dynamics relevant to the work of this thesis. First there was a description of the governing equations of fluid dynamics, which lead on to how computers can be used to solve these equations using numerical methods. The use of numerical methods is necessary because finding analytical solutions to the governing equations is impossible in most cases. This has lead to the development of many different CFD solvers and a description was given of the CFD solver Code_Saturne. This solver will be used for the investigations conducted in this work in later chapters. The drawback in using numerical methods, which was not discussed in this chapter, is that errors and uncertainties are introduced into the calculations. The mechanism of error and uncertainty generation when using these numerical methods will be discussed in the following chapter. This introduces the concepts of error identification and reduction, leading on to a description of some of the different methods that have been used in the past to produce error estimates.
Chapter 2

Error Estimation Background and Theory

Chapter 2 aims to provide a number of useful definitions and provides a framework to demonstrate the aims of this research. First the difference between an error and an uncertainty is defined with respect to CFD. The difference is simply whether there is a deficiency in the modelling process that is due to a lack of knowledge or not. A description of the most common sources of errors and uncertainties is given with spatial and temporal errors being the most common and significant. Understanding their sources leads onto their identification and reduction in CFD simulations. The terms verification and validation are defined which informally ask whether the equations have been solved correctly and whether the correct equations have been solved respectively. Finally the most common current error determination methods are described. They range from very precise computationally expensive methods to less expensive methods that are less mathematically rigorous.

Spatial numerical errors and their verification are the main focus of this work, developing an error estimation method that is relatively computationally cheap and has a mathematical basis.
2.1 Error and Uncertainty in CFD Calculations

There are a number of sources of errors and uncertainties that can cause CFD simulations to differ from their true or exact values. These not only apply to the CFD code, but other computer programs used in the analysis process such as CAD packages, grid generators, and flow visualizers. Sources of errors and uncertainties and ways to reduce them are discussed here drawing upon references [12, 45 & 49]. First the difference between them in the context of CFD is defined. Error and uncertainty are commonly used by the experimentalist to mean “the difference between the measured value and the exact value” and “the estimate of error” respectively. However, in most computational simulations the exact value is typically not known and so these definitions would be inadequate. More suitable definitions are outlined below, they importantly differentiate the two terms by what is known and what is not known [20][38].

2.1.1 Definition of Uncertainty

An uncertainty is a potential deficiency in any phase or activity of the modelling process that is due to the lack of knowledge, indicating that deficiencies may or may not exist [3]. This would mostly be the result of a lack of knowledge about the physical processes that go into building a model. For example, the uncertainties that arise from deficiencies in turbulence modelling due to areas of turbulence modelling that are not yet understood [35].

2.1.2 Definition of Error

An error is a recognisable deficiency in any phase or activity of modelling and simulation that is not due to lack of knowledge, implying that the deficiency is identifiable upon examination [3][35]. Errors can also be classified as acknowledged or unacknowledged:

- Acknowledged errors have procedures for identifying them and so may be removed or left to remain in the code with their error estimated and listed. Examples would be physical approximation error, computer round-off error, iterative convergence error and discretisation error.

- Unacknowledged errors have no set procedures for finding them and may continue within the code or simulation and are ‘unknown unknowns’ about which very little can be done other than significantly increase safety margins. Examples include computer programming error and usage error.

2.1.3 Sources of Error and their Reduction

Thanks to the aid of computers the ability to undertake more and more complicated CFD problems has advanced faster than the standards and quality in the running of CFD simulations. Extensive consultation with European industry revealed an urgent demand for guidelines on quality and standards of CFD. The Best Practice Guidelines (BPG) [12] were commissioned to meet the new requirements set by the European Research Community on Flow, Turbulence and Combustion (ERCOFTAC) and to provide a basis for quality management and regulation of safety submissions which rely on CFD. To
do this, input and advice was sought from a wide cross-section of CFD specialists, eminent academics, end-users and the leading commercial code vendors established in Europe. The BPG can therefore be considered to represent the consensus view of the European CFD community. The first edition was completed in January 2000 and provides advice on several specific topics on how to carry out quality CFD calculations. These guidelines include comments on: mesh design, construction of numerical boundary conditions where problem data is uncertain, mesh and model sensitivity checks, distinction between numerical and turbulence model inadequacy, preliminary information regarding the limitations of turbulence models, etc. With reference to their findings, the five main sources of errors and uncertainties are discussed now and suggestions are made on how to systematically reduce them where possible.

2.1.3.1 Numerical Errors

Numerical errors result from the differences between the solutions of the exact equations and the discretised equations solved by the CFD code. Examples of numerical errors include:

- Spatial or temporal discretisation errors – errors that arise from replacing the analytical derivatives or integrals in the exact governing equations by numerical approximations, which have a certain truncation error. Defined as the difference between the exact solution to the discretised equations and analytical solution to the un-discretised equations [44]. For consistent discretisation schemes, these errors can be evaluated by grid convergence studies where successively finer grids are used, and would be expected to reduce as the spatial grid density increased. The same can be done with the size of the time step where the errors are expected to reduce with smaller time steps.

- Iteration error – errors that result from the premature ending of a simulation of a flow problem using an iterative method before the solution has converged. To reduce this error, the convergence criteria can be decreased or the simulation time lengthened.

- Round-off errors – errors resulting from the fact that a computer only solves the equations with a finite number of digits and the computer cannot differentiate between numbers that are different by an amount below the available accuracy. Including the iteration errors, these are defined as the difference between the exact solution to the discretised equations and computer solution [44]. These errors can be identified by grid convergence studies, but often they are simply shown to be insignificant. Double precision values are preferred, if the computer power is available, to reduce these errors.

Spatial and temporal discretisation errors are the most significant of the three. Numerical errors in general are the main source of error in CFD and the focus of this work will be identification of errors of these types.

2.1.3.2 Modelling Uncertainties

To avoid the excessive computing requirements needed to solve the exact governing equations directly, a DNS approach, it is necessary to describe flow phenomena like turbulence, combustion, and multi-
phase flows by empirical models to bridge the gap between the real flow and the statistically averaged equations. This will reduce the resolution requirements by several orders of magnitude [38].

For turbulence the classical model is based on time or ensemble averaging of the equations resulting in the so-called RANS equations. Due to the averaging procedure, information from the full Navier-Stokes equations is lost and this must be compensated for by the turbulence model which will introduce inaccuracies called modelling errors. The accuracy of the simulation cannot be increased beyond the capabilities of the model and they require the calibration of model coefficients for certain classes of flows. Here, calibration is defined as the process of adjusting numerical or physical modelling parameters in the computational model for the purpose of improving agreement with experimental data [3][39]. There are a wide variety of models used for CFD simulations and the most widely used industrial models are two-equation models, like the $k - \omega$ or $k - \epsilon$ models, and a number of simulations should be run with a variety of turbulence models to see how the different models affect the results [39].

2.1.3.3 Application Uncertainties

Application uncertainties are related to insufficient information to define a CFD simulation. Examples are: insufficient information on the boundary conditions, insufficient information on the geometry, and uncertainty in experimental data for solution evaluation. Where possible simulations, lacking such information should be avoided as there is no guarantee results will match the physically correct solution.

2.1.3.4 User Errors

User errors result from misuse of the resources available for a CFD simulation and are usually a result of insufficient expertise and attention to detail by the CFD code user. Mistakes may include over-simplification, poor grid generation, incorrect boundary conditions, acceptance of non-converged solutions and post-processing errors. They can be reduced or avoided by additional training and experience in combination with a high quality project management and by provision and use of the BPG.

2.1.3.5 Code Errors

Code and software errors are the result of an inconsistency between the documented equations and the actual implementation in the CFD software. They can be the result of programming errors, errors in the graphical user interface (GUI), documentation errors or incorrect support information. These can be detected by using verification and validation methods such as grid convergence studies for a problem with an exact solution. Removal of such programming errors contributes towards code certification which is the process of establishing the credibility of a code and encompasses verification, validation (discussed in the next section) and calibration, as well as documentation, quality assurance, and version control.
2.1.4 Numerical Errors

Numerical errors and their estimation will be the main focus of this research. A more detailed discussion of the source of spatial and temporal numerical errors is given here to better frame the problem the research aims to improve. CFD methods normally estimate the spatial or temporal gradients in the governing equations by discretising them through the use of Taylor expansions. These gradients are represented as an infinite series, and then for practicality of computation they are truncated to a short finite series. The truncated terms are expected to be negligibly small, however, for simplicity of the discretisation in the solver, only very few terms might be included. The terms removed in this way create a truncation error and this information is lost from the simulation. This can have a noticeable effect especially if the gradients being estimated are changing quickly.

The choice of truncation point will lead to an estimate of the gradient that depends on a length scale; the power of this length scale in the leading truncated term is called the order of the discretisation scheme. To reduce the size of these errors the order of the scheme should be increased by inclusion of more terms from the Taylor expansion, and the length scale should be reduced which captures more of the changing gradients. However, at times it will not be possible to do one or both of these and so simulations will contain non-negligible errors. This is one major source of error in CFD which is unavoidable. Therefore methods to predict location and size of errors are required for trustworthy CFD [25]. When errors are discussed in this work, they will in general refer to numerical errors unless stated otherwise.
2.2 Verification and Validation

There are two possible ways for a simulation to produce inaccurate results in the solving of a fluid dynamics problem. The first is the use of inappropriate governing equations that do not match the physical system attempting to be described. This will result in a solution containing uncertainties, and the solution may not necessarily resemble the correct one. The second is through inaccurate numerical solution procedure where the set of equations that describe the system are not solved to a high enough temporal or spatial accuracy, or convergence, etc [9]. Demonstrating during a CFD simulation that the errors and uncertainties from these sources are minimised is termed Verification and Validation respectively. These two concepts were first defined by Blottner before adoption by the American Institute of Aeronautics and Astronautics (AIAA). The development of methods to do this is a necessary step towards the quality of simulations now desired by the CFD community. These two processes are defined in greater detail in this section drawing upon references [2, 3, 12, 45 & 50].

2.2.1 Verification

Verification informally asks if the equations are being solved correctly. It is the process of determining that a model implementation accurately represents the developer’s conceptual description of the model and the solution to the model. The strategy of verification is the identification, quantification, and reduction of errors in the computational model implementation and its solution, to remove any errors in the code and to determine if the resulting code can be properly used for analysis. Verification activities are performed early in the development and certification of a computational code but must be repeated if the code is modified or enhanced. The two aspects of verification are the verification of a code and the verification of a calculation.

2.2.1.1 Verification of a Code

This involves error evaluation, which looks for bugs, incorrect implementations of conceptual models, errors in inputs, and other errors in the code and usage. Identifying and quantifying each type of error is important because errors can interact and cancel each other, leading to erroneous conclusions in the validation process. Verification of a code involves consistency checks which examine basic relationships expected in the solutions (i.e. mass conservation) and using the code to simulate highly accurate verification cases which should be analytic or numeric solutions to ordinary and partial differential equations, not comparisons with experimental data. One potentially useful method of code verification is comparing the results of two codes.

2.2.1.2 Verification of a Calculation

The objective of verifying a calculation is error estimation, which concerns determining the accuracy of a single calculation and putting an error band on the final value. This would involve performing a grid convergence study to determine the observed order of convergence, error bands, and grid convergence indices.
2.2.1.3 Verification Assessment Process

The general view now is that verification testing of computer codes must be held to a high standard to ensure reliability. The steps required for a reliable verification assessment of a CFD code and/or simulation can be summarised as:

- Examine the code - undertake a review of the computer programming and coding to check for and identify computer programming errors or “bugs” by visually checking the coding, as well as performing numerical tests and running subprograms using a test code.

- Examine iterative convergence - verification assessment requires that a simulation demonstrates iterative convergence where the residual is shown to be below a certain limit so that it does not mask errors from other sources.

- Examine consistency - check for consistency in the CFD solution for example mass conservation.

- Examine spatial convergence - a grid refinement study should be conducted, where the simulation is performed on two or more successively finer grids. This intends to bring out potential errors, to determine the order of the discretisation error in a CFD simulation, and to show that the CFD simulation results demonstrate spatial convergence where the difference between the solutions from one grid compared to the next more refined grid is below a desired value. From these results a higher spatial order solution can be estimated via methods such as Richardson extrapolation.

- Examine temporal convergence - time-accurate simulations involve taking discrete time steps whereby the sensitivity of the simulation results in variation of the magnitude of the time step must be investigated.

- Compare CFD results to highly accurate solutions - the veracity of a code can be examined by comparing the CFD simulation results to highly accurate solutions of the models used within the CFD code. This can include analytical solutions and benchmark numerical solutions to partial differential equations.

2.2.2 Validation

Validation informally asks if the correct equations are being solved for the particular fluid flow problem. It is the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model. The strategy of validation consists of comparisons of the results from a simulation, once verified to quantify bias and random errors from other sources, with real world observations and is a method to identify and quantify error and uncertainty. It is therefore a test of the particular conceptual models, computational models as implemented into the CFD code, and computational simulation for a certain type of flow problem. It is not possible to validate the entire CFD code, only validate the code for a specific range of applications for which there is experimental data.
2.2.2.1 Validation Assessment Process

After verification exercises have been completed the approach to validation assessment is to perform a systematic comparison of CFD simulation results to experimental data. The process for Validation Assessment of a CFD simulation can be summarised as:

- Verification exercises as before - before validation, verification must be completed and so first iterative convergence, consistency, spatial convergence and temporal convergence are examined as before.

- Compare CFD results to experimental data - experimental data are the real world observations mentioned earlier and good agreement with the CFD results increases confidence that the physical models and the code represents the ‘real world’ for this class of simulations. The experimental data will contain some level of error, which is related to the complexity of the experiment. A ‘building block’ approach is recommended where a hierarchy of experiment complexity is used consisting of phases of successively more complex flow physics, geometry, and interaction. Plots comparing CFD results and experimental data should include a visual display of the error bars on the experimental data.

- Examine model uncertainties - the physical models of flow phenomena in the CFD code contain uncertainties due to a lack of complete understanding or knowledge of the physical processes. Some of the models with the most uncertainty are the ones for turbulence modelling. The uncertainty can be monitored by running a number of simulations with multiple turbulence models that are applicable to the model problem and examining the effect on the results.
2.3 Current Error Determination Methods

The main focus of the research is estimating the numerical errors on simulation results. From the definitions before, this falls under verification since we are asking how well the defined equations have been solved rather than if they are the right ones. This section will go through some of the current methods used to identify and quantify calculation errors, and the ones described will deal with verification of calculations only. The methods assume that a rigorously verified code is used on a well behaved problem. Before continuing, a distinction should be made between two error determination methods: error indicators and error estimators.

A combination of gradients of flow variables that is used, for example, in adaptive refinement is called an error indicator. These do not reveal information about the size of the errors but can highlight regions where better resolution is required. The indicators are cheap to compute, however their use is limited to flows that have discontinuities, adaptive refinement for smooth flows cannot be controlled by an error indicator [25].

An error estimate is expected to give information about both the shape and size of the errors and is an improvement over the error indicator. An error estimate will be more expensive to compute, but offer more information about the errors. The goal of the work in this report is to produce an error estimate that is cheap to compute. The quality of an error estimate can be measured by the effectivity index, which is the ratio of the estimated error with the exact error. Good estimations produce indices close to one [25].

The focus of the research was to produce an error estimate, something that will predict both the size and location of the errors. Here methods to create both types are described to enrich the background of the literature review. There are a few different ways to go about determining the simulation errors, they can be found either systematically or non-systematically. Ways to systematically calculate them require changing the calculation to decrease (or increase) one or more of the sources of error discussed earlier. This way the solution can be observed to converge and the error identified. The process could therefore be time or computer power expensive. Examples of systematic methods discussed in this section include:

- additional solutions of the governing equations on other grids: grid refinement, grid coarsening, other unrelated grids. These involve direct unambiguous evaluation of any error measure of engineering or scientific interest. Only grid refinement provides rigorous verification because it can be continued to any desired level of accuracy, within the limits of the solver and modelling applicability.

- additional solutions of the governing equations on the same grid: higher order accuracy solutions (or lower order accuracy solutions) which are more easily incorporated into codes.

Non-systematic methods will use the CFD solution, complete or incomplete, to attempt to directly calculate the error. Examples include:

- auxiliary PDE solutions on the same grid. This does not simply involve a local evaluation of the error. The key aspect here is that errors are transported, advected and diffused, etc.
• auxiliary algebraic evaluations on the same grid. These are relatively cheap, need no additional
grid generation, and use no significant dynamic memory. However, the error evaluated usually
has no direct relation to any error measure of engineering or scientific interest.

A discussion of these classic methods is given below, as well as some recent attempts to create error
estimates that are quick and simple to compute that lead on to the current research.

2.3.1 Grid Refinement using Multiple Grids

Grid refinement represents the most reliable method for dependable quantification of numerical uncer-
tainty for all quantities. For error evaluation it involves a straightforward refinement (or coarsening)
of the discretisation grid. For boundary-fitted structured grids, the simplest method is to generate the
finest grid first, and then obtain the coarser grids by removing every other point [45]. Once two or more
completely solved solutions have been obtained using coarse and fine grids, the coarse grid solution is
used to estimate the error of the fine grid solution by comparison. For example if the convergence rate
\( p \) is known, then a technique called Richardson extrapolation can be used to quantify the error with
systematically refined grids.

Grid refinement will be easier to implement with a CFD solver than other systematic methods
because normally it will be impractical to use, for example, a higher order scheme. As a result grid
refinement is the most commonly used technique [45]. The disadvantage to the technique is that the
generation of multiple grids can quickly lead to time and power expensive simulations. For the sake of
efficient use of resources the step size should be selected so that a grid independent converged solution
can be found while running efficiently. Over refining the grid rapidly leads to very computer power
expensive simulations, especially in 3D.

Another use for grid refinement is to ensure confidence that the errors based on the step size are
negligibly small. The process of discerning when the solution has become grid independent is termed
a grid refinement study. In one of these studies, solutions to the flow problem are found on several
different grids with increasing refinement. Convergence criteria are monitored to judge when the
solution is no longer changing with refinement and hence the solution is grid independent [14]. More
detailed descriptions can be found in [43]. A generalisation of the method has been tested in [17].

2.3.1.1 Richardson Extrapolation

Richardson extrapolation is a sequence acceleration method, used to improve the rate of convergence
of a sequence [42]. It extends the grid refinement process and is used on a variety of situations across
CFD. It has been an often used error estimation method in finite volume CFD for some time and on
many different types of flows [7][54]. The process uses two solutions to the problem found with two
meshes with different grid spacings.

To outline the process consider a numerical method that depends on a step size \( h \) which is used to
produce an approximation, \( A(h) \), to a desired result \( A \). The order of the error generated in finding
the approximation is known. This situation can be written as

\[
A = A(h) + mh^p + O(h^{p+1})
\]  

(2.3.1)
where \( p \) is a known constant and is the order of the scheme used, \( m \) is a constant, and higher order \( O(h^{p+1}) \) terms can be excluded [14].

If this is now repeated using a step size of \( 2h \) a similar result is reached and it leads to

\[
A = A(h) + mh^p + O(h^{p+1}) = A(2h) + m(2h)^p + O((2h)^{p+1})
\]

(2.3.2)

It follows on from here that the error on the mesh with spacing \( h \) is approximately

\[
mh^p \approx \frac{A(h) - A(2h)}{2^p - 1}
\]

(2.3.3)

The ratios of the grid sizes does not need to be 2 and further information on this can be found in [43]. As an example of Richardson extrapolation in use, one widely used numerical integration algorithm, called Romberg integration, applies this formula repeatedly to the trapezoidal rule [18].

### 2.3.2 Higher Order Accuracy Solutions on the Same Grid.

A higher order accuracy solution is an additional solution found using a higher order discretisation of the governing equations. Increasing the order of the discretisation means including more terms from the Taylor expansion, which therefore decreases the numerical error. It is analogous to grid refinement, where instead of varying the step size, it is the truncation point in the discretisation scheme that is varied. Methods based on this estimate the errors through comparison of low order accuracy solutions with higher order ones [42][44][45].

The advantages of the method are that it can be used for any solved quantity and that multiple grid generation is not required. A major disadvantage of using higher order discretisation schemes is that they require additional code capability, which may not be practical. In the finite volume method, an increase in order means that the number of cells included in the gradient calculation increases. For example in a second order scheme the calculation around a certain cell includes its nearest neighbours. In a fourth order scheme the next nearest neighbours would also have to be included, which in 3D would create a very large discretisation network. This would continue to quickly increase becoming computationally expensive to calculate and solve, and each scheme would need to be coded separately. For this reason the user may not use increasingly higher ordered schemes, in the same way that one could continue to refine the grid, in order to study the convergence. As a result the technique is not widely used. Grid refinement and higher order solution techniques therefore share opposing benefits and drawbacks.

The higher order schemes may not be coded in the CFD software due to the excessive computation required to solve the resulting discretised equations. However, the effort required for coding can be made more attractive if they are only intended to be used for error estimation rather than finding full high order solutions. This is because error estimation of a second order solution using fourth order methods is less demanding numerically than obtaining a fourth order solution for direct use.
2.3.2.1 Example Use of Higher Order solutions

As noted by Richardson [42] the difference between the solutions found using a second and fourth order scheme is an ordered error estimator. To show this consider the exact solution $f_e$, a second order solution $f_2$ and a fourth order solution $f_4$. The exact solution in 1D is then given by

$$f_e = f_2 + C_2 \Delta x^2 + R_2$$

(2.3.4)

as well as

$$f_e = f_4 + C_4 \Delta x^4 + R_4$$

(2.3.5)

where $C_2$ and $C_4$ are the coefficients of the Taylor’s theorem expansion for the second and fourth order solutions, and $R_2$ and $R_4$ are the remaining terms in the complete second and fourth order expansions. Defining the error of the second order solution $\delta_2$ to be

$$\delta_2 = f_2 - f_e$$

(2.3.6)

and substituting for $f_e$ from Equation 2.3.5 we obtain

$$\delta_2 = f_2 - f_4 + O (\Delta^4)$$

(2.3.7)

This result shows that the difference between the second and fourth order solutions on the same grid is a fourth order error estimate for the second order solution. Similarly, if a fourth order solution error $\delta_4$ is defined as

$$\delta_4 = f_4 - f_e$$

(2.3.8)

then substituting for $f_e$ from Equation 2.3.4 we obtain

$$\delta_4 = f_4 - f_2 + O (\Delta^2)$$

(2.3.9)

which is a second order error estimator for the fourth order solution.

2.3.3 Auxiliary PDE solutions on the Same Grid

This technique relies on the fact that errors are transported by convection and diffusion, etc. similar to other quantities, and methods based on this approach to directly calculate the error. Beginning with Equation 1.1.16, the scalar transport equation, in steady state, and with the distinction made that $f_e$ is the exact solution

$$u \cdot \nabla f_e - \nabla (\beta \nabla f_e) = S$$

(2.3.10)

where $u$ is the advection velocity, $f_e$ is the concentration of the transported scalar, $\beta$ is the diffusivity, and $S$ is the source of $f_e$ per unit mass [45][51][56]. This can be contracted by representing the entire partial differential equation operator in a simplified form $L$ so that

$$L (f_e) = S$$

(2.3.11)
The process of discretisation will change \( L \) to the discretised form \( L_h \), and so the numerical solution, \( f_h \), found in the solving process will actually be a solution to the equation

\[
L_h (f_h) = S \quad (2.3.12)
\]

\( f_h \), which currently is defined only at nodal points, will not be a solution to the original operator \( L \) and will not be equal to \( f_e \). This leads to defining the error function as \( \delta = f_h - f_e \). As long as \( f_h \) represents a continuous function that is defined in the operator domain of \( L \), i.e. it must possess at least second degree derivatives and the same number of dimensions, we can define the operator residual as

\[
R(f_h) = L(f_h) - S \quad (2.3.13)
\]

which assumes we were able to extend the defined-at-nodes function \( f_h \) to a continuous function.

Once we have defined \( R(f_h) \), a substitution can be made for \( f_h \) in \( L \) using the error function definition giving

\[
R(f_h) = L(f_e + \delta) - S \quad (2.3.14)
\]

which separates into

\[
R(f_h) = L(f_e) + L(\delta) - S \quad (2.3.15)
\]

The terms including \( f_e \) will cancel with \( S \) by definition and what remains is the auxiliary error equation

\[
L(\delta) = R(f_h) \quad (2.3.16)
\]

whose solution gives the error, \( \delta \), exactly. The numerical solution of this equation provides the auxiliary PDE solution on the same grid. This treats the global error distribution as a scalar governed by a transport equation that has a source term \( R \).

### 2.3.3.1 Estimating Errors Using Higher Order Schemes

As mentioned earlier, the use of higher order schemes is not usually possible due to the computing power or coding needed. However, they can be utilised along with the auxiliary error equation for error estimation of the low order solution errors. A solution that satisfies a low order discretisation will not satisfy the higher order equation. This is similar to before where \( f_h \) did not satisfy the un-discretised equation leaving residual \( R(f_h) \). This means a low order solution could be substituted into a high order discretisation thereby approximating \( R(f_h) \). For example if we have a second order solution, \( f_2 \), and a fourth order discretisation scheme available, \( L_4 \), then the approximation of the residual is

\[
R_4(f_2) = L_4(f_2) - S \quad (2.3.17)
\]

If we use the approximation \( R(f_2) \approx R_4(f_2) \), then this can be substituted into Equation 2.3.16 to give

\[
L(\delta) \approx R_4(f_2) \quad (2.3.18)
\]
The practical application would be to solve this now complete equation using the second order scheme, thereby approximating the error distribution. As long as the higher order scheme is available for use then an error estimate can be found at the cost of running an additional low order simulation, which is relatively cheap. The drawback is that this second simulation would need to occur after the low order solution had been found before it can be used to calculate $R$. Then a second simulation would be run which means the error analysis would come at the expense of additional time.

2.3.4 Auxiliary Algebraic Evaluations on the Same Grid

These methods are relatively cheap to compute, require no additional grid generation or discretisation schemes. They will typically use the solution itself to calculate an error estimate based on some process shown to produce a good match. Therefore the methods are not truly attempting to calculate the error directly like in the previous sections. This means that the error estimate found has no direct relation to any error measurement of scientific interest.

2.3.4.1 Estimating the Leading Truncated Term from the Taylor Series

As discussed before, a $m^{th}$ order accurate discretisation method describes the local variation of a variable with the first $m$ terms of the Taylor series. The discretisation process truncates the infinite series of derivatives this creates. The discretisation error can therefore be expressed as the infinite sum of higher order derivatives of $f$ that were truncated.

To give a brief description, consider the first four terms of a Taylor expansion in space of a function $f$ around $P$

\[
    f(x) = f_P + (x - x_P) \cdot (\nabla f)_P + \frac{1}{2} (x - x_P)^2 : (\nabla \nabla f)_P + \frac{1}{3!} (x - x_P)^3 \cdot : (\nabla \nabla \nabla f)_P + \cdots
\]

\[
    + \frac{1}{n!} (x - x_P)^n \cdot : \cdots : (\nabla \nabla \cdots \nabla f)_P + \cdots
\]  

(2.3.19)

where $(x - x_P)^n$ is the $n^{th}$ tensorial product of the vector $x - x_P$ with itself and is rank $n$ [25]. The operator $\cdot : \cdots :$ represents the $n^{th}$ inner product between two rank $n$ tensors and produces a scalar.

If the discretisation scheme is $m^{th}$ order it includes the first $m$ terms of Equation 2.3.19 and so the truncation error will be

\[
    \delta(x) = \sum_{n=m}^{\infty} \frac{1}{n!} (x - x_P)^n \cdot : \cdots : (\nabla \nabla \cdots \nabla f)_P
\]  

(2.3.20)

Assuming $f$ is smooth and the control volume is small successive terms in the truncation error become increasingly small. The leading term will be a close approximation to the error. Using the first term where $n = m$, and integrating over the control volume, the error estimate after some manipulation becomes

\[
    \text{estimate} = \frac{1}{V_P} \frac{1}{m!} \left| \int_V (x - x_P)^m dV \cdot : \cdots : (\nabla \nabla \cdots \nabla f)_P \right|
\]  

(2.3.21)
Methods such as grid refinement and using higher order schemes are attempting to systematically calculate this. However, a recent effort by Jasak [26] attempts to estimate the leading $m^{th}$ order term in the truncation from the $m^{th}$ order solution and mesh geometry alone. This estimate is very quick and cheap to compute. The method calculates the $\int_V (x - x_P)^m dV$ term according to the geometry and a more detailed description of this can be found in Helf and Kuster [28]. Then calculation of the $\left( \nabla \nabla \cdots \nabla f \right)_P$ term is done via multiple application of Gauss theorem. As an example, the result for a second order discretisation scheme on a square regular mesh with cell size $h$ will be

$$\text{estimate} = \frac{1}{24} h^2 \sqrt{(\nabla \nabla f)_P : (\nabla \nabla f)_P} \quad (2.3.22)$$

The advantage of the method is that it uses only a single solution. Even so, it is considered to be quite expensive because it requires a product of two second rank tensors. It typically underestimates the error as a result of the double Gauss theorem used to calculate the second gradient which produces the average value of the gradient over the nearest neighbours, rather than the required cell-centre value [26].

2.3.4.2 Second Moment Residual Method

This method makes use of a property of the exact solution, $f_e$, where the exact solution of the original transport equation for $f$ will also satisfy all higher moments of $f$. A detailed explanation of what this means will be given in the next chapter because it is crucial to the research. For now the results are simply stated. As an example consider the first moment for $f$

$$u \cdot \nabla f - \beta \nabla^2 f = S \quad (2.3.23)$$

Through algebraic manipulation, which will be discussed later, we can arrive at the second moment for $f$

$$u \cdot \nabla \frac{f^2}{2} - \beta \nabla^2 \frac{f^2}{2} = S f - \beta \nabla f \cdot \nabla f \quad (2.3.24)$$

which is a transport equation for $\frac{f^2}{2}$. So the exact solution will satisfy both of these equations. However, through the CFD process we will find the numerical solution $f_h$, which is a solution of the discretised equation and will not satisfy the second moment. If $f_h$ is substituted into Equation 2.3.24 and integrated over the control volume it will not balance, and will leave a residual $r_p$

$$r_p = \int_{V_p} \left[ u \cdot \nabla \frac{f_h^2}{2} - \beta \nabla^2 \frac{f_h^2}{2} - S f_h + \beta \nabla f_h \cdot \nabla f_h \right] dV \quad (2.3.25)$$

In [25] and [26] this residual was rescaled by an appropriate length scale to calculate an estimate which was used for automatic mesh refinement. The rescaling is based on the local transport conditions

$$u_t = |u| + \frac{\beta}{h} \quad (2.3.26)$$
where $h$ is the local mesh size. The characteristic time scale $T$ is then

$$T = \frac{\dot{h}}{u_t} = \frac{h^2}{|u| h + \beta}$$  \hspace{1cm} (2.3.27)

Combining Equations 2.3.25 and 2.3.27 gives the final form of the estimate with the same dimensions as $f$ as

$$2 \sqrt{\frac{|r_p| T}{V_p}}$$  \hspace{1cm} (2.3.28)

It is found from a single simulation on one grid and is therefore a very quick and cheap to compute estimate. It has been reported to perform better on coarser meshes and it works best in flows that contain high gradients such as with shock waves. Therefore in general CFD simulations with gently changing gradients the method is less applicable, despite its low computational requirements.
2.4 Chapter 2 Summary

Chapter 2 gave a brief overview of the types of errors and uncertainties that appear in CFD simulations and describes some of the methods available for their reduction. It described several sources of errors, with numerical errors identified as the most significant. In the following work they will be the main focus. The chapter gave a description of the commonly used methods for the determination of the errors. Each have their strengths and weaknesses when considering computational time and mathematical justification. A conclusion drawn is that a gap is available for a method that strikes a balance between these two. The intention of the work in this thesis is to attempt to fill this gap. The next chapter introduces the proposed error estimation method.
Chapter 3

Proposed Estimation Method: Second Moment Solution Estimate Method

Recall the second moment equation (SME) which was used in section 2.3.4.2 to calculate a residual that became an error estimate. While the first moment equation represents the transport of the primary variable, the second moment equation represents the transport of the primary variable squared. For example, a primary variable could be the velocity, and the squared variable would be the total kinetic energy, $K = \frac{u \cdot u}{2}$. The proposed method will take this a step further by attempting to obtain a solution to the second moment equation at the same time as the main calculation. The intention was to investigate whether additional information can be found using the squared-variable solution of this second equation (i.e. by comparisons with the unsquared-variable solution). The inspiration for this comes from the calculation of the variance

$$
\text{Var}(x) = \langle x \rangle^2 - \langle x^2 \rangle
$$

where variations between the two are revealed by comparing the square of the mean of $x$ and the mean of $x^2$. Through the squaring process the differences were exaggerated so that the two components are not identical and a measure of the deviation is found. Similarly this effect is expected when solving the second moment equation for the squared variables.

Chapter 3 will first outline the creation of the SME, starting from the governing equations of CFD for both the scalar and vector forms. In later sections Code_Saturne will be used to solve these equations for a range of fluid flow problems to get solutions to the first and second moments. Next in chapter 3 there is a discussion of how these solutions will be used to create an error estimate for both scalar and momentum results. There is then a discussion about how the estimate depends on the real errors of the first and second moment solutions.

Finally, two techniques are introduced that judge the accuracy of the error estimate. These are the summation error coefficients that judge the accuracy of the scale prediction, and the correlation coefficient that judges how well the shape distribution is predicted.
3.1 Creation and Solution of the Second Moment Equation

3.1.1 Governing Equations of Motion in Fluid (First Moments)

As described earlier, the motion within fluid, either the motion of the fluid itself or some scalar quantity, is governed by a set of equations. When applied to the velocity in steady state the resulting equations are the Navier-Stokes equations

\[ \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla P}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{F} \]
\[ \nabla \cdot \mathbf{u} = 0 \]  \hfill (3.1.1)

which are a set of four equations for the four unknowns. This equation will be referred to as the velocity first moment.

The transport of a passive scalar \( f \) in steady state by a fluid can be described by equations that represent the balance of convection and diffusion processes as well as any other sources. In steady state and in Cartesian, the scalar transport equation is

\[ \mathbf{u} \cdot \nabla f - \beta \nabla^2 f = S \]  \hfill (3.1.2)

where \( S \) is any additional source term. This equation is the scalar first moment.

3.1.2 Manipulation of the Scalar Transport Equation into the Second Moment Equation

Starting with the simpler case of scalar transport, the transformation into the second moment begins by multiplying Equation 3.1.2 by \( f \) to give

\[ \mathbf{u} \cdot f \nabla f - \beta f \nabla^2 f = fS \]  \hfill (3.1.3)

and, when rearranged, it produces a similar equation

\[ \mathbf{u} \cdot \nabla q - \beta \nabla^2 q = fS - \beta \nabla f \cdot \nabla f \]  \hfill (3.1.4)

this time representing the transport of \( q \left( = \frac{f^2}{2} \right) \), where the relations \( f \nabla f = \nabla \frac{f^2}{2} \) and \( f \nabla^2 f = \nabla^2 \frac{f^2}{2} - \nabla f \cdot \nabla f \) have been used [26].

From the point of view of a numerical solver, finding the solution to Equation 3.1.4 is exactly the same process as solving Equation 3.1.2. The differences are that \( f \) in the LHS of Equation 3.1.2 has been replaced by the variable \( q \), and the addition of an extra source term \( fS - \beta \nabla f \cdot \nabla f \). This equation is termed the scalar second moment.
### 3.1.3 Manipulation of the Momentum Equation into the Second Moment

The momentum equation is a more complicated case than for the scalar. It begins by taking the scalar product of Equation 3.1.1 with the velocity $\mathbf{u}$ to get

$$
\mathbf{u} \cdot (\mathbf{u} \cdot \nabla \mathbf{u}) = -\frac{\mathbf{u} \cdot \nabla P}{\rho} + \nu \mathbf{u} \cdot \nabla^2 \mathbf{u} + \mathbf{u} \cdot \mathbf{F}
$$

(3.1.5)

Rearranging this produces

$$
\mathbf{u} \cdot \nabla K - \nu \nabla^2 K = -\frac{\mathbf{u} \cdot \nabla P}{\rho} - \nu \nabla u_i \cdot \nabla u_i + \mathbf{u} \cdot \mathbf{F}
$$

(3.1.6)

which is similar to the scalar second moment. This required $K = \frac{1}{2} \mathbf{u} \cdot \mathbf{u}$, and the use of $\nabla \cdot \mathbf{u} = 0$ to arrive at $\mathbf{u} \cdot (\mathbf{u} \cdot \nabla)\mathbf{u} = (\mathbf{u} \cdot \nabla)(\frac{1}{2} \mathbf{u} \cdot \mathbf{u})$ and $\mathbf{u} \cdot \nabla^2 \mathbf{u} = \nabla^2 (\frac{1}{2} \mathbf{u} \cdot \mathbf{u}) - \nu \nabla u_i \cdot \nabla u_i$ (If viscosity were set to zero with zero force term this equation gives Bernoulli’s theorem, $\frac{1}{2} \mathbf{u} \cdot \mathbf{u} + \frac{P}{\rho} = constant$) [26].

From the point of view of a solver this is no longer like Equation 3.1.1. It has become one for the transport of the total kinetic energy, $K$, which is a scalar with an additional source while the first moment was three equations, one for each vector component. This is the velocity second moment equation and can be solved as easily as the scalar second moment.

### 3.1.4 Solving the Second Moment Equations

The solving of the SMEs is not an issue for the solver because the equations for $q$ and $K$ are the same as any scalar transport equations. However there is one difference: the source terms on the RHS of the SMEs require information about the first moment exact solutions, $f_e$ and $\mathbf{u}_e$, which are unknown. The SMEs cannot be solved without these and so the next best option is to estimate them from the solver solutions $f$ and $\mathbf{u}$. To do this, while the first moments are being solved to find $f$ or $\mathbf{u}$, their current solutions are used to calculate and update the source terms in the SME simulation. This allows the source terms to be approximated and the SME can be solved using the CFD solver.

Two additional errors are introduced into the SME calculation in this approach beyond the numerical errors from the solver: first $f$ and $\mathbf{u}$ will have their own associated errors; secondly there are errors from estimating the gradients to calculate the source term. These will be transferred to the SME and will affect the solution.

### 3.1.5 Combining the Solutions of the First and Second Moments

The numerical solution of the first moment, $f$ will not be the same as the exact solution, $f_e$, because there will be errors caused by discretisation and other sources. Similarly the solution of the second moment, $q$, will be different from the exact solution, $q_e$. Similarly for $\mathbf{u}$ and $K$. Therefore the numerical solutions do not obey the relations that their exact counterparts do, i.e.

$$
q \neq \frac{f^2}{2}
$$

(3.1.7)
Any comparison of the two solutions will leave a residual. Comparisons that provide information of interest to the scalar transport investigation are

\[ q - \frac{f^2}{2} \] (3.1.8)

and

\[ f - \sqrt{2q} \] (3.1.9)

For the momentum transport the equivalent comparisons are

\[ K - \frac{|u|^2}{2} \] (3.1.10)

and

\[ |u| - \sqrt{2K} \] (3.1.11)

The first of the comparisons for the scalar or velocity errors have the units of \( f^2 \) or \( u^2 \). This is not appropriate as an estimate of the errors in this form because it must have the units \( f \) or \( u \). It will be shown later that this first set of combinations produces the best estimate of the shape. Simply taking the square root to produce the correct dimensions would distort this shape because regions of low value will be artificially raised in value relative to regions of high value. The second set of combination have the correct dimensions and would not require rescaling. However, it does not produce a good estimate of the shape. The next section will develop the mathematical understanding of these combinations and will justify their use to create an error estimate.
3.2 Creation of the Second Moment Solution Error Estimate

This section defines the errors produced by the first and second moment solutions, and uses the definitions to analyse the combinations presented in section 3.1.5. This allows the full error estimate to be defined, based on mathematical justifications. Finally two methods to judge the success of the error estimate are described.

3.2.1 Mathematical Analysis for Scalar Transport

The combinations presented above can be broken down to reveal what they represent in terms of the solution errors. First the simulation errors need to be defined. The errors are the difference between the numerical solution and the exact solution. Therefore the error on the \( f \) simulation is

\[
\delta = f - f_e
\]  

(3.2.1)

Using this to substitute for \( f \) in the additional source term in Equation 3.1.4 with no other sources, the scalar second moment equation becomes

\[
\mathbf{u} \cdot \nabla q - \beta \nabla^2 q = -\beta \nabla (f_e + \delta) \cdot \nabla (f_e + \delta)
\]  

(3.2.2)

Expanding the source term and grouping terms gives

\[
\mathbf{u} \cdot \nabla q - \beta \nabla^2 q = -[\beta \nabla f_e \cdot \nabla f_e] - [\beta \nabla \delta \cdot \nabla (2f - \delta)]
\]  

(3.2.3)

where the source term has been split into an exact part, which depends only on \( f_e \), and the discrepancy introduced by the inaccurate \( f \) solution. If the second term vanishes then the solving of the second moment equation would produce an estimation of \( q \) which would only have errors associated with the discretisation in finding the numerical solution. As it stands, the solving of this equation, with a source with a non-zero second term, produces a solution

\[
q = q_e + \varepsilon_d + \varepsilon_f
\]  

(3.2.4)

where \( \varepsilon_d \) is the unavoidable numerical errors caused by the solution process, and \( \varepsilon_f \) is the error introduced by the second term on the RHS of Equation 3.2.3 caused by an inexact \( f \) solution. Therefore high changes in the gradient of the \( f \) solution increase \( \varepsilon_f \) because the approximation of the gradient contains increased error. Equation 3.2.4 can be simplified to just

\[
q = q_e + \varepsilon
\]  

(3.2.5)

where \( \varepsilon \) represents all of the errors on the \( q \) solution.
3.2.1.1 Shape Estimate

Consider the combination $q - \frac{f^2}{2}$. This can be broken down into contributions from the two errors, $\delta$ and $\varepsilon$. Using Equations 3.2.1 and 3.2.4, $q - \frac{f^2}{2}$ becomes

$$q - \frac{f^2}{2} = q_e + \varepsilon - \frac{(f_e + \delta)^2}{2}$$

(3.2.6)

After expanding and grouping terms we get

$$q - \frac{f^2}{2} = q_e - \frac{f_e^2}{2} + \varepsilon - \delta f_e - \frac{\delta^2}{2}$$

(3.2.7)

Cancelling the two exact parts and substituting again with Equation 3.2.1, this becomes

$$q - \frac{f^2}{2} = \varepsilon - \delta \left( f - \frac{\delta}{2} \right)$$

(3.2.8)

This work is concerned with whether this is some representation of the errors, enough to identify regions of high or low error. Each of the three terms, $\varepsilon$, $\delta f$ and $\frac{\delta^2}{2}$, will not match the scale of the $f$ solution errors, $\delta$. They also have the units of $[f]^2$, while $\delta$ has the units of $[f]$. Therefore this combination would not be suitable for predicting the scale of the errors. However, each of the terms positively reinforce the prediction of the shape of $\delta$:

- $\frac{\delta^2}{2}$ will correctly contribute the locations of high and low error to the shape prediction.
- $\varepsilon$ will not match the shape of $\delta$, but it is expected to be in some way similar because of the similarity of the calculations.
- $\delta f$ will not match the shape of $\delta$ because the term will be skewed towards regions of high $f$.

This means that it will add greater importance to regions of high error and high $f$ compared to regions of high error and low $f$. However, regions of high error are expected to be found in regions of high $f$, and similarly regions of low error found in regions of low $f$. So, although there will be some slight distortion, this term will still reinforce the prediction of the shape.

Therefore $q - \frac{f^2}{2}$ will provide a shape estimate that will be able to identify the regions of high and low error.

3.2.1.2 Scale Estimate

Now consider the combination $f - \sqrt{2q}$. This can again be broken down into contributions of the two errors, $\delta$ and $\varepsilon$. Again using Equations 3.2.1 and 3.2.4, $f - \sqrt{2q}$ becomes

$$f - \sqrt{2q} = (f_e + \delta) - \sqrt{2(q_e + \varepsilon)}$$

(3.2.9)

Expanding the square root term gives

$$f - \sqrt{2q} = f_e + \delta - \sqrt{2q_e} \cdot \left( 1 + \frac{\varepsilon}{2q_e} \right)^{\frac{1}{2}}$$

(3.2.10)
which expands further using a Taylor expansion to give

\[ f - \sqrt{2q} \approx f_e + \delta - \sqrt{2q_e} \left[ 1 + \frac{1}{2} \frac{\varepsilon}{2q_e} - \frac{1}{8} \left( \frac{\varepsilon}{2q_e} \right)^2 + \ldots \right] \]  

(3.2.11)

A final expansion leads to

\[ f - \sqrt{2q} \approx f_e - \sqrt{2q_e} + \delta - \frac{1}{2} \frac{\varepsilon}{\sqrt{2q_e}} + \frac{1}{8} \sqrt{2q_e} \left( \frac{\varepsilon}{2q_e} \right)^2 - \ldots \]  

(3.2.12)

Again the exact terms are cancelled, a substitution \( \sqrt{2q_e} = f_e \) is made, and the higher order terms of \( \varepsilon \) excluded, to give

\[ f - \sqrt{2q} \approx \delta - \frac{\varepsilon}{2f_e} \]  

(3.2.13)

This combination breakdown directly contains the \( f \) simulation errors, \( \delta \), which contribute to it the correct scale and shape of \( \delta \). The second term, \( \frac{\varepsilon}{2f_e} \), will contribute poorly to the prediction of the shape. The reason is a result of \( f_e \) and \( \varepsilon \) being different shapes, with large parts of \( f_e \) being near zero in value in regions where non-zero errors may occur. The division by \( f_e \) will magnify these regions of non-zero \( \varepsilon \) in \( \frac{\varepsilon}{2f_e} \) relative to other regions (e.g. low \( \varepsilon \) with low \( f_e \)). Therefore the term incorrectly contributes a high prediction of error in regions where there should be very little, creating an incorrect shape prediction. This is unlike the previous analysis that included the term \( \delta f \). The difference is that \( \delta f \) contributes positively to regions of high and medium error, which is acceptable, while \( \frac{\varepsilon}{2f_e} \) contributes positively to regions where there may not be any error at all.

\( \frac{\varepsilon}{2f_e} \) will be the same approximate order of magnitude as \( \delta \), and so the approximate scale of this combination will be of the same order as \( \delta \). Therefore this can be used to predict the scale of the simulation errors to a value which is the within same order of magnitude.

### 3.2.2 Mathematical Analysis for Momentum Transport

The analysis for the momentum equation is more complicated than for the scalar. The first problem is how to define the velocity errors because there are two options: the definition for the velocity error could be chosen to either acknowledge that the error is a vector with three Cartesian components, or not.

The error could be defined as the velocity vector error, \( \chi \), as

\[ \chi = u - u_e \]  

(3.2.14)

whose magnitude can also be taken similar to finding the velocity magnitude. The error can also be defined as the velocity scalar error, \( \varsigma \), by comparing the magnitudes of the exact and numerical solutions as

\[ \varsigma = |u| - |u_e| \]  

(3.2.15)

The analysis process used for the scalar in section 3.2.1 can be carried out using both definitions of the velocity errors.

The \( K \) errors can be defined similarly to the \( q \) errors. The velocity first moment produces a solution
\( u = u_e + \chi \) and pressure field \( P = P_e + \Delta P \). Substituting these into Equation 3.1.6, with no force term, the velocity second moment becomes

\[
(u_e + \chi) \cdot \nabla K - \nu \nabla^2 K = -\frac{(u_e + \chi) \cdot (\nabla P_e + \Delta P)}{\rho} - \nu \nabla (u_{ei} + \chi_i) \cdot \nabla (u_{ei} + \chi_i)
\] (3.2.16)

When rearranged and the exact terms are separated from the non exact, we get

\[
u \cdot \nabla K - \nu \nabla^2 K = \left[ -\frac{u_e \cdot \nabla P_e}{\rho} - \nu \nabla u_{ei} \cdot \nabla u_{ei} \right] - \left[ \frac{\chi \cdot \nabla P_e}{\rho} + \frac{u \cdot \nabla (\Delta P)}{\rho} + \nu \nabla \chi_i \cdot \nabla (2u_i - \chi_i) \right]
\] (3.2.17)

The source term has been split into a term that depends only on \( u_e \) and \( P_e \), and the discrepancy introduced by the incorrect \( u \) and \( P \). If the second part were to be zero then this second moment equation would produce an estimation of \( K \) which would have numerical errors associated with solving for \( K \). As it stands, the solving of this equation, with a non-zero second source part, produces a solution

\[
K = K_e + \eta_d + \eta_u
\] (3.2.18)

where \( \eta_d \) is the unavoidable numerical error caused by the solution process, which also includes additional error from the inaccurate \( u \) solution and are again increased when large changes of gradient are present. \( \eta_u \) is the additional error introduced by the second term on the RHS of Equation 3.2.3 caused by the non exact \( u \) and \( P \) solutions. These additional sources of error will have a more significant impact on the \( K \) solution than was seen in the equivalent situation for the \( q \) solution. The \( K \) solution can be simplified to just

\[
K = K_e + \eta
\] (3.2.19)

where \( \eta \) represents all of the errors found on the \( K \) solution.

### 3.2.2.1 Shape Estimate

Consider the combination \( K - \frac{|u|^2}{2} \). This can be broken down into contributions from the \( u \) and \( K \) errors. Considering first the velocity scalar error, substituting Equations 3.2.15 and 3.2.19 into \( K - \frac{|u|^2}{2} \) gives

\[
K - \frac{|u|^2}{2} = K_e + \eta - \frac{(|u_e| + \varsigma)^2}{2}
\] (3.2.20)

which, after rearranging and cancelling the two exact parts similar to the earlier analysis, becomes

\[
K - \frac{|u|^2}{2} = \eta - \varsigma \left( |u| - \frac{\varsigma}{2} \right)
\] (3.2.21)

Alternatively, a similar result could have been found for the velocity vector error using Equation 3.2.14

\[
K - \frac{|u|^2}{2} = \eta - \chi \cdot \left( u - \frac{\chi}{2} \right)
\] (3.2.22)
This implies the relationship between the velocity vector error and velocity scalar error is

$$\chi \cdot (u - \frac{\chi}{2}) = \varsigma (|u| - \frac{\varsigma}{2})$$  \hspace{1cm} (3.2.23)

The result is similar to the equivalent seen with the scalar analysis. Through similar reasoning, this should be a good estimate for the shape of the velocity errors.

### 3.2.2.2 Scale Estimate

Now consider the combination $|u| - \sqrt{2K}$. The same process as before is used to break down the combination, substituting Equations 3.2.15 and 3.2.19 into $|u| - \sqrt{2K}$ gives

$$|u| - \sqrt{2K} = (|u_e| + \varsigma) - \sqrt{2(K + \eta)}$$ \hspace{1cm} (3.2.24)

Following the same method to expand $\sqrt{2(K + \eta)}$ seen in section 3.2.1.2, the result for the breakdown is

$$|u| - \sqrt{2K} \approx \varsigma - \frac{\eta}{2|u_e|}$$ \hspace{1cm} (3.2.25)

The result is similar to the equivalent seen with the scalar analysis. Through similar reasoning, this should be a good estimate for the scale of the velocity errors.

### 3.2.3 Error Estimate Definitions

The previous sections identified combinations of the first and second moment solutions that will provide meaningful information about the shape and scale of the numerical solution errors. This section proposes a method such that they can be utilised to form an error estimate. The suggested method is to rescale the better estimation of the shape using the better estimation of the scale. The shape estimate will be normalised by dividing it by its maximum absolute value. Then it is rescaled by multiplying it by the maximum absolute value of the scale estimate.

In this work both the errors and the estimates will be further normalised by expressing them as a percentage of a typical value of the simulations. This is to allow comparison of the results found on different test cases. The typical value used in this work will be the volume average of the numerical solution, i.e. the volume average of $f$ for the scalar error and error estimate, and the volume average of $|u|$ for the velocity error and error estimate. The volume averages are indicated by $\langle \rangle$.

#### 3.2.3.1 Scalar Error and Estimate

The scalar error is calculated using

$$\delta = \frac{|f - f_e|}{\langle |f| \rangle} \times 100\%$$ \hspace{1cm} (3.2.26)

where $f_e$ is approximated using a refined mesh solution or is the analytical solution.
The scalar error estimate, $\xi_s$, is calculated using

$$
\xi_s = \frac{|q - \frac{f^2}{2}| \cdot \max |f - \sqrt{2q}|}{\max |q - \frac{f^2}{2}|} \cdot \frac{100\%}{\langle |f| \rangle} \quad (3.2.27)
$$

### 3.2.3.2 Momentum Error and Estimate

The velocity scalar error is calculated using

$$
\zeta = \frac{|u| - |u_e|}{\langle |f| \rangle} \cdot \frac{100\%}{|u|} \quad (3.2.28)
$$

and the velocity vector error is calculated using

$$
|\chi| = \frac{|u - u_e|}{\langle |f| \rangle} \quad (3.2.29)
$$

where $u_e$ is approximated using a refined mesh solution or is the analytical solution.

The velocity error estimate, $\xi_v$, is calculated using

$$
\xi_v = \frac{|K - \frac{|u|^2}{2}| \cdot \max |u| - \sqrt{2K}}{\max |K - \frac{|u|^2}{2}|} \cdot \frac{100\%}{\langle |u| \rangle} \quad (3.2.30)
$$

### 3.2.4 Summation Coefficients: Averaged Quantities to Compare the Scale Estimate

The estimate may have superficial differences in shape that do not match the actual error, caused by the unusual combination of the $f$ and $q$ simulation. As a comparison between the scale of the error and the scale of the estimate a volume averaging of the values of each cell over both distributions was considered. The averaged result will provide a comparison between the scales of the error and the estimate unaffected by any local skewing. This would then produce a global error indicator, rather than a local error estimate.

Many of the cells in the following simulation results will be zero so a cut off was introduced. This allows a typical value of the results to be found which will better represent the level of error in the simulation. It was chosen to be 5% of the range created by the maximum and minimum error or estimate on the grid. Anything lower than 5% was excluded, i.e. for the quantity $a$ the limit will be

$$
\text{Lim}(a) = \min(a) + 0.05(\max(a) - \min(a)) \quad (3.2.31)
$$

The scalar error summation is calculated using

$$
\sum_{\delta_i > \text{Lim}(\delta)} \delta_i \Omega_i
$$

and similarly for the other error distributions. The estimate summations are calculated using
\[ \frac{\sum_{\xi > \text{Lim}(\xi)} \xi \Omega_i}{\sum_{\xi > \text{Lim}(\xi)} \Omega_i} \]  

(3.2.33)

### 3.2.5 Cross Correlation Coefficient to Compare the Shape Estimate

The estimation of the scale may be found to be very different to the scale of the errors, while the shape is predicted well. To compare the shapes of the error and estimate an image comparison technique was used. The chosen technique is the normalised cross correlation coefficient, \( N_{cc} \). It assumes there is a linear relationship between the two compared images rather than simple Gaussian noise. It provides a single number to assess the shape prediction, similar to the role the averaged quantities performed for the scale. The correlation coefficient between the error and the estimate distributions is defined as

\[ N_{cc} = \frac{1}{n} \sum \frac{(\delta_i - \bar{\delta}) \cdot (\xi_i - \bar{\xi})}{\sigma_{\delta} \sigma_{\xi}} \]  

(3.2.34)

where \( n \) is the number of cells, \( \bar{\delta} \) and \( \bar{\xi} \) are the average across the whole errors and estimates, \( \sigma_{\delta} \) and \( \sigma_{\xi} \) are the standard deviations of the error and the estimate. For the velocity error and estimate distributions the cross correlation coefficient is defined similarly. \( N_{cc} \) ranges from \(-1\) to \(1\), and the closer it is to \(1\) the closer the shapes are matched. Using this, the shapes can be compared without considering the scale [21][29].
3.3 Implementation in Code_Saturne

3.3.1 Simulation of the First Moment Equations

Finding the solutions to the first moment equations using Code_Saturne involves the simulation of the velocity or scalar according to the standard Code_Saturne procedure. Information about how to set up the standard simulations within Code_Saturne can be found in the comprehensive user manual in [15]. Simulations can be periodic across the inlet and outlet, or non-periodic. If the simulation is periodic the flow is driven by an additional forcing term to the Navier-Stokes equations for each cell in the fluid domain. This is applied in the user subroutine usrns by setting the coefficient crexp equal to the constant value of the force required to produce the desired Re. In a non-periodic simulation the inlet values are defined for each case. These are applied in the user subroutine usclim, the subroutine that controls the boundary conditions, by defining the cell face centre values across the inlet wall in the coefficient rodel.

In this work two wall boundary condition types are used for the first moment equations. These are either constant flux or constant value and they are selected using the coefficient icodel in usclim. For each boundary face, the value or flux is defined at the face centre by setting the coefficient rodel equal to the desired value or flux. The flux is defined as the flux perpendicular to the face.

In the user subroutine usini1 certain control coefficients were altered to control the order of the simulation. The coefficient blence was used to change the order of the simulation between first and second order. The coefficient ischec was altered to change the second order scheme used, i.e. to switch between using the SOC and SOLU schemes. The coefficient istspc allows the slope test to be deactivated so that the second order simulations could be purely second order, allowing testing of each scheme individually. It should be noted that this means the second order schemes are running without limiters. This risks unphysical solutions, such as those produced through unboundedness [37], which places a limitation on the maximum cell size.

3.3.2 Simulation of the Second Moment Equations

To simulate the velocity or second moment equations additional source terms require approximation and inclusion. This takes place during the simulation of the first moment equations using the current iteration's first moment solutions. The second moment equations are always scalar transport equations and so the additional sources are applied in the user subroutine ussce by setting the coefficient creexp equal to the source value for each cell. The source term is calculated by calling the subroutine grdcel with one of either the three velocity components’ solution or the scalar solution at a time to compute the gradient of that solution in each cell. The gradients are then used to calculate the additional source term for each cell according to Equations 3.1.6 and 3.1.4, and added to any other force or source terms.

The second moment equations must be simulated under the same boundary types as the main calculation and not require any additional code capability. Therefore the second moment boundary condition type will match the first moment calculation, and the cell face definitions follow a similar procedure as before. If, when using the constant value boundary condition, the boundary wall in the first moment simulation has a constant value \( m \), then the same boundary type is used for the second moment simulation and the value is equal to \( \frac{m^2}{2} \). When using the constant flux boundary condition the
wall flux will vary across the wall, and also vary as the simulation converges, because the boundary value will depend on the first moment solution. If the first moment, $f$, has a flux equal to $j$, the same boundary type is used for the second moment simulation and the flux is equal to $j \frac{\partial f}{\partial y'}$, where $y'$ is the normal to the wall.

The control coefficients used to control the order of the simulation will match the ones used in the first moment simulation.
3.4 Chapter 3 Summary

Chapter 3 outlined the proposed error estimation method and its implementation in *Code_Saturne*. The method makes use of the second moment equations, which are the first higher moments of the governing equations of CFD. First in this chapter there was a description of how the governing equations can be transformed into their second moments. These are able to be solved using *Code_Saturne* similar to the first moment equations, and the steps required to do so were outlined. The proposed error estimate uses the solutions from both the first and second moment equations. Next in the chapter the mathematical justifications for using the SME solution to create an error estimate were outlined. This allowed the second moment solution error estimate to be defined. The coefficients used to quantify the success of the error estimate, the summation and correlation coefficients, were also defined. Finally in this chapter, the technical details of how to implement the error estimation method in *Code_Saturne* were listed.

The thesis now moves on to the testing of the error estimate. The following chapters describe the test cases used in the work, and the results are presented and analysed using the quantifying coefficients. The next chapter introduces the three preliminary test cases, discusses the results, and draws preliminary conclusions, before moving on to the main test cases in later chapters.
Chapter 4

Test Cases: Preliminary Investigations

Chapter 4 describes and presents results for the very simplistic test cases used to initially test the error estimate. The first section will give descriptions of the three test cases along with the numerical or analytical solutions for $f$, $q$, $u$, and $K$. The second section presents the comparisons between the real error distributions with the error estimate distributions. The first test case is the 1D convection-diffusion equation. It was used to initially test the validity of the method and uses the finite element method, demonstrating that the methodology is not dependent on the numerical method used. The second test case is the point source in a cross flow and investigates the error estimate for scalar solution. This test case showed that *Code_Saturne* can be used to solve for the second moment and demonstrates that the error estimate is successful on a simplistic test case. The third test case is the Burggraf flow and investigates the error estimate for the velocity solution. It shows that the method shows some promising success at predicting the shape of the error and scale. Finally there is a discussion about the successes of the method applied to the preliminary investigations.
4.1 Descriptions of Preliminary Test Cases

4.1.1 Investigation of Convection-Diffusion Transport of $f$ and $q$ in 1D

The 1D convection-diffusion equation was used to test the method. Its simplicity allows the process to be investigated with a FORTRAN solver at each step. The FORTRAN solver was written by the author to solve the discretisation matrices using Gaussian elimination. The problem is the solution to Equation 3.1.2 for the transported variable $f$ in one dimension with the uniform velocity of the fluid $U$, specific diffusivity $\beta$, and no source $S$, with the boundary conditions $f(x=0) = 0$ and $f(x=L) = 1$ and becomes

$$\frac{df}{dx} - \beta \frac{d^2f}{dx^2} = 0 \quad (4.1.1)$$

The simplicity of the case also allows an analytical solution to be found and used to investigate the solutions. The exact solution to this equation is

$$f_e(x) = e^{\frac{x Pe}{L} - \frac{1}{e^{Pe} - 1}} \quad (4.1.2)$$

where $Pe = \frac{UL}{\beta}$ is the Peclet number. This equation represents the convection diffusion of a source of a passive scalar from a point source, for example temperature when the flow situation is a small hot object in a cross flow of fluid.

The second moment equation becomes

$$u \frac{dq}{dx} - \beta \frac{d^2q}{dx^2} = -\beta \left( \frac{df}{dx} \right)^2 \quad (4.1.3)$$

with solution

$$q_e(x) = \frac{1}{2} \left( e^{\frac{x Pe}{L} - \frac{1}{e^{Pe} - 1}} \right)^2 \quad (4.1.4)$$

Dividing the region into $n$ equally-spaced grid points, the differentials in Equations 3.1.2 and 4.1.3 can be estimated using central differencing. The method uses the Taylor series to find expressions for the gradients of $f$ using values of $f$ \[31\][34]. For $\frac{df}{dx}$ it is possible to find a first order accurate expression

$$\left( \frac{df}{dx} \right)^{(1)}_i = \frac{f_{i+1} - f_i}{\Delta x} \quad (4.1.5)$$

or a second order expression

$$\left( \frac{df}{dx} \right)^{(2)}_i = \frac{f_{i+1} - f_{i-1}}{2\Delta x} \quad (4.1.6)$$

where subscripts denote the grid point at which variables are evaluated, superscripts represent the order, and $\Delta x = \frac{L}{n}$ is the constant grid spacing separating them. For $\frac{d^2f}{dx^2}$ a first order expression is not possible and the second order expression is

$$\left( \frac{d^2f}{dx^2} \right)^{(2)}_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{(\Delta x)^2}, \quad (4.1.7)$$
When substituted into Equation 4.1.1 these form a set of simultaneous equations which can be used to calculate an estimate of $f$. After rearrangement the first order discretised equation becomes

$$-f_{i-1} + (2 - P e_L) f_i + (P e_L - 1) f_{i+1} = 0$$

(4.1.8)

and similarly the second order discretised equation is

$$\left( -\frac{P e_L}{2} - 1 \right) f_{i-1} + 2 f_i + \left( \frac{P e_L}{2} - 1 \right) f_{i+1} = 0$$

(4.1.9)

Once boundary conditions are applied these become equivalent to a matrix equation characterised by

$$A_{ij} f_i = b_i$$

(4.1.10)

where $A_{ij}$ is the matrix representation of Equation 4.1.8 or 4.1.9 and $b_i$ is the source term that includes any sources and the effect of the boundary conditions. These can be solved using matrix operations.

The second moment, Equation 4.1.3, is discretised similarly with the extra source term estimated from the solution for $f$ using Equations 4.1.8 or 4.1.9 and 4.1.6 to discretise the gradient to give

$$-q_{i-1} + (2 - P e_L) q_i + (P e_L - 1) q_{i+1} = -2 (f_{i+1} - f_i)^2$$

(4.1.11)

$$\left( -\frac{P e_L}{2} - 1 \right) q_{i-1} + 2 q_i + \left( \frac{P e_L}{2} - 1 \right) q_{i+1} = -\frac{1}{2} (f_{i+1} - f_{i-1})^2$$

(4.1.12)

or

$$A_{ij} q_i = b'_i$$

(4.1.13)

where $b'_i$ is the altered source term.

### 4.1.1.1 Solving for $f$ and $q$

The FORTRAN solver was used to solve Equation 4.1.9 using Gaussian elimination [14]. The range $x = [0, L]$ was divided into either fifty two or twenty seven regularly-spaced grid points, creating fifty or twenty five simultaneous equations. After applying the boundary conditions $x = 0$, $f = 0$ and $x = 1$, $f = 1$, the equations were solved to find the solution vector $f_i$. The exact solution is plotted in Figure 4.1.1 for five Peclet numbers. Increasing the Peclet number is equivalent to increasing the effect of convection or the velocity of the fluid. The effect of this is seen in the results as an increase in solution curvature.
The solution vector $f_i$ was used to estimate the $-\beta \left( \frac{df}{dx} \right)^2$ source term in Equation 4.1.3, creating the altered source vector $b'_i$. The boundary conditions for $q$ change to $x = 0$, $q = 0$ and $x = 1$, $q = \frac{1}{2}$. The FORTRAN solver was used to solve the SME for the solution vector $q_i$. The $q$ solution is shown in Figure 4.1.2, and shows more curvature in the profiles than for $f$. 

Figure 4.1.1: Solution of the 1D convection diffusion equation for the five Peclet numbers with 50 grid points.

Figure 4.1.2: Solution of the second moment form of the 1D convection diffusion equation for the five Peclet numbers with 50 grid points.
4.1.2 Investigation of Convection-Diffusion of a Passive Scalar from a Point Source in a Crossflow

The case considered now is a point source of a passive scalar in a crossflow of fluid with a uniform velocity, $u$. A physical example that this represents would be a small crystal of solute slowly dissolving in a uniform flow of solvent, and $f$ would represent the concentration of the solute. The equation governing the movement of the scalar $f$ in the fluid, from Equation 3.1.2, is

$$\frac{df}{dx} - \beta \nabla^2 f = S$$

where $S$ is the source term for $f$ representing a point source. This situation has an analytical solution in 3D which is

$$f(x, y, z) = \frac{S}{4\pi r^2} e^{-\frac{u(x-x)}{4\beta}}$$

where $r^2 = x_i x_i$, $x$ is the distance downstream from the source, and $S$ is the source strength [22][26]. Equation 4.1.15 allows exact error analysis of the Code_Saturne results.

The second moment equation for this test case becomes

$$u \frac{dq}{dx} - \beta \nabla^2 q = S f - \beta \nabla f \cdot \nabla f$$

with solution

$$q(x, y, z) = \frac{S^2}{32\pi \beta^2} e^{-\frac{u(x-x)}{4\beta}}$$

This test case was used previously by Jasak [26] in investigations of proposed error estimating methods. Here that investigation is extended by using the second moment solution method. Code_Saturne was used with laminar conditions and a steady flow solver to solve for $f$ and $q$.

Figure 4.1.3 displays the setup of the point source in the uniform flow and the location of the mesh geometry. The computational domain was set up by a 3D rectangular grid starting at $x = 0.05m$ just next to the point source at $x = 0m$, avoiding the infinity of the point source. As shown in Figure 4.1.3, in the $y$ direction the grid was centred on the point source. It was the same in the $z$ direction. The dimensions of the domain were $1m$ by $1m$ in the $y$ and $z$ directions and $4m$ in the $x$ direction. The grid densities were 24 cells per $m$ for the coarse mesh and 12 cells per $m$ for the very coarse mesh.
The inlet boundary next to the source was set as a Dirichlet boundary equal to the analytical values of the scalar concentration using Equation 4.1.15 and with uniform velocity in the $x$ direction. The opposite boundary, at $x = 4.05m$, was set as a free outlet. The remaining boundaries were also set as Dirichlet boundaries equal to the analytical solution as well, also with $x$ velocity $u$.

A number of velocities were used which varied the cell Peclet number, $Pe_L$, from low to high and these are summarised in Table 4.1.1. It shows a list of values of the $Pe_L$ that were investigated alongside the velocity that was required to produce them on each mesh.

<table>
<thead>
<tr>
<th>$Pe_L$</th>
<th>Very Coarse Mesh $u \text{ ms}^{-1}$</th>
<th>Coarse Mesh $u \text{ ms}^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.417</td>
<td>0.25</td>
<td>0.5</td>
</tr>
<tr>
<td>0.833</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>1.250</td>
<td>0.75</td>
<td>1.5</td>
</tr>
<tr>
<td>1.667</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2.500</td>
<td>1.5</td>
<td>3</td>
</tr>
<tr>
<td>3.333</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4.167</td>
<td>2.5</td>
<td>5</td>
</tr>
<tr>
<td>5.000</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>6.667</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>8.333</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>10.000</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>11.667</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td>13.333</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>15.000</td>
<td>9</td>
<td>18</td>
</tr>
<tr>
<td>16.667</td>
<td>10</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 4.1.1: Summary of $Pe_L$ tested and the corresponding velocities on both of the meshes for the point source in a cross flow test case.
4.1.2.1 Solving for $f$ and $q$

The distribution of the Code_Saturne $f$ solution found using the second order discretisation on the coarse mesh is shown as a colour map in Figure 4.1.4. Only a cut through the 3D geometry is shown. The plane is cut normal to the $z$ axis that goes through the origin and the source. The cross flow causes the source to be convected downstream while diffusion spreads it out. The walls were constrained to the analytical values and so any error in the simulation will appear in the middle. This isolates only a single region of interest in this test case allowing a controlled investigation. Later test cases will have multiple regions generating error, and effects from individual ones may be indistinguishable from the effects of others.

![Image: Coarse Mesh Second Order $f$ Solution](Coarse Mesh Second Order $f$ Solution)

Figure 4.1.4: Solution of the scalar transport equation for the point source in a cross flow test case in Code_Saturne.

The same boundary types, mesh and discretisation scheme were used to solve Equation 4.1.16 for $q$ using Code_Saturne, and is shown in Figure 4.1.5. It shows a similar shape to the $f$ solution but with higher gradients, as expected from squaring. The maximum $q$ value is of the correct order as the squared maximum $f$ value, though the two do not match exactly. $\max(f)^2 \approx 49600$, $\max(q) \approx 51600$. This discrepancy between the two is an example of where information about the errors may be located.

![Image: Coarse Mesh Second Order $q$ Solution](Coarse Mesh Second Order $q$ Solution)

Figure 4.1.5: Solution of the second moment equation for the point source in a cross flow test case in Code_Saturne.
4.1.3 2D Navier-Stokes Solver Applied to Burggraf Flow

To test the Navier-Stokes version of the proposed method the case of Burggraf flow was used. This is a lid driven flow in a square box that also has an additional forcing term that drives the solution to one that is analytically solvable. This test case can only test the velocity error estimation because the transport of a scalar in this flow will not lead to a steady solution.

The fluid is enclosed in a square box with four walls. The upper wall of the box was set as a moving wall boundary condition where the velocity was prescribed. This geometry is shown in Figure 4.1.6. It shows that the prescribed velocity moves in the horizontal direction along the top wall.

![Figure 4.1.6: The geometry of the Burggraf flow test case.](image)

For this situation the momentum equation includes the force \( \mathbf{F} \) and becomes

\[
\mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla P}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{F} \tag{4.1.18}
\]

The velocity solution to this was chosen to be \( \mathbf{u}_e = (u, v, 0) \), where \( u = 8g'h', \ v = -8g''h, \) with \( g = \frac{5}{5} - \frac{4}{7} + \frac{3}{9}, \ h = y^4 - y^2, \) and primes indicate differentiation with respect to their respective variables [28]. The forcing term required was chosen accordingly so as to drive the numerical solution towards the analytical solution. The forcing term applied can be derived from \( \mathbf{u}_e \), here it is simply stated as \( \mathbf{F} = (0, F_y, 0) \) where

\[
F_y = \left\{ \frac{8}{Re} \left[ 24g + 2g''h'' + g''''h \right] + 64 \left[ \frac{g'^2}{2} (hh'' - h'h'') - hh' (g'g'' - g''') \right] \right\} \tag{4.1.19}
\]

Figure 4.1.6 indicates that the forcing term acts on the fluid throughout the geometry, and will have the effect of forcing the fluid round the box.

The magnitude of this source term is small compared to other terms in the Navier-Stokes equation but its inclusion makes a more interesting case to investigate. This technique is the method of
manufactured solutions. The system is not necessarily a real physical case because the complexity of the forcing term would be almost impossible to set up in the fluid. This method was used for this investigation because an analytical solution will allow an error analysis that is simple to implement.

The second moment equation for this test case is

$$\mathbf{u} \cdot \nabla K = -\frac{\mathbf{u} \cdot \nabla P}{\rho} + \nu \nabla^2 K + \mathbf{u} \cdot \mathbf{F} - \nu [\nabla \mathbf{u} \cdot \nabla \mathbf{u} + \nabla \mathbf{v} \cdot \nabla \mathbf{v}] \quad (4.1.20)$$

with solution

$$K_e = \frac{u^2 + v^2}{2}$$

### 4.1.3.1 Solving for the Velocity $\mathbf{u}$ and the Total Kinetic Energy $K$

The simulation was performed in the domain $(0,0)$ to $(1,1)$ using two meshes with the number of cells along the walls being 25 and 50, referred to as the very coarse mesh and the coarse mesh respectively. The Reynolds number was equal to 10. The velocity along the top wall was set equal to the analytical solution. In Figure 4.1.7 the stream lines are shown for the velocity, found with the coarse mesh. The velocity solution shows that the fluid is moving in a circle around a centre position at approximately three quarters up the centre line. The fastest velocity is at the top wall where most of the driving of the fluid occurs.

![Figure 4.1.7: The velocity stream lines for the Code_Saturne simulation of the Burggraf flow with coarse mesh.](image)

The SME was solved similarly with constant value boundary conditions. The value of the total kinetic energy at the top wall was found using the analytical solution for $K$, and for the rest of the walls it was set equal to zero. The kinetic energy distribution predicted by this is shown in Figure 4.1.8. The highest kinetic energy is along the top wall where the velocity is forced, and there is a ring of lower kinetic energy around the centre point.
Figure 4.1.8: The total kinetic energy for the *Code_Saturne* simulation of the Burggraf flow with coarse mesh.
4.2 Errors and Error Estimates for the Preliminary Test Cases

4.2.1 1D Convection Diffusion

4.2.1.1 $f$ Error and Error Estimate Distributions

The solution errors are displayed in Figure 4.2.1 as a percentage of the volume averaged value of the scalar solution, $f$. Alongside these are the corresponding estimates calculated using Equation 3.2.27. Higher Peclet numbers produce more curved $f$ solutions, and the larger gradients in these solutions cause a greater loss of information in the Taylor series truncation. Therefore the errors are larger for higher Peclet numbers. The errors are zero at the boundaries where the solution is defined, and higher in the middle with the peak around $x = 0.7$. The skewing away from the centre is a result of the higher gradients at higher $x$ leading to larger losses during the discretisation, hence a higher error is seen.

The estimates show a very close resemblance to the errors. The estimates go to zero at the boundaries and there is one main central peak skewed towards high $x$, similar to the error distributions. A slight difference is that the estimates are slightly more curved than their corresponding errors. The scales are within the same order of magnitude although the estimate has over-predicted.

![Numerical Error and Error Estimate](image)

Figure 4.2.1: The 1D convection-diffusion equation solution errors (left) and corresponding estimates (right), using the second order centred (top) and first order (bottom) discretisation schemes, for the five Peclet numbers. They are expressed as a percentage of the volume average of $f$. 

86
4.2.1.2 Correlation Coefficient

The correlation coefficients were calculated using Equation 3.2.34 for the three simulations and these are shown in Figure 4.2.2 as simulation vs correlation coefficient. The values are close to 1 indicating the estimates are good predictions of the shapes of the errors. The correlation decreases as the Peclet number increases indicating that the shape prediction gets worse.

Figure 4.2.2: The 1D convection-diffusion equation correlation coefficients for the two meshes (coarse and very coarse), the two discretisation schemes (first order upwind and second order centred), and the five Peclet numbers.

4.2.1.3 Summation Error

The summation error was calculated according to Equation 3.2.32 for the three sets of results. They are shown in Figure 4.2.3 graphed as summation error against summation estimate. A linear relationship is seen between the two. The estimate scale consistently over-predicts the error scale by a factor of 3 for the second order scheme and by a factor of 2 for the first order scheme.

Figure 4.2.3: The summation error and summation estimate for the two meshes (coarse and very coarse) and the two discretisation schemes (first order upwind and second order centred). They are expressed as a percentage of the maximum value of $f$ found in each of the simulations.
4.2.2 Point Source in a Crossflow

4.2.2.1 $f$ Error and Error Estimate

To view the exact absolute error the $\text{Code}_\text{Saturne}$ solution was subtracted from the analytical solution evaluated at the centre of each cell. The estimate was calculated according to Equation 3.2.27. Two examples are shown in Figure 4.2.4, one using the first order discretisation scheme, the other the second order centred scheme. Both were found using the coarse mesh with $Pe_L = 0.833$ (or equivalently $u = 1ms^{-1}$).

![Coarse Mesh First Order](image1.png)

![Coarse Mesh Second Order Centred](image2.png)

Figure 4.2.4: Examples of the exact errors for the point source in a cross flow $\text{Code}_\text{Saturne}$ solution with $u = 1$. The error distributions include a contour outlining the values at 5% of the maximum value of $f$ for each.

In both there is a peak of error that quickly spreads and drops off to low error. The highest error is found in a three cell radius around the closest point to the source at the inlet. The error spreads out from here in angular bands of high and zero error. These zero error bands are highlighted by contours. The zero-bands are caused by the numerical solution swapping between being larger than the analytical solution to being lower. Finding the absolute value therefore creates the bands of ‘zero’ error that are seen.

The estimates are similar to the errors with respect to the important features. There is a large maximum value close to the source at the inlet which drops off quickly downstream and to the sides. However, the zero-bands were not seen in the estimate. If the banding had not been present in the error distributions the two would have been a closer match in shape. These bands of ‘zero’ error are misleading. These are still regions that are capable of generating error, and it was a coincidence that under these particular simulation conditions $f - f_e$ changed sign at those particular locations. By ignoring the misleading bands, the error estimate is therefore providing more valuable information.
about the error generating locations than the actual real error distribution.

The scale prediction seen in the estimate is a large over-prediction. The banding has had an effect on this too, thereby lowering the maximum value in the distribution. For example in the coarse mesh first order error distribution the bands have intersected the main region of error. Therefore the maximum value has been reduced and the scale seen in the estimate is a large over-prediction. The banding is also going to lead to a reduced averaged error, so the summation errors are expected to be lower than the summation estimates as well.

### 4.2.2.2 Correlation Coefficient

The correlation coefficients were calculated according to Equation 3.2.34. These are displayed in Figure 4.2.5 as correlation coefficient against $Pe_L$. Again a relationship between $Pe_L$ and the results is seen. The correlations are lower for low $Pe_L$ at about 0.7 and increase to 0.9 at higher $Pe_L$. As the Reynolds number increases the shape evolves and the zero-banding disappears from the error distribution. The error distribution becomes more similar to the error estimate and the correlation coefficients increase. Therefore the relatively low correlation coefficients for the low $Pe_L$ results are a consequence of the banding.

![Correlation Coefficients for all Results](image)

Figure 4.2.5: Point source in a cross flow test case correlation coefficients against $Pe_L$ for the four simulations with two meshes and two discretisation schemes.

### 4.2.2.3 Summation Error

The summation errors and summation estimates were calculated using Equations 3.2.32 and 3.2.33. In Figure 4.2.6 they are shown as error against estimate for the $Pe_L$ values tested. Each set describes an clear relationship between the averaged quantities that is dependent on $Pe_L$. The low $Pe_L$ results start close to the origin, and increase in error approximately linearly with the estimate as $Pe_L$ increases with the estimate over-predicting the scale by a factor of 2. Then the cell Pelet number reaches a certain limit and the estimate stops increasing as quickly, while the error continues to increase.
The point where the lines begin to diverge is the same for both meshes. The effects are visible after the 6th data point, which corresponds to $Pe_L = 4.167$. In standard practice the recommended upper $Pe_L$ limit to ensure numerical stability is $Pe_L = 2$ [14]. The method stops being able to predict the scale of the errors accurately soon after the limit has been passed. This topic will be revisited in the main test case sections.

![Averaged Error against Averaged Estimate for Results on the Very Coarse Mesh](image1)

![Averaged Error against Averaged Estimate for Results on the Coarse Mesh](image2)

Figure 4.2.6: The summation error and estimate for the point source in a cross flow test case. Displayed as error against estimate for the two discretisation schemes: FO and SOC. They are expressed as a percentage of the volume average of $f$. 

90
4.2.3 Burggraf Flow

4.2.3.1 Laminar Flow Error and Error Estimate

The errors were calculated according to Equation 3.2.14, which is the velocity error vector, \( \chi = u - u_e \). The error estimate was calculated according to Equation 3.2.30. Figure 4.2.7 shows the error and estimate distributions for the first order upwind and second order centred schemes, found using the coarse mesh. The summation and correlation coefficients are summarised in Table 4.2.1. The first order error shows a ring of high error in the centre, along the path of the fluid. The error goes to zero at the boundaries. The estimate predicts a similar shape, which is reflected in the high correlation coefficients between 0.7 and 0.8. The main difference is that the centre of the ring does not go to zero in the estimate.

The second order centred error distribution is a different shape. The main source of error is seen around the top and side edges in the first cells next to the walls, and there is much lower found in the centre of the geometry. These features are seen in the estimate distribution; however, the prediction at the top wall spreads out into the centre of the geometry. This caused lower correlation coefficients of between 0.45 and 0.55.

The estimate scale prediction is consistently over-predicting the error scale by up to a factor of 15. The error magnitudes are very small and the worry is that these will give misleading results when compared to the estimates. This is because comparing two values that are approaching zero could be influenced by some small scale effects not related to the simulation error. Higher Reynolds numbers would produce higher errors. However, this cannot be tested further because the method used to force an analytical solution requires low Reynolds numbers. At higher Reynolds number the fluid flow becomes more asymmetrical as it diverges away from the analytical solution. The beginning of this asymmetry can be seen in the error distributions as a result of a slightly too high Reynolds number.

<table>
<thead>
<tr>
<th>Simulation</th>
<th>Summation Error</th>
<th>Summation Estimate</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very coarse mesh FO</td>
<td>1.52</td>
<td>23.4</td>
<td>0.704</td>
</tr>
<tr>
<td>Very coarse mesh SOC</td>
<td>0.686</td>
<td>3.73</td>
<td>0.530</td>
</tr>
<tr>
<td>Very coarse mesh SOLU</td>
<td>0.686</td>
<td>4.00</td>
<td>0.554</td>
</tr>
<tr>
<td>Coarse mesh FO</td>
<td>0.742</td>
<td>14.8</td>
<td>0.783</td>
</tr>
<tr>
<td>Coarse mesh SOC</td>
<td>0.167</td>
<td>2.33</td>
<td>0.422</td>
</tr>
<tr>
<td>Coarse mesh SOLU</td>
<td>0.167</td>
<td>2.41</td>
<td>0.459</td>
</tr>
</tbody>
</table>

Table 4.2.1: Summary of the Burggraf flow summation and correlation coefficients with \( Re = 10 \).
4.2.4 Discussion of Preliminary Test Case Results

In general the error estimation method showed success when applied to the preliminary test cases, including the scalar and the momentum variants. In all cases it was possible to use the estimate to determine the regions of high error and the regions of low error. Often the differences that arise were the result of zero-bands cutting through the distributions. The shapes did not match exactly as sometimes the location of maximum error was predicted in slightly different positions, but the estimates were able to produce consistently high values of the correlation coefficient. In most cases the estimate scale prediction was an over-prediction of the error scale. This is ideal in the industrial circumstances for which this method is intended. It provides confidence that the prediction is below a certain level rather than leaving uncertainty if it is above or below. The factor by which the estimate scale over-predicts varies, but it is normally within an order of magnitude. A feature of the results that requires further investigation is the breakdown of the scale prediction as the Reynolds number/Peclet number increases.

Figure 4.2.7: Examples of the Burggraf flow error and estimate distributions with \( Re = 10 \).
4.3 Chapter 4 Summary

Chapter 4 presented and analysed the results found using the three preliminary test cases; 1D convection diffusion, the point source in a cross flow, and the Burggraf flow. The results showed that the method had some success at predicting the errors, but also hinted at some limitations that may be present. To investigate further, more complex test cases must be used. The remainder of the thesis is spent on the main test cases. These involve more complex flows than were encountered in chapter 4.

Chapter 4 introduced the layout of the test case descriptions being presented separately from the results and discussion. This structure will be used again to reduce repetition of test case descriptions, descriptions of the methods, and discussion of the results. Chapter 5 will give the descriptions of the three main test cases; the ribbed channel, the impinging flow, and the hexagonally stacked tube bundles. The results and analysis will be presented in later chapters.
Chapter 5

Main Test Case Descriptions

Chapter 5 gives a description of the three main test cases used to investigate the method. Each case tests the error estimate for scalar transport in laminar fluid flow, and for laminar momentum-KE. The test cases investigated are ones that are similar to commonly found situations in power generation systems. These are, in 2D, the ribbed channel, impinging flow, and hexagonally packed tube bundles. The next three sections describe each of the test case geometries and display example distributions of the velocity and scalar solutions. The error and estimate comparisons are reserved for the chapter 6. This was done to produce results chapters that grouped together similar parts of the analysis, instead of grouping by test case. This was done to reduce the repetition of test case descriptions, the methodology, and of the important discussions and conclusions.

The scalar transport investigations involve transport of a passive scalar across a ‘hot’ wall into the fully developed fluid flow moving over it. The scalar will be introduced by one of the boundaries being set to have either constant flux or constant value of the scalar \( f \). All other boundaries will have the value of the scalar equal to zero. If the constant flux boundary has a flux of \( f \) equal to \( j \), the same boundary type is used for the \( q \) simulation and the flux is equal to \( j \frac{\partial f}{\partial y'} \), where \( y' \) is the normal to the wall. If the constant value boundary has a value \( m \), then the same boundary type is used for the \( q \) simulation and the value is equal to \( \frac{m^2}{2} \). At the inlet both \( f \) and \( q \) are equal to zero so that it is the relative change in the scalar that is seen.

The velocity is driven by either a set inlet velocity in non-periodic simulations or a forcing term in a periodic simulation. When a forcing term is used this must be included as an additional source in the second moment equation. The kinetic energy boundary conditions match the velocity boundary conditions with a constant value equal to zero at the walls. For laminar simulations this is all that needs to be set.

The preliminary work suggested some dependence on the cell Peclet number. However, for these test cases, a single \( Pe_L \) is more complicated to define than it was for the cross flow test case. This is because \( Pe_L \) varies with the velocity over the domain. The centre of the channel will have the highest \( Pe_L \), and close to the walls \( Pe_L \) will be small. For example:

- In general the scalar does not reach the centre of the flows significantly and so there will be low error here. The possible effects of a high \( Pe_L \) in the centre will not affect the error estimate
significantly in this region.

- Near the walls the value of the scalar will be high, leading to more significant error in this region. However, $Pe_L$ is lowest here, therefore the effects will not affect the error estimate significantly either.

Another complication in choosing the $Pe_L$ definition is that these simulations have multiple locations of high error. The different regions will reach a critical $Pe_L$ limit at different times as the Reynolds number increases. This means that as $Re$ increases there could be multiple stages where successive regions may experience the effects of a high $Pe_L$. Compare this to the simplicity of the cross flow test case where there was only one significant region and one global $Pe_L$ value per $Re$. For this work, the cell Peclet number will be simply calculated using the bulk velocity rather than the local velocities. This is effectively an averaged $Pe_L$ value that was chosen to represent the flow.

Each test case uses three meshes. The first is a refined mesh that is fine enough to provide an essentially error free solution. The test cases investigated do not have analytical solutions and so the refined mesh solutions serve that purpose. The two other meshes were the coarse and very coarse mesh, which were coarse enough to generate errors of numerical significance. The meshes were created using the mesh generation software ANSYS ICEM CFD. Each mesh was created by first defining the edges and surfaces of the geometry. The number of cells along each set of parallel edges was defined. Finally the mesh is generated under these parameters, creating a structured grid of hexahedral cells. More information about the software and the mesh generation process can be found in [4].
5.1 Ribbed Channel

The first more complicated test case examines the situation of flow along a ribbed channel. The flow geometry is a channel in 2D with small, square, regularly spaced protrusions sticking out of the lower wall, which are the ribs. A diagram of this is shown in Figure 5.1.1 that includes the relative dimensions of the ribs, their spacing and the channel width. The coarse mesh had fifty cells across the channel length and five cells along the rib walls. The very coarse mesh had twenty five cells across the channel length and two cells along the rib walls. These meshes are displayed in Figure 5.1.2. The flow presents complications to finding CFD solutions of the flow because of the corners of the ribs. These cause singularities in the flow; refining the mesh in these areas changes the entire flow solution. The scalar simulation is the scalar transport of a passive scalar into the fully developed fluid flow from the ribbed wall.

A physical example of the situation is the temperature of the coolant flowing across the hot ribbed fuel rods in a nuclear reactor. The ribs are there to maximise heat transfer into the fluid. The bottom wall and rib walls will have either a constant flux of the scalar into the fluid or held at a constant value. Physically, a constant flux boundary would be similar to the fuel being at criticality where the output of heat is constant (with the assumption the output spreads to the ribs). A constant value would be more similar to flow over a boiler in a nuclear reactor. The walls in the boiler are kept at a constant temperature by the rapidly replaced reactor coolant flowing through. The boiler fluid therefore experiences a constant temperature on the wall surfaces.

![Figure 5.1.1](image)

Figure 5.1.1: A diagram of the ribbed channel geometry indicating the direction of flow and the non-zero scalar boundaries.
5.1.1 Laminar Velocity Solution

This test case finds converged solutions to the laminar flow at Reynolds numbers that reproduce certain features seen in a turbulent case. In [6] results were presented for a turbulent ribbed channel flow that shows a large recirculation region with the stagnation point about 40\% along the length of the bottom wall. The flow solution will be found in a periodic simulation, representing the fully established flow in a long ribbed channel.

A range of Reynolds numbers were used to investigate the flow and scalar transport for this test case. The velocity was driven with a constant source term to the velocity equation, which was adjusted by regular intervals to produce a range $Re$. Table 5.1.1 shows the Reynolds numbers used along with the typical value of the cell Peclet number they produced in each of the two meshes.

<table>
<thead>
<tr>
<th>$Re$</th>
<th>Very Coarse Mesh $Pe_L$</th>
<th>Coarse Mesh $Pe_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>44.6</td>
<td>0.992</td>
<td>0.496</td>
</tr>
<tr>
<td>88.4</td>
<td>1.97</td>
<td>0.983</td>
</tr>
<tr>
<td>132</td>
<td>2.93</td>
<td>1.46</td>
</tr>
<tr>
<td>175</td>
<td>3.88</td>
<td>1.94</td>
</tr>
<tr>
<td>260</td>
<td>5.77</td>
<td>2.88</td>
</tr>
<tr>
<td>344</td>
<td>7.63</td>
<td>3.82</td>
</tr>
<tr>
<td>674</td>
<td>15.0</td>
<td>7.49</td>
</tr>
<tr>
<td>999</td>
<td>22.2</td>
<td>11.1</td>
</tr>
<tr>
<td>1320</td>
<td>29.4</td>
<td>14.7</td>
</tr>
<tr>
<td>1640</td>
<td>36.5</td>
<td>18.2</td>
</tr>
</tbody>
</table>

Table 5.1.1: Table showing the Reynolds numbers tested in the ribbed channel test case and the corresponding cell Peclet numbers.

The flow found for the refined mesh simulation for Reynolds numbers 344 and 1320 are shown in Figure 5.1.3. In each there is a recirculation region after the first rib and it increases in size.
as the Reynolds number increases. In each there is also a much smaller recirculation region in the corner created by the downstream rib and the lower wall which also increases with Reynolds number. Although not shown here, it is worth noting that the shape of the recirculation region does change in size with refinement but not significantly. This is because the two simulations produce slightly different flow speeds and the recirculation region is affected by this.

Figure 5.1.3: Examples of the velocity streamlines for the ribbed channel test case. The scale was normalised by $U_0$.

Figure 5.1.4 shows the velocity magnitude and total kinetic energy found using the three mesh densities at $Re = 674$. It shows that almost all of the kinetic energy is found in the centre of the channel, and it is much higher than the kinetic energy found around the ribs.

Figure 5.1.5 shows the convergence of the $U$ and $K$ simulations with $Re = 674$ on the coarse mesh, displayed as the volume average of $U$ and $K$ divided by the final volume averaged value they achieve against iteration number. The $K$ convergence initially lags behind the velocity simulation, however, the two reach convergence after approximately the same number of iterations.
Figure 5.1.4: The $U$ and $K$ solutions for the ribbed channel test case found using the three mesh densities at $Re = 674$. The velocity scale was normalised by $U$, and the total kinetic energy solution by $\frac{U^2}{2}$. 
5.1.2 Scalar Solution in Laminar Flow

The fully developed periodic flow solutions, found using the SOC scheme including the slope test, for each mesh were used to initialise the scalar simulations. This frozen velocity, one that is not transported, keeps the velocity field the same as the fully developed periodic flow. This is necessary because the scalar will not reach a steady state in a periodic simulation. The investigation simulates the transport of a passive scalar into this fully developed flow as if the first ‘hot’ section ribs were just being encountered. Using the three discretisation schemes the first and second moments were solved to find $f$ and $q$. The values of both scalars were set to equal zero at the inlet. On the upper wall, both had flux equal to zero.

For the constant flux boundary condition simulation the flux of $f$ through the lower wall and rib walls was set as $0.1[f] n^{-1}$. The flux of $q$ through those walls was set to $0.1 \frac{\partial f}{\partial x'}[f]^2 n^{-1}$ where $x'$ is the wall normal direction. Therefore this requires information from the $f$ simulation to set the boundary condition. For the constant value boundary condition simulation the value of $f$ at the lower wall and rib walls was set as $10[f]$. The value of $q$ at these walls was set as $50[f]^2$.

The $f$ distributions found using the three mesh densities and both boundary types are shown in Figure 5.1.6 for $Re = 674$. The constant flux result shows more cooling between the ribs than for constant value. For the constant flux result the highest values of $f$ are in the corner where there is the most input of $f$ and least fluid circulation. Compared to the constant value boundary result, the walls are held at the highest value and so there is less cooling in-between the ribs. As the fluid velocity is increased the amount of cooling will increase.
Figure 5.1.6: The $f$ solutions for $Re = 674$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the bottom walls.

The $q$ solutions found using the three mesh densities and both boundary types are shown in Figure 5.1.7 for $Re = 674$. Using the constant flux boundary the $q$ distribution has the highest values in the corners similar to the $f$ solution, as expected from simply squaring. Using the constant value boundary the $q$ distribution has highest value along the walls.

Figure 5.1.8 shows the convergence of the $f$ and $q$ simulations with $Re = 674$ on the coarse mesh for both boundary condition types, displayed as the volume average of $f$ and $q$ divided by the final volume averaged value they achieve against iteration number. In the constant value boundary simulation the $f$ and $q$ simulations converge almost in unison and they both converge faster than in the constant flux simulation. In the constant flux boundary simulation the $q$ convergence initially lags behind the $f$ simulation, however, the two reach convergence after approximately the same number of iterations.

The convergence of the tube bundle simulation with $Re = 144$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number.
Figure 5.1.7: The $q$ solutions for $Re = 674$ found using the three mesh densities and with constant flux (left) and constant value (right) boundary conditions along the bottom walls.

Figure 5.1.8: The convergence of the ribbed channel simulation with $Re = 674$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number.
5.2 Impinging Flow

This test case investigates the situation of flow through a 2D channel that impinges onto a perpendicular wall, splitting the flow. Figure 5.2.1 shows a diagram of the geometry and indicates the relative dimensions, the directions of flow and the wall that is heated. The coarse mesh had sixteen cells across the channel width and the very coarse mesh had eight cells across, and these are displayed in Figure 5.2.2. The geometry is an inlet channel long enough to set up developed flow that then meets a perpendicular channel of the same thickness. This creates an impingement on the back wall which is held at a constant flux or value of $f$. The impingement point in this flow, where the centre line of the inlet channel meets the heated wall, is a singularity and so it should be able to generate significant errors in the simulation.

This situation is very common in practice, whenever a pipe joins another for example. A physical example of this would be cold coolant coming from outside a system and joining a pipe that is inside a hot chamber. The back wall could be hot and the scalar would be the temperature of the fluid.

![Figure 5.2.1: The geometry of the 2D impinging flow test case, indicating the relative dimensions and the fluid directions.](image)
5.2.1 Laminar Velocity Solution

The velocity inlet condition consisted of a constant speed into the pipe and was uniform across the inlet. The flow down the inlet channel was able to become fully developed before the junction. The inlet speed was adjusted to change the Reynolds numbers for the investigation. The $Re$ tested are summarised in Table 5.2.1, along with the typical $Pe_L$ values they produced on both of the meshes.

<table>
<thead>
<tr>
<th>$Re$</th>
<th>Very Coarse Mesh $Pe_L$</th>
<th>Coarse Mesh $Pe_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>62.5</td>
<td>0.78</td>
<td>0.39</td>
</tr>
<tr>
<td>125</td>
<td>1.56</td>
<td>0.78</td>
</tr>
<tr>
<td>187.5</td>
<td>2.34</td>
<td>1.17</td>
</tr>
<tr>
<td>250</td>
<td>3.13</td>
<td>1.56</td>
</tr>
<tr>
<td>375</td>
<td>4.69</td>
<td>2.34</td>
</tr>
<tr>
<td>500</td>
<td>6.25</td>
<td>3.13</td>
</tr>
<tr>
<td>1000</td>
<td>12.50</td>
<td>6.25</td>
</tr>
<tr>
<td>1500</td>
<td>18.75</td>
<td>9.38</td>
</tr>
<tr>
<td>2000</td>
<td>25.00</td>
<td>12.50</td>
</tr>
<tr>
<td>2500</td>
<td>31.25</td>
<td>15.63</td>
</tr>
</tbody>
</table>

Table 5.2.1: Table showing the Reynolds numbers tested in the impinging flow test case and the corresponding cell Peclet numbers.

Velocity stream lines for $Re = 1000$ and $Re = 2000$ are shown in Figure 5.2.3. The inlet channel was long enough for the flow to become fully developed before it enters the horizontal pipe. It hits the back wall and splits at the impingement point. The flow then moves down the side channels and
returns to a developed channel flow. There are recirculation regions after the corners of the junction. As $Re$ increases the recirculation regions become larger.

Figure 5.2.3: Examples of the velocity streamlines for the impinging flow test case. The scale was normalised by $U_b$.

Figure 5.2.4 shows the velocity magnitude and total kinetic energy found using the three mesh densities at $Re = 1000$. Only one half of the geometry is shown. The highest kinetic energy is in the centre of the inlet flow. After the junction the flow has high kinetic energy as it moves past the recirculation region. Then the flow returns to a developed channel flow with lower kinetic energy equal to half the energy of the inlet flow.
Figure 5.2.4: The $U$ and $K$ solutions for the impinging flow test case found using the three mesh densities at $Re = 1000$. The velocity scale was normalised by $U$, and the total kinetic energy solution by $\frac{U^2}{2}$.

Figure 5.2.5 shows the convergence of the $U$ and $K$ simulations with $Re = 1000$ on the coarse mesh, displayed as the volume average of $U$ and $K$ divided by the final volume averaged value they achieve against iteration number. The $K$ convergence lags behind the velocity simulation and converges significantly later than the $U$ solution.

Figure 5.2.5: The convergence of the impinging flow simulation with $Re = 1000$ on the coarse mesh, displayed as the volume averaged velocity, $\langle U \rangle$, and total kinetic energy, $\langle K \rangle$, divided by their respective final values, against iteration number.
5.2.2 Scalar Solution in Laminar Flow

The scalar simulations were run at the same time as a velocity calculation that used the SOC scheme including the slope test. Using the three discretisation schemes the first and second moments were solved to find \( f \) and \( q \). The values of both scalars were set equal to zero at the inlet.

For the constant flux boundary condition simulation the flux of \( f \) through the lower wall was set as \( 100[f]m^{-1} \). The flux of \( q \) through this wall was set as \( 100\frac{\partial f}{\partial x}[f]^2m^{-1} \) where \( x' \) is the wall normal direction. For the constant value boundary condition simulation the value of \( f \) at the lower wall and rib walls was set as \( 10[f] \). The value of \( q \) at these walls was set as \( 50[f]^2 \). All other walls had flux of the scalars equal to zero. The inlet values for both were set to zero.

The \( f \) distributions found using the three mesh densities and both boundary types are shown in Figure 5.2.6 for \( Re = 1000 \). The constant value result shows that the region around the impingement point is cooled relative to further downstream, but remains high due to the fixed wall value. This is where the inlet fluid impinges and rapidly convects the scalar, and as the fluid velocity is increased the amount of cooling will increase. At high Reynolds numbers this will lead to high gradients of the scalar near the wall in the constant value boundary simulations. In comparison, the constant flux boundary results show much more cooling in the impingement region and the wall gradients will be lower.

The \( q \) solutions found using the three mesh densities and both boundary types are shown in Figure 5.2.7 for \( Re = 1000 \). They show a similar pattern to the \( f \) solutions. Both show cooling near the impingement point where the scalar is convected away. The constant value \( q \) distribution will have a larger gradient of the scalar next to the wall compared to the constant flux simulation.

Figure 5.2.8 shows the convergence of the \( f \) and \( q \) simulations with \( Re = 1000 \) on the coarse mesh for both boundary condition types, displayed as the volume average of \( f \) and \( q \) divided by the final volume averaged value they achieve against iteration number. In the constant value boundary simulation the \( f \) and \( q \) simulations converge almost in unison and they both converge as fast as the constant flux \( f \) simulation. In the constant flux boundary simulation the \( q \) convergence initially lags behind the \( f \) simulation, however, the two reach convergence after approximately the same number of iterations.
Figure 5.2.6: The $f$ solutions for $Re = 1000$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the bottom wall.
Figure 5.2.7: The $q$ solutions for $Re = 1000$ found using the three mesh densities and with constant flux (left) and constant value (right) boundary conditions along the bottom wall.

Figure 5.2.8: The convergence of the impinging flow simulation with $Re = 1000$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number.
5.3 2D Stacked Tube Bundles

This test case investigates the situation of flow around hexagonally stacked tube bundles, one of which is heated. This flow does not have an analytical solution. The flow domain is an infinite array of parallel cylinders with fluid flowing perpendicular to them as shown in Figure 5.3.1. The fluid is shown coming from the left and meeting the cylinders. The domain that is meshed for the test case simulations is also shown.

Figure 5.3.1: A diagram to demonstrate the stacked tube bundle set up and highlight the flow geometry that was simulated.

Figure 5.3.2 shows an enlargement of the geometry meshed for the simulations. The top, bottom, front and back walls are symmetry boundaries. For the velocity simulations the inlet and outlet are replaced with periodic boundary conditions creating an infinite array of cylinders. For simulations where a scalar is transported the simulation is non-periodic. The cylinder walls are standard wall boundaries. The central cylinder is heated, while the left and the right cylinders are not. The meshes for this geometry split each quarter of the geometry into five sections. The two coarse meshes used are displayed in Figure 5.3.3 and indicate the five sections for clarity. The coarse mesh had sixteen cells across the four edges of each of these sections. The very coarse mesh had eight cells across these lengths.

This situation is commonly seen when cooling is required. A physical example would be in a nuclear reactor core where fluid flows through stacks of fuel rods, which are producing heat. If one was exceptionally hot, then a situation matching this test case would arise.
Figure 5.3.2: The geometry of the meshes used to simulate flow through stacked tube bundles. Red lines are heated, blue lines are not.

Figure 5.3.3: The two coarse meshes used in the tube bundle test case. On the left side of the very coarse mesh five sections are labeled and outlined.
5.3.1 Velocity and Total Kinetic Energy Solution

The flow is driven by a constant source term to the \( x \) component of the momentum equation in the \textit{Code_Saturne} user subroutine \textit{ustsns}. This driving source was varied to produce a range of Reynolds numbers, and the \( Re \) that were tested are summarised in Table 5.3.1. The velocity solutions were found with periodic boundary conditions along the inlet and outlet.

\[
\begin{array}{|c|c|c|}
\hline
Re & \text{Very Coarse Mesh } Pe_L & \text{Coarse Mesh } Pe_L \\
\hline
79.9 & 4.99 & 2.50 \\
88.7 & 5.54 & 2.77 \\
96.3 & 6.02 & 3.01 \\
104 & 6.53 & 3.26 \\
112 & 6.98 & 3.49 \\
119 & 7.42 & 3.71 \\
132 & 8.22 & 4.11 \\
144 & 7.42 & 4.48 \\
164 & 10.3 & 5.14 \\
182 & 11.4 & 5.69 \\
\hline
\end{array}
\]

Table 5.3.1: Table showing the Reynolds numbers tested in the tube bundle test case and the corresponding cell Peclet numbers.

The velocity streamlines are shown in Figure 5.3.4 for \( Re = 88.7 \) and \( Re = 182 \). At \( Re = 88.7 \) there is a small recirculation region downstream of the cylinders. This increases in size as the Reynolds number increases and by \( Re = 182 \) it is approximately one cylinder radius in size. Above \( Re = 182 \) the velocity solution begins to pulse and oscillate; a steady solution is unable to be found. For this reason the testing of the error estimate method was not investigated above \( Re = 182 \).
Figure 5.3.4: Examples of the velocity streamlines for the tube bundles test case. The scale was normalised by $U_b$.

Figure 5.3.5 shows the velocity magnitude and total kinetic energy found using the three mesh densities at $Re = 144$. The kinetic energy is located along the centre of the flow, with peak of energy close to where the fluid impinges on the cylinder walls. The magnitude here is much larger than the values seen in the recirculation bubbles.

Figure 5.1.5 shows the convergence of the $U$ and $K$ simulations with $Re = 144$ on the coarse mesh, displayed as the volume average of $U$ and $K$ divided by the final volume averaged value they achieve against iteration number. The $K$ convergence initially lags behind the velocity simulation, however, the two reach convergence after approximately the same number of iterations.

The convergence of the tube bundle simulation with $Re = 144$ on the coarse mesh, displayed as the volume averaged velocity, $\langle U \rangle$, and total kinetic energy, $\langle K \rangle$, divided by their respective final values, against iteration number.
Figure 5.3.5: Example of the total kinetic energy solution for the tube bundles test case. The scale was normalised by $\frac{U^2}{2}$.

Figure 5.3.6: The convergence of the tube bundle simulation with $Re = 144$ on the coarse mesh, displayed as the volume averaged velocity, $\langle U \rangle$, and total kinetic energy, $\langle K \rangle$, divided by their respective final values, against iteration number.
5.3.2 Scalar Solution in Laminar Velocity Solution

The fully developed periodic flow solutions for each mesh were used to initialise the scalar simulations. This frozen velocity, one that is not transported, keeps the velocity field the same as the fully developed periodic flow. This is necessary because the scalar will not reach a steady state in a periodic simulation. The investigation simulates the transport of a passive scalar into this fully developed flow from a single ‘hot’ cylinder out of an infinite array of ‘cold’ ones. Using the three discretisation schemes the first and second moments were solved to find $f$ and $q$. The values of both scalars were set equal to zero at the inlet. The left and right cylinder walls both had flux equal to zero.

For the constant flux boundary condition simulations the flux of $f$ through the centre cylinder wall was set as $10[f]m^{-1}$. The flux of $q$ through this wall was set to $10\frac{\partial f}{\partial x'}[f]^2m^{-1}$ where $x'$ is the wall normal direction. For the constant value boundary condition simulation the value of $f$ at the centre cylinder wall was set as $1[f]$. The value of $q$ at these walls was set as $0.5[f]^2$.

The $f$ distributions found using the three mesh densities and both boundary types are shown in Figure 5.3.7 for $Re = 144$. For the constant flux result the highest values of $f$ are behind the centre cylinder where there is the least fluid convection. Comparing to the constant value boundary result, the walls are held at the highest value and so there is a high value round the entire cylinder as well as high values in the recirculation region. As the Reynolds number is increased the amount of cooling will increase, these regions will become smaller, and the gradients next to the wall in the constant value simulations increase.

The $q$ solutions found using the three mesh densities and both boundary types are shown in Figure 5.3.8 for $Re = 144$. The features are similar to the $f$ solution. The gradients of the scalar are high next to the walls in the constant value boundary simulation.

Figure 5.3.9 shows the convergence of the $f$ and $q$ simulations with $Re = 144$ on the coarse mesh for both boundary condition types, displayed as the volume average of $f$ and $q$ divided by the final volume averaged value they achieve against iteration number. In the constant value boundary simulation the $f$ and $q$ simulations converge together with the $q$ convergence lagging behind slightly, and they both converge faster than in the constant flux simulation. In the constant flux boundary simulation the $q$ convergence initially lags behind the $f$ simulation, and reaches convergence later than the $f$ simulation.
Figure 5.3.7: The $f$ solutions for $Re = 144$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the central cylinder wall.
Figure 5.3.8: The $q$ solutions for $Re = 144$ found using the three mesh densities with constant flux (left) and constant value (right) boundary conditions along the central cylinder wall.

Figure 5.3.9: The convergence of the tube bundle simulation with $Re = 144$ on the coarse mesh, displayed as the volume averaged $\langle f \rangle$ and $\langle q \rangle$, divided by their respective final values, against iteration number.
5.4 Chapter 5 Summary

Chapter 5 outlined the three main test cases; the ribbed channel, the impinging flow, and the hexagonally stacked tube bundles. Each test case was given a description of the geometry, the meshes, the Reynolds numbers tested, and the boundary conditions used in the investigations. To give the reader an idea of the flow fields examples of the velocity and scalar distributions were given with brief descriptions, mentioning how they develop with changing Reynolds number. This chapter contains all the information required to set up the test cases so that additional definitions are not required in the following chapters. The next two chapters investigate the proposed error estimation method when applied to the transport of a scalar in a laminar fluid flow, and when applied to the transport of the fluid at Reynolds numbers within the laminar regime. First is chapter 6, which is about the scalar error estimate, and it is divided into two sections. These present separately the results found using the two different boundary condition types.
Chapter 6

Scalar Error Estimate in Laminar Fluid Flow

Chapter 6 presents the results for the scalar error estimate in laminar fluid flow. The real error distributions were found by the difference between the coarse Code_Saturne solutions and the refined mesh Code_Saturne solution. The value in each coarser mesh cell was subtracted from the value of the refined solution at the location of the coarser mesh cell center, and the absolute value was taken. The estimate was calculated according to Equation 3.2.27.

The solving of the additional equations will increase the computation time. This increase will be monitored by comparing the time it takes to complete the simulations with and without the simulation to find the error estimate. The comparisons will be made by expressing the increased computation time as a percentage of the computation time without the method.

This chapter is separated into three sections, the first presents the results using the constant flux boundary condition, the second presents the results using the constant value boundary condition, and the third is a discussion of conclusions that can be drawn from the results about how the method performs.

The two results sections are laid out similarly. First they present examples of the $f$ error distributions alongside corresponding error estimate distributions. Qualitative descriptions are given of these. Next the correlation coefficients, calculated using Equation 3.2.34, are presented to judge how well the shape has been predicted. To give context to these numbers, examples of high and low correlation coefficient results are shown for qualitative comparison. Finally the averaged summation quantities for the error and estimate, calculated using Equations 3.2.32 and 3.2.33, are presented to judge the scale prediction.

The discussion section explains the trends seen in the results and discusses the limitations of the method when applied to the transport of a scalar in a laminar fluid flow set by the boundary type, near-wall refinement and cell Peclet number. The final part of this section performs the Richardson extrapolation procedure on the impinging flow test case results and makes comparisons between the performances of the two method.
6.1 Constant Flux Boundary Results

6.1.1 Constant Flux Boundary $f$ Error and Error Estimate Qualitative Comparisons

This section will present some examples of the error distributions and their corresponding estimate distributions for the three test cases using the constant flux boundary condition. In general, each discretisation scheme produces slightly different sets of features in the error and estimate distributions. Typically, the results that use the first order scheme produced distributions different from the more similar results that use either of the two second order schemes. For each test case two sets of error and estimate distributions are presented, giving an example from the first order simulations and an example from the second order simulations. The physical characteristics observed in the distributions using the two orders of discretisation are discussed in general, referring to the given examples only to illustrate the comparisons.

6.1.1.1 Ribbed Channel

Figure 6.1.1 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 18% computation time after 150 iterations, on a system with an Intel(R) Core(TM) i7 CPU 870 at 2.93GHz, with 4GB RAM and running openSUSE Linux. Typically, high error is found around both ribs, with less error found in-between them. Behind the upstream rib the error coincides with the recirculation region. Around the downstream rib the error coincides with the impingement of the fluid on to the protruding wall. The error does not follow the path of the fluid. Zero error bands are seen around the downstream rib, where the difference between the refined and numerical solution changes sign. The same pattern is seen in the estimate, except without the banding. In the example shown, it can be seen that the error behind the upstream rib is similar in scale to the error around the downstream rib. However, in the estimate there is a relative over-prediction of the error downstream. This is the case in many of the constant flux boundary simulations.

![Figure 6.1.1: An example of the $f$ error and estimate from the ribbed channel with constant flux boundary simulations. They were found with $Re = 175$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.82.](image)
Figure 6.1.2 is an example from the simulations using the second order centred discretisation scheme. Typically, high error is found behind the upstream rib in the recirculation region, with low error found around the downstream rib and in-between the two ribs. This pattern is visible in the estimate predictions. In the example shown the downstream error is higher relative to the upstream error than predicted by the estimate.

Figure 6.1.2: An example of the $f$ error and estimate from the ribbed channel with constant flux boundary simulations. They were found with $Re = 175$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.89.

6.1.1.2 Impinging Flow

Figure 6.1.3 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 10% computation time after 100 iterations. The highest error is found next to the impingement point where the fluid is most stressed, with lower, but still significant, error predicted further downstream across the channel width following the path of the fluid. In the given example there is also a band of zero error in the stream-wise direction. The same features are seen in the estimate, but without the zero-band. However, moving downstream the estimate generates an over-prediction. This has caused the estimate prediction near the impingement point to be over-shadowed by the excessively high prediction downstream. The over-prediction moving downstream is a feature seen in many of the constant flux simulations.

Figure 6.1.3: An example of the $f$ error and estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.44.

Figure 6.1.4 is an example from the simulations using the second order centred discretisation scheme.
There is high error found next to the impingement point, with low error found downstream similar to the first order results. This pattern is also seen in the estimate. Again, moving downstream the estimate becomes a relative over-prediction compared to the impingement point.

![Image of error and estimate](image1.png)

Figure 6.1.4: An example of the $f$ error and estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.53.

### 6.1.1.3 Hexagonally Stacked Tube Bundles

Figure 6.1.5 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 20% computation time after 200 iterations. There is one main region of high error behind the centre cylinder coinciding with the recirculation region. The error does not follow the path of the fluid. There is also very low error found around the rest of the centre cylinder. These same features are visible in the estimate.

![Image of error and estimate](image2.png)

Figure 6.1.5: An example of the $f$ error and estimate from the tube bundles with constant flux boundary simulations. They were found with $Re = 79.9$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.91.

Figure 6.1.6 is an example from the simulations using the second order centred discretisation scheme. The errors show the same features as the first order results with a main region of high error behind the centre cylinder coinciding with the recirculation region. There is also a band of zero error cutting through this region. The estimate predicts high error behind the centre cylinder, however, the shape does not match exactly due to the lack of prediction of the zero-band. The estimate also predicts a more significant amount of error around the rest of the centre cylinder wall that is not seen in the error distributions.
6.1.2 Constant Flux Boundary Correlation Coefficients

How well the shape has been predicted is judged by the correlation coefficient. This section will present the coefficients for the three test cases plotted as correlation coefficient against Reynolds number. For each test case two examples are given, one each for a high and a low correlation coefficient, to give insight into the meaning of these values.

6.1.2.1 Ribbed Channel

Figure 6.1.7 the correlation coefficients are shown for the different simulations. In general, on both meshes the correlation coefficients are high, between 0.8 and 0.9, with a drop in value at low Reynolds number below $Re = 100$. The main differences are that the first order coefficients on the coarse mesh are 10% lower than the second order coefficients, and that the second order centred coefficients on the very coarse mesh begin to decrease with increasing Reynolds number, dropping to as low as 0.6.

Figure 6.1.8 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order centred scheme at $Re = 1640$, which produced the lowest correlation coefficient of 0.63. The main feature of the errors after the upstream rib is seen, but its significance is reduced by a large over-prediction of the error near the downstream rib. This is typical of the error estimates that produce low correlation coefficients in this test case and with the constant flux boundary condition. As the Reynolds number increases this over-prediction increases and the correlation decreases. In the discussion, this effect will be attributed to a combination of unboundedness of the solutions and the poor prediction of gradients in the boundary layer.

The second example is for the coarse mesh with the second order centred scheme at $Re = 674$, which produced the highest correlation coefficient of 0.94. The main features of the error distribution have been predicted. Slight differences arise in the shape of the main regions of error and the zero error/estimate bands. On the coarse mesh the over-prediction around the downstream rib does not develop with changing Reynolds number, indicating a dependence on the cell size.

Figure 6.1.6: An example of the error and estimate from the tube bundles with constant flux boundary simulations. They were found with $Re = 79.9$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.51.
Figure 6.1.7: Correlation coefficients plotted against $Re$ for the ribbed channel with constant flux boundary simulations and using three discretisation schemes; FO, SOC and SOLU.
Figure 6.1.8: Two examples of the error and estimate distributions for the ribbed channel with constant flux boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

### 6.1.2.2 Impinging Flow

In Figure 6.1.9 the correlation coefficients are shown for the different simulations. For both meshes at Reynolds numbers lower than $Re = 500$, the correlation coefficients are low producing values between 0.4 and 0.6. At higher Reynolds numbers the correlation coefficients increase. Both the first order upwind and second order centred results reach correlation coefficients between 0.8 and 0.9, while the second order linear upwind scheme coefficients remain low at approximately 0.6.

Figure 6.1.10 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order linear upwind scheme at $Re = 1000$, which produced a correlation coefficient of 0.57. The most prominent feature in the error distribution is the peak of error along the lower wall. In the error estimate this feature is seen a few cells away. With this scheme, as the Reynolds number increases, the location of the peak in the error distributions shifts downstream. In the estimate distribution all along the lower wall there is high estimate, so the prediction never matches exactly to the errors. The effect is the lowering of the correlation coefficients for this discretisation scheme. In the discussion this effect is attributed to the gradient estimation in the first cells next to the walls.

The second example is for the coarse mesh with the first order upwind scheme at $Re = 2500$, which produced the highest correlation coefficient of 0.89. The main feature of high error along the wall
and downstream is predicted in the estimate distribution. However, there is a band of zero estimate running through the estimate distribution, which is not seen in the errors. There is also over-prediction in the first cells along the wall at the impingement point, similar to the previous example.

Figure 6.1.9: Correlation coefficients plotted against $Re$ for the impinging flow with constant flux boundary simulations and using three discretisation schemes; FO, SOC and SOLU.
### 6.1.2.3 Hexagonally Stacked Tube Bundles

In Figure 6.1.11 the correlation coefficients are shown for the different simulations. The results on both of the meshes are similar; the refinement has nearly no effect on the correlation coefficients. The first order coefficients are highest at 0.9 and after $Re = 144$ they begin to decrease with increasing Reynolds number. The second order results are lower, with the second order linear upwind scheme producing coefficients of 0.6 and the centred scheme producing coefficients between 0.4 and 0.5. The coefficients from both second order schemes decrease with increasing Reynolds number until $Re = 144$ and then begin to increase again. If it were possible to test higher Reynolds numbers it would be informative to investigate whether the results show a pattern similar to the ribbed channel and impinging flow results up to $Re = 2000$.

Figure 6.1.12 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order centred scheme at $Re = 164$, which produced a correlation coefficient of 0.32. The causes for the low correlation coefficient are the zero error/estimate bands seen behind the centre cylinder, and the over-prediction around the upstream side of the centre cylinder. The zero-bands are present for all Reynolds numbers. The same is seen for the second order linear upwind scheme. The upstream over-prediction is also seen across all Reynolds numbers. These issues are revisited in the discussion section.

The second example is for the coarse mesh with the first order upwind scheme at $Re = 119$, which produced the highest correlation coefficient of 0.92. Although there is a slight difference in shape between the two, the region behind the centre cylinder is predicted across all Reynolds numbers. The zero error band is not predicted but the correlation coefficients remain consistently high.
Figure 6.1.11: Correlation coefficients plotted against $Re$ for the tube bundles with constant flux boundary simulations and using three discretisation schemes; FO, SOC and SOLU.
6.1.3 Summation Coefficients

How well the scale has been predicted is judged by the summation coefficients. In this section the summation coefficients are presented as graphs of average error against average estimate for each test case. Frequently the profiles of error against estimate show a linear section at low Reynolds numbers and then after a certain critical point they diverge and the estimate stops increasing while the error continues to do so. This leads to a single value of the estimate implying multiple values of the error. This behaviour, which will be explained further in the discussion section, is attributed to the boundary layer not being fully resolved by the first few rows of cells.

6.1.3.1 Ribbed Channel

Figure 6.1.13 shows the average error against average estimate for the three discretisation schemes and the two meshes. In each of the graphs, the error and estimate increase together as a result of increasing the Reynolds number. At low Reynolds numbers the scale of the estimate is a close approximation to the scale of the error with an over-prediction by a factor of 1.5. As the Reynolds number increases the estimate scale begins to under predict the error scale. The estimate scale has slowed its increase while the error scale continues to rise.

For example, consider the results for the first order scheme on the very coarse mesh: after the seventh data point the error vs estimate profile diverges from the very linear behaviour seen before the seventh data point. The gradients of the profiles are still positive, although they are now steeper. Therefore there is a breakdown of the estimate scale prediction above a certain point. This is similar to
the behaviour observed in the point source in a cross flow test case. This causes a problem for predicting accurately the scale errors. A single value of the estimate after the breakdown could indicate multiple possible values for the error. This issue will be revisited in the discussion section.

Figure 6.1.13: Plots of the summation coefficients for the ribbed channel with constant flux boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

6.1.3.2 Impinging Flow

Figure 6.1.14 shows the averaged error against the averaged estimate for the three discretisation schemes and the two meshes. Each of the data sets show that, up to a certain point, the estimate increases linearly with the error as the Reynolds number is increased. After this there is a breakdown and the estimate slows its increase while the error continues to rise, similar to the ribbed channel flow coefficients. In the linear section the estimate is over-predicting the error by a factor of 2. After this
the gradient is steeper but the two still increase together.

![Averaged Error against Averaged Estimate for the Very Coarse Mesh](image1)

![Averaged Error against Averaged Estimate for the Coarse Mesh](image2)

Figure 6.1.14: Plots of the summation coefficients for the impinging flow with constant flux boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

6.1.3.3 Hexagonally Stacked Tube Bundles

Figure 6.1.15 shows the average error against average estimate for the three discretisation schemes and the two meshes. A difference between the first and second order schemes is seen which was not observed in the previous test cases. The first order profiles are linear with increasing Reynolds number, where the error is under-predicted by a factor between 2 and 3. The second order profiles are separated from the first order ones and do not collapse together in the same way as was seen in the previous test cases. They also show evidence of the breakdown. Before the breakdown the error is under-predicted by a factor of between 2 and 3. This is unlike the previous test cases that typically over-predicted
the error in the linear region. It indicates that the Reynolds numbers tested were already past the breakdown point. Due to time constraints further error analysis was not possible, but the issue can still be investigated using only estimate predictions, and will be revisited in the discussion section.

Figure 6.1.15: Plots of the summation coefficients for the tube bundles with constant flux boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.  

![Average Error against Average Estimate for the Very Coarse Mesh](image)

![Average Error against Average Estimate for the Coarse Mesh](image)
6.2 Constant Value Boundary Results

6.2.1 Constant Value Boundary $f$ Error and Error Estimate Qualitative Comparisons

This section will present some examples of the error distributions and their corresponding estimate distributions for the three test cases using the constant value boundary condition. In general, each discretisation scheme produces slightly different sets of features in the error and estimate distributions. Typically, the results that use the first order scheme produced distributions different from the more similar results that use either of the two second order schemes. For each test case two sets of error and estimate distributions are presented, giving an example from the first order simulations and an example from the second order simulations. The physical characteristics observed in the distributions using the two orders of discretisation are discussed in general, referring to the given examples only to illustrate the comparisons.

6.2.1.1 Ribbed Channel

Figure 6.2.1 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 11% computation time after 150 iterations. The main source of high error is found in a small region located next to the corner of the upstream rib. This does not coincide with any features of the fluid flow. There is also low error around the downstream rib which increases in size and value with the Reynolds number. The same features are seen in the estimate distribution, although the exact location of the upstream peak of estimate does not match exactly to the location of the peak in the errors. In the example, the prediction of the maximum estimate value is two cells upstream from the maximum error. This is consistent across the Reynolds numbers tested.

![Figure 6.2.1](image)

Figure 6.2.1: An example of the $f$ error and estimate from the ribbed channel with constant value boundary simulations. They were found with $Re = 175$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.62.

Figure 6.2.2 is an example from the simulations using the second order centred discretisation scheme. The second order results are similar to the first order results. There is a source of high error found in a small region next to the corner of the upstream rib, with lower error around the downstream rib.
that increases with Reynolds number. The estimate predicts the main source of high error, although the exact location of the upstream peak of estimate does not match exactly to the location of the peak in the errors. In the example the error predicts the maximum two cells upstream. Again, this is consistent across the Reynolds numbers tested. As Reynolds number increases, extraneous estimate prediction develops in the very coarse mesh simulations along the first row of cells next to the heated walls (seen in Figure 6.1.8 (right)). This is discussed in the correlation coefficient section.

![Figure 6.2.2: An example of the $f$ error and estimate from the ribbed channel with constant value boundary simulations. They were found with $Re = 175$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.58.](image)

### 6.2.1.2 Impinging Flow

Figure 6.2.3 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 8% computation time after 100 iterations. The region of high error starts near the impingement point and spreads downstream and upwards towards the upper wall, following the path of the fluid. This pattern is seen in the estimate. Comparing this to the constant flux boundary results, the estimate does not become a large over-prediction when moving further downstream.

![Figure 6.2.3: An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.74.](image)

Figure 6.2.4 is an example from the simulations using the second order centred discretisation scheme. High error is found close to the impingement point where the fluid is under the most stress,
with low error spreading downstream. Although in the estimate distribution there is high estimate at the impingement point, the shape does not match well to the errors. A feature that is seen across the Reynolds numbers tested is that the first cells along the lower wall are often extremely over-predicted relative to the rest of the distribution. This may be shadowing the correct prediction of the other features in the error distribution. The explanation for this will be revisited in the discussion section.

Figure 6.2.4: An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.42.

### 6.2.1.3 Hexagonally Stacked Tube Bundles

Figure 6.2.5 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 11% computation time after 200 iterations. Similar to the constant flux boundary results, there is a main region of high error located behind the centre cylinder coinciding with the recirculation region. A difference is that there is lower, but not insignificant, error seen around the rest of the cylinder that follows the path of the fluid. There is a band of zero error seen, following the path of the fluid flow. The estimate predicts these same features except for the zero-band.

Figure 6.2.5: An example of the $f$ error and estimate from the tube bundles with constant value boundary simulations. They were found with $Re = 79.9$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.65.

Figure 6.2.6 is an example from the simulations using the second order centred discretisation scheme. A main region of high error is seen downstream of the centre cylinder with a zero error band cutting through the recirculation region. There is also lower, but not insignificant, error seen around the rest of the cylinder that follows the path of the fluid starting from the high stress region at the
impingement point. The estimate does not predict these features. Similar to the impinging flow test case with second order schemes, there is a large over-prediction relative to the rest of the distribution across the first row of cells where the fluid impinges on the upstream side of the cylinder. As a result of this no other features are seen. This appears to be a problem associated with the second order schemes and will be addressed in the discussion section.

Figure 6.2.6: An example of the $f$ error and estimate from the tube bundles with constant value boundary simulations. They were found with $Re = 79.9$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.15.

### 6.2.2 Constant Value Boundary Correlation Coefficients

How well the shape has been predicted is judged by the correlation coefficient. This section will present the coefficients for the three test cases plotted as correlation coefficient against Reynolds number. For each test case two examples are given, one each for a high and a low correlation coefficient, to give insight into the meaning of these values.

#### 6.2.2.1 Ribbed Channel

In Figure 6.2.7 the correlation coefficients are shown for the different simulations. On both of the meshes the first order results were found to be similar. At low Reynolds number the correlation coefficients are low, and as $Re$ increases the coefficients increase to plateau between 0.7 and 0.8. Differences are seen between the two meshes for the second order results. The coefficients on the coarse mesh remain higher at between 0.6 and 0.7, while on the very coarse mesh the coefficients drop as the Reynolds number increase until they plateau 0.4.

Figure 6.2.8 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order centred scheme at $Re = 344$, which produced a correlation coefficient of 0.29. In this example the reason the coefficient is so low can be seen; in the estimate there is a large prediction of error along the first cells against the heated wall that is not seen in the error. This is incorrect because the error is expected to go to zero next to the constant value walls. For both second order schemes using the very coarse mesh this is seen across the range of Reynolds numbers. The cause of this will be revisited in the discussion section.

The second example is for the coarse mesh with the first order upwind scheme at $Re = 1640$, which produced the highest correlation coefficient of 0.79. At low $Re$ the high error is located in a very small region next to the upstream rib, as seen in Figure 6.2.1. As $Re$ increases the error spreads in-between
the ribs, as seen in the example in Figure 6.2.1. This is reflected in the estimate. The correlation coefficients are sensitive to slight changes of location of the peaks when the regions are small. When they increase in size the slight differences are less important and lead to the coefficient increasing, explaining the coefficient profile seen. The example shows a slight relative over-prediction near the downstream rib.

![Correlation Coefficients against Reynolds Number for the Very Coarse Mesh](image)

![Correlation Coefficients against Reynolds Number for the Coarse Mesh](image)

Figure 6.2.7: Correlation coefficients plotted against $Re$ for the ribbed channel with constant value boundary simulations and using three discretisation schemes; FO, SOC and SOLU.
6.2.2.2 Impinging Flow

In Figure 6.2.9 the correlation coefficients are shown for the different simulations. On both meshes the first order correlation coefficients are high and remain level. The coefficients on the very coarse mesh results are between 0.7 and 0.8, and on the coarse mesh the values are lower, between 0.6 and 0.7. The second order scheme profiles show similarly shaped profiles to each other and on both meshes. At low Reynolds numbers below $Re = 500$ the coefficients are low. They increase and become level as $Re$ is increased. On the very coarse mesh they plateau between 0.5 and 0.6, which is lower than the first order coefficients. On the coarse mesh they plateau between 0.6 and 0.7, which is higher than the first order coefficients.

Figure 6.2.10 shows two examples of the error and estimate distribution. The first is for the coarse mesh with the second order centred scheme at $Re = 125$, which produced a correlation coefficient of 0.060. Two features affect the correlation coefficient in this example. First, at this Reynolds number, the general error levels have become low enough that the error in the cells adjacent to the corner of the intersection have become significant. This is not seen in the estimate. Second, in the estimate there is a large over-prediction in the cells adjacent to the impingement point. This is similar to the behaviour observed in the ribbed channel test case. In this case it has obscured any other features that were predicted, and it does not resemble the errors. As the Reynolds number increases the error
levels rise so the corner error becomes less significant compared to the rest of the distribution. The shape that develops has a main region of error next to the impingement point, as seen in Figure 6.2.4. The correlation coefficient increases because the error peak and incorrect estimate peak are in the same approximate region, rather than becoming a better prediction. This issue will be revisited in the discussion section.

The second example is for the very coarse mesh with the first order upwind scheme at $Re = 2500$, which produced the highest correlation coefficient of 0.79. In this set of data the peak in the error along the wall is predicted as well as the additional error spreading out from it. As the Reynolds number changes the location of the peak in error shifts and the estimate reflects this.

![Correlation Coefficients against Reynolds Number for the Very Coarse Mesh](image)

![Correlation Coefficients against Reynolds Number for the Coarse Mesh](image)

Figure 6.2.9: Correlation coefficients plotted against $Re$ for the impinging flow with constant value boundary simulations and using three discretisation schemes; FO, SOC and SOLU.
Figure 6.2.10: Two examples of the error and estimate distributions for the impinging flow with constant value boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.2.2.3 Hexagonally Stacked Tube Bundles

In Figure 6.2.11 the correlation coefficients are shown for the different simulations. A comparison of the correlation coefficients between the two meshes show them to be similar, except that the second order coefficients are 30% lower on the very coarse mesh. The three schemes produce profiles that increase in correlation coefficient with increasing Reynolds number. However, the second order coefficients are significantly lower than the first order. The first order coefficients are between 0.7 and 0.8 and the second order coefficients are between 0.1 and 0.2.

Figure 6.2.12 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order centred scheme at $Re = 79.9$, which produced a correlation coefficient of 0.096. In this example there is a relatively high prediction of error in the estimate distribution along the upstream wall of the centre cylinder that is not seen in the error distribution. This over shadows any other features that were predicted and therefore the correlation coefficients are very low. This is similar to the behaviour observed in the ribbed channel and impinging flow with constant value boundary test cases. This appears to be a common issue with the second order schemes and constant value boundary condition, and will be revisited in the discussion section.

The second example is for the coarse mesh with the first order upwind scheme at $Re = 182$, which produced the highest correlation coefficient of 0.84. The estimate predicts the high error behind the centre cylinder as well as the lower error spreading around the rest of the centre cylinder. It does not predict the zero error band that follows the path of the fluid flow.
Figure 6.2.11: Correlation coefficients plotted against $Re$ for the tube bundles with constant value boundary simulations and using three discretisation schemes; FO, SOC and SOLU.
Figure 6.2.12: Two examples of the error and estimate distributions for the tube bundles with constant value boundary simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

6.2.3 Constant Value Boundary Summation Coefficients

The prediction performance of the scale is assessed using the summation coefficients. In this section the summation coefficients are presented as graphs of average error against average estimate for each test case. Frequently the profiles of error against estimate show a linear section at low Reynolds numbers and then after a certain critical point diverge and the estimate stops increasing while the error continues to do so. This leads to a single value of the estimate implying multiple values of the error. This behaviour, which will be explained further in the discussion section, is attributed to the boundary layer not being fully resolved by the first few rows of cells.

6.2.3.1 Ribbed Channel

Figure 6.2.13 shows the average error against average estimate for the three discretisation schemes and the two meshes. The first order profiles are initially linear, over-predicting the error by a factor of 1.5. Then the breakdown occurs and the estimate slows its increase with Reynolds number. The second order results are less clear. On the very coarse mesh, before the breakdown, both second order schemes produce a linear profile that matches the first order result. After the breakdown the values of the estimate become scattered and stop showing a simple relationship to the error. On the coarse mesh the relationship between error and estimate is unclear. While the error increases with Reynolds number the values of the estimate are scattered.
Figure 6.2.13: Plots of the summation coefficients for the ribbed channel with constant value boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

6.2.3.2 Impinging Flow

Figure 6.2.14 shows the averaged error against the averaged estimate for the three discretisation schemes and the two meshes. Each of the profiles increase linearly together up until a certain Reynolds number where the breakdown occurs. Before the breakdown the estimate is over estimating the errors by a factor of 2. After the breakdown the estimate slows its increase with Reynolds number while the error continues to increase. For example, the first order profile on the very coarse mesh diverges from the linear relationship after the fifth data point. One data set, the second order linear upwind on the very coarse mesh, displays a speeding up of the increase of the estimate with Reynolds number.
Figure 6.2.14: Plots of the summation coefficients for the impinging flow with constant value boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

6.2.3.3 Hexagonally Stacked Tube Bundles

Figure 6.2.15 shows the averaged error against the averaged estimate for the three discretisation schemes and the two meshes. The profiles seen are similar to the ones for the constant flux simulations. The first order profiles are linear, increasing in error and estimate with increasing Reynolds number. The scales of the estimate are a close match to the scales of the error. The second order profiles show a steeper gradient, with the estimate scale under-predicting the error scale by up to a factor of 10. Evidence of the breakdown is visible. On the very coarse mesh the estimate becomes scattered, while the error continues to increase with Reynolds number.
Figure 6.2.15: Plots of the summation coefficients for the tube bundles with constant value boundary simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. 
6.3 Discussion of the Scalar in Laminar Fluid Flow Error Estimate

The results show that the proposed method is able to predict the numerical errors to a degree of accuracy that, in general, allows identification of the important regions of error and their approximate scale. The method, when used with the first order scheme, consistently produced the best estimations of the shape of the errors. Under certain circumstances the method was also able to make good estimations of the shape using the second order schemes. Some limitations have presented themselves, which will be discussed in this section.

In some simulations, certain features seen in the error distributions were not predicted by the estimate, while in other simulations there are features predicted extraneously in the estimate distributions where peaks of estimate are seen that are not present in the errors. ‘Zero error bands’ were often not predicted, for example the first order upwind impinging flow with coarse mesh, constant flux boundary and $Re = 375$ simulation seen in Figure 6.1.3. For some of the second order simulations any predictions of the errors were over-shadowed by the extraneous features, for example the second order centred impinging flow test case with constant value boundary and $Re = 375$ seen in Figure 6.2.4. This has the effect of reducing the correlation coefficients. The explanation for this is a combination of different factors, including:

1. The development of unboundedness, which is observed in some of the first and second moment solutions using the second order schemes.

2. The boundary layer not being resolved causing the error to accumulate in the SME. In turn the estimate degrades, affecting both scheme orders and boundary condition types.

3. The near-wall estimation of the cell centre gradients being worse for the constant value boundary condition. This leads to additional source term error for the SME.

At low values of $Re$ (or $Re_L$) the estimation of the scale is consistently within an order of magnitude of the error scale. Under these conditions the relationship between the error and estimate scales is linear, and they are close in value; typically the estimate is an over-prediction by a factor of 3 or less. However, for all schemes, a ‘breakdown’ of the estimate scale prediction was seen. As the Reynolds number is increased the estimate increase slows, while the error continues to rise. This section will discuss and compare the strengths and weaknesses of the method further, and present explanations for the limitations observed.

6.3.1 Zero Error Bands

Zero error bands were observed in the preliminary investigations, and they are seen again in the main test cases. For example, the first order upwind tube bundles simulation with the coarse mesh, constant value boundary and $Re = 79.9$ seen in Figure 6.2.5. These occur when the fine grid solution is subtracted from the coarser grid solutions (or when the second moment solution is subtracted from the squared first moment solution) and the result changes sign from positive to negative. It is the magnitude of these differences that is shown in the colour map distributions above, and so what is
seen is a band of zero error. These are rarely predicted in the estimate distributions, which causes them to be visually less similar to the error distributions. Due to their not being predicted this will decrease the correlation coefficient.

Another effect of a zero-band is a decrease in the error summation coefficient relative to the estimate coefficient. This is because a number of the cells included in the averaging are at a lower value, while the estimate often ignores these zero-bands so the averages remain high. These will be partially responsible for a number of the simulations’ estimate scales being predicted larger than their corresponding error scales. For example in Figure 6.1.3 the maximum values in the distributions were close in value, however, the estimate summation coefficient was a factor of 3 higher than the error coefficient.

Despite the issues related to the coefficients, it is not necessarily a drawback for the method to predict error in the location of these zero-bands. An arbitrary set of simulations, in which the Reynolds number is varied, will produce a set of error distributions that have similar features. The shape of the features will develop as $Re$ changes, and any zero error bands that are present can shift position. Therefore in a specific simulation from this arbitrary set the zero-band is only zero along a certain line coincidentally. In actuality these regions are capable of generating error. For a user, in the circumstances this method aims to address, this is a benefit. Additional information is revealed that a grid refinement study at a single $Re$ would not have given, information that would be applicable over the range of Reynolds numbers.

### 6.3.2 Unboundedness in Solutions

The first contribution to the extraneous features that were predicted is unboundedness in the first and second moment solutions. The evidence for this can be visually seen in the figures already presented. Considering Figures 6.1.8 (left) and 6.2.8 (left), it can be seen that the error appears to be ‘checkerboarding’ in the centre of the channel [37]. This is where the numerical solution oscillates around the exact solution. It is caused by the rapid change of the exact solution gradient across a small number of cells. Combining this with the use of the second order scheme, in which adjacent cells are not coupled, allows the numerical solution to repeatedly over- and under-shoot. Figure 6.3.1 is a profile of the error across the second order centred ribbed channel simulation with the very coarse mesh, constant flux boundary and $Re = 1640$ (a colour map distribution can be seen in Figures 6.1.8). The profile runs from the top of the upstream rib to the top of the downstream rib. It shows that the error distribution loses the gently changing profile after $x = 28$ and develops the oscillation pattern typical of unboundedness.

This, in part, explains why the second order centred scheme frequently produced the worst shape predictions. The second order linear upwind scheme will also be affected, but to a lesser degree than the centred scheme. The first order upwind scheme will not be affected by this and this is reflected in the higher correlation coefficients that were consistently seen. The conclusion is that the shape prediction using the second order centred scheme will have a dependence on the cell Peclet number, which in standard practice means that above $Pe_L = 2$ the estimate may begin to degrade.
Figure 6.3.1: A profile of the error from the second order centred ribbed channel simulation using the very coarse mesh, constant flux boundary and $Re = 1640$. The profile ran from the top of the upstream rib to the top of the downstream rib.

### 6.3.3 Capturing of the boundary layer

A second contribution to the extraneous features results from the boundary layer not being resolved by the near-wall cells. The boundary layer is a region close to the walls that has large changes of gradients of the velocities and scalars. Increasing the Reynolds number decreases the size of the boundary layer. Similarly, in the case of impingement the fluid is impacting the wall at high velocity and being deflected, causing the boundary layer to further decrease in size. The work presented here used coarse meshes in order to generate significant error. This means it is possible for a single row of cells to completely encompass the boundary layer.

Figure 6.3.2 shows a diagram of the first row of cells next to a heated wall to give a simplistic explanation of the effect to aid the discussion. It also shows a superimposed graph of the value of a solved variable against the wall normal distance with the boundary layer completely within the first row of cells. It shows that where large changes in gradient of the solved for variable are present the interpolation across the cell leads to a larger error. Therefore if the near-wall behaviour is not resolved an error is introduced into the calculation of the first moment. These errors may be small and not visible in the error distributions, however, they will be magnified in the SME via the second moment source term, which will be discussed in the next section.

Again, the boundary layer may not be resolved, introducing more error into the solution process, these errors may still be small and not visible in the error distributions. The noticeable effect is to increase the difference between the first and second moment solutions near the wall. This is visible in the estimate shape prediction.
An example of this occurring is the second order centred tube bundles simulation with very coarse mesh, constant flux boundary and $Re = 164$ seen in 6.1.12 (left). There is a high value of estimate seen around the upstream side of the heated cylinder where there is fluid impingement which is not seen in the error distribution. Another example is the second order centred impinging flow with coarse mesh, constant value boundary and $Re = 375$ seen in Figure 6.2.4. The over-prediction next to the impingement point is so severe that no other features can be seen. As a result, the correlation coefficients are very low. The effect will also be partially responsible for incorrect scale predictions through a mechanism that will be discussed in section 6.3.5.

The effect will occur when using both scheme orders, for example the first order upwind impinging flow with coarse mesh, constant flux boundary and $Re = 2500$ seen in Figure 6.1.10 (right). However, it is typically more visible when using the second order schemes compared to first order. It is significantly more severe when using the constant value boundary condition compared to constant flux, to the extent that no other features except the near-wall over-prediction can be seen. This is due to an additional contribution related to the gradient approximation for the SME source term.

### 6.3.4 Gradient Approximation in Near-Wall Cells

The third contribution is introduced via the calculation of the SME source terms seen in Equation 3.1.4. This involved approximating the gradients of the exact first moment solution using the numerical solution. As explained in section 3.2.1, this process introduces additional errors from the first moment solution into the second moment calculation.

If unboundedness is present or the boundary layer is not resolved, the near-wall cell centre values of the first moment solution will have a higher error level that is above the error level seen farther away from the wall. Therefore the first moment gradient calculated in the near-wall cell centres may contain error of numerical significance that is notably above that which is seen away from the wall. Errors are capable of being convected and diffused so the effect will spread into the rest of the
second moment solution. The additional errors may be small and not visible in the error distributions, but they will create an increased difference between the first and second moment solutions when the comparisons are made to find the shape and scale estimates. Therefore the near-wall over-prediction is seen. However, this does not explain why simulations using the constant value boundary condition are affected significantly more than the constant flux boundary simulations.

The explanation comes from how the gradients are calculated in the cell centres: they are interpolated from the face flux values. The fluxes at the internal cell faces are calculated according to the discretisation scheme used, as outlined in section 1.3.3. These are calculated directly from the numerical solution and will contain error from the sources previously discussed. When using the constant flux boundary condition the boundary face flux values are prescribed, constraining the near-wall gradient interpolation close to the correct value. Therefore the interpolation of the cell centre flux values will deviate only slightly. This causes a limited amount of error to enter the SME.

When using the constant value boundary condition the boundary face flux values are not prescribed and they themselves need to be interpolated. This requires the unavoidable use of a first order interpolation. The lack of constraint on the boundary face flux value allows significant deviation from the correct value. When using the first order upwind scheme for the main calculation the interpolations of the fluxes are of the same order of accuracy for both the boundary and internal wall faces. The absolute value of the error is high everywhere, so no relative difference in the estimate is seen between the near-wall and internal cells. However, if one of the second order accurate schemes is used there will be a relative difference. The boundary face flux is found using a first order accurate method while the internal face fluxes are found using a second order accurate method. The near-wall cell centre gradient approximations will contain a higher level of error compared to the internal cells. This enters the SME and now significantly increases the second moment solution error near the wall.

As stated previously, the absolute value of the additional solution error may be small, but when the first and second moment solutions are combined the effect of the additional SME error has entered the estimate via the shape and scale combinations, Equations 3.1.8 and 3.1.9. There may now be a noticeable increase of the estimate value in the near-wall cells compared to the internal cells. The result is a single large peak along the boundary wall with the rest of the distribution over-shadowed.

6.3.5 Scale Estimate Breakdown

Termed the breakdown, this is the point where the summation error against estimate profile changes from a linear relationship with close agreement in scale, to one where the increase of the estimate with the Reynolds number has slowed. It is seen across all of the test cases, discretisation schemes, and boundary conditions. The preliminary investigations indicate that it is dependent on the Reynolds number or cell Peclet number. In the point source in a cross flow (PSC) test case the breakdown consistently occurred at \( Pe_L = 4.2 \).

The cause is from a number of contributions, some of which will be the ones already discussed that affect the shape prediction. To describe the mechanism for this, consider the calculation of the estimate for cases where incorrect features develop with increasing \( Re \). The estimate calculation procedure is outlined in section 3.2.3. The process involves rescaling the shape estimate using the scale estimate. The first step of the rescaling of the shape estimate normalises it by its maximum value. If a large
peak is incorrectly predicted in the shape estimate, for example in Figure 6.1.8, the whole estimate distribution is divided by an incorrectly high value. Already, the estimate summation coefficient has been reduced because the averaging over the distribution contains lower values in the majority of the cells.

The next stage in the rescaling multiplies the current shape estimate by the maximum value of the scale estimate. The calculation of this value will also be affected by the additional errors in the second moment solution. Recalling the mathematical analysis in section 3.2.1, the scale estimate can be decomposed into contributions from the first and second moment solution errors. For the scalar the decomposition is $\delta - \frac{\varepsilon}{\delta}$, where $\delta$ is the first moment solution error and $\varepsilon$ is the second moment solution error. For accurate scale estimation we want $\frac{\varepsilon}{\delta}$ to be as small as possible. Through the cumulative effects of numerical errors, unboundedness, boundary layer non resolution, and the gradient approximation (all of which increase with increasing Reynolds number), $\varepsilon$ increases. Therefore the subtraction in the decomposed scale estimate provides a mechanism to produce lower values.

For practical usage, the breakdown point needs to be analysed. The summation estimate coefficients can be separated from the summation error coefficients and considered individually. Figure 6.3.3 shows examples of the averaged error against $Re$. Across all of the results, the error increases linearly with $Re$ with no change observed at high Reynolds number. Therefore the breakdown must be a result of the estimate scale prediction alone. This allows a more detailed investigation of the effect because the time consuming simulation of the refined solutions necessary to calculate the errors is not required.

Additional data was created to map out the estimate-$Re$ relationship to identify the exact point of breakdown. Figure 6.3.4 shows example plots of averaged estimate against Reynolds number, including the additional data, for simulations matching the ones shown in Figure 6.3.3. It can be seen that there are initial linear regions before a change in gradient. From these, the exact point where the breakdown occurs can be found. In real world applications, the fact that this can be done would allow relatively easy calibration of the method to a particular test case since information about the errors is not needed.
Figure 6.3.3: Examples of summation error expressed as a percentage of the volume average of $f$ against Reynolds number for the three test cases.
Figure 6.3.4: Examples of summation estimate expressed as a percentage of the volume average of $f$ against Reynolds number.
The breakdown point was taken from each of the data sets. The scalar is under consideration so
the breakdown points were converted into the cell Peclet numbers. The results were grouped by mesh
and boundary condition to be averaged, and they are summarised in Table 6.3.1. The breakdown $Pe_L$
varied from between 3.9 and 5.9, which is consistent with the value of 4.2 found from the PSC test
case. The breakdown $Pe_L$ shows a slight variation between the boundary condition types. This can
be explained by a combination of contributions from the boundary layer not being resolved and the
gradient approximation in the near-wall cells as well as the definition of the $Pe_L$ being an averaged
value using $U_B$. With respect to the sources of error that depend on the near-wall cells, it is the $Pe_L$
close to the wall that is of importance. The averaged value does not reflect that.

<table>
<thead>
<tr>
<th>Boundary Type</th>
<th>Very Coarse Mesh $Pe_L$</th>
<th>Coarse Mesh $Pe_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant Flux</td>
<td>5.86</td>
<td>3.94</td>
</tr>
<tr>
<td>Constant Value</td>
<td>5.56</td>
<td>4.45</td>
</tr>
</tbody>
</table>

Table 6.3.1: Summary of breakdown $Pe_L$ for both mesh densities and boundary conditions.

Using the very coarse mesh the breakdown occurred between approximately 5.5 and 6.0, and using
the coarse mesh the breakdown occurred at lower Peclet numbers between approximately 4.0 and
4.5. This is not consistent with the behaviour seen in the PSC test case, which found the breakdown
point was independent of the mesh density. This indicates that the cell Peclet number is not the only
contributing factor to the breakdown in the main test case results, and that the breakdown is sensitive
to the mesh density. The simplicity of the PSC test case did not allow for this to be seen. This is
because a complex velocity was not simulated in the PSC test case, unlike the main test cases.

The during the main test case simulations the velocity solutions, in which the scalar was trans-
ported, were found using a SOC scheme including the slope test. The velocity solution will have its
own error associated with it, which will influence the scalar simulations. The only influences in the
PSC test case result from the scalar solution, so the cell Peclet number is the only driving factor in
the breakdown. The explanation for the mesh density dependency seen here is that the error from
the velocity calculation contributes towards the scale breakdown as well. This will be revisited in the
discussion of the velocity error estimate.

The differences may also be exaggerated due to lack of data. More test cases and mesh densities
are needed to define the breakdown point further. There is also a possible source of error involved due
to the exact breakdown point being unclear in some of the data sets, either through gently developing
curves or scattered data points with a difficult to discern pattern.

The conclusion, for practical usage of the method, is that below a cell Peclet number of 4 the error
scale can be reliably predicted from the estimate scale. There is a linear relationship between the error
scale and the estimate scale, with the estimate consistently over-predicting the error by a factor less
than 3. This allows the user to be confident that the error level in a simulation is below the level
that is predicted by the estimate. After $Pe_L = 4$ the estimate scale is no longer guaranteed to hold
the same linear relationship and would need case dependent analysis. As a worst case scenario the
highest under-prediction seen is by a factor of 10. The estimate scale can be multiplied by 10 to give a
cautious upper limit to the error scale. This upper limit should be used if preliminary case dependent
analysis is not possible.
6.4 Richardson Extrapolation of Impinging Flow Results

In this section the results seen using the proposed error estimate will be compared to those produced using an implementation of the Richardson extrapolation (RE). Equation 2.3.2 gives the relation between the numerical solutions found with two grids that have a refinement ratio equal to 2 to be

\[ f_e = f(h) + mh^p + O(h^{p+1}) = f(2h) + m(2h)^p + O((2h)^{p+1}) \] (6.4.1)

where \( p \) is the order of the discretisation scheme, \( m \) is a known constant, \( f(h) \) and \( mh^p \) are the numerical solution on a mesh with grid spacing \( h \) and the leading truncation error term respectively, and \( f(2h) \) and \( m(2h)^p \) are similarly found on a mesh with grid spacing \( 2h \). With the numerical solutions on these two grids known, and using a first order scheme, this allows an approximation of the first order truncation error \( mh^1 \) to be made as

\[ mh^1 = f(h) - f(2h) \] (6.4.2)

For a second order scheme the second order truncation error \( mh^2 \) can be approximated as

\[ mh^2 = \frac{f(h) - f(2h)}{3} \] (6.4.3)

Out of the three main test cases, the impinging flow is the only one that has a consistently regular refinement across the whole domain, which allows this implementation of the Richardson extrapolation to be applied. The RE estimate was calculated using the results found on the coarse and very coarse meshes to produce an error estimate for the coarse mesh solution. To achieve this the values in the very coarse mesh cell centres were interpolated to the coarse mesh cell centre locations. The interpolation was linear such that to find the value at the midpoint between two points is taken using an average.

Figure 6.4.1 shows a diagram of a number of very coarse mesh cells, overlaid with the regular refinement to allow definition of the interpolations. The very coarse mesh cell centres are labelled \( a - f \), and their boundary wall face centres are also indicated, labelled in the format \( B_{bx} \) for the boundary face on cell \( b \) that is parallel to the \( x \) direction, etc. Inside some of the very coarse cells, labelled \( 1 - 5 \), are 5 coarse cell centres that represent 5 unique locations that require separate interpolations. The values of the scalar, \( f(2h) \), at these locations are referred to as \( f_1(2h), f_a(2h), f_{B_{bx}}(2h) \), etc. The interpolations used were

\[ f_1(2h) = \frac{f_a(2h) + f_{B_{bx}}(2h) + 3f_b(2h) + 3f_{B_{by}}(2h)}{8} \] (6.4.4)

\[ f_2(2h) = \frac{3f_a(2h) + 9f_b(2h) + f_c(2h) + 3f_d(2h)}{16} \] (6.4.5)

\[ f_3(2h) = \frac{-f_{B_{by}}(2h) + 4f_b(2h) + 7f_{B_{bx}}(2h) + 7f_{B_{by}}(2h) - f_{B_{dx}}(2h)}{16} \] (6.4.6)

\[ f_4(2h) = \frac{13f_b(2h) + 10f_{B_{bx}}(2h) + 6f_c(2h) + f_f(2h) + f_{B_{fy}}(2h)}{32} \] (6.4.7)
\[ f_5(2h) = \frac{5f_d(2h) + 2f_{B,y}(2h) + 18f_e(2h) + 5f_f(2h) + 2f_{B,y}(2h)}{32} \] (6.4.8)

Figure 6.4.1: A diagram showing the very coarse mesh cells \( a - f \) (bold lines), and the coarse mesh cells (thin lines) created from their regular refinement. 5 coarse mesh cell centres are labelled, as well as the very coarse mesh boundary wall face centres.

6.4.1 \( f \) Error and Richardson Extrapolation Error Estimate Qualitative Comparisons

Some examples are presented here to provide a qualitative comparison between the RE error estimate and the real error. Figures 6.4.2 and 6.4.3 show first and second order examples found using the constant flux boundary condition with \( Re = 375 \), similar to Section 6.1.1.2. The large features of the errors, such as the peak in error at the impingement point and the medium error levels spreading downstream, are reproduced in the RE estimate. The RE estimate appears to be more accurate at predicting these large features than the proposed SMSE method. The second order example show that zero error bands are also sometimes predicted incorrectly. The second order RE estimate also displays the effects of unboundedness, where there is evidence of ‘checkerboarding’ that not seen in the first order result. This is present in the very coarse mesh solution and has transferred directly to the RE estimate calculation.
Figure 6.4.2: An example of the $f$ error and RE estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.96.

Figure 6.4.3: An example of the $f$ error and estimate from the impinging flow with constant flux boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.69.

Figures 6.4.4 and 6.4.5 show first and second order examples found using the constant flux boundary condition with $Re = 375$, similar to Section 6.2.1.2. The large features of the errors are reproduced in the RE estimate, similar to the constant flux results. Zero error bands are again sometimes predicted incorrectly. Again, the second order RE estimate shows the effect of the solution unboundedness.

Figure 6.4.4: An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.97.
Figure 6.4.5: An example of the $f$ error and estimate from the impinging flow with constant value boundary simulations. They were found with $Re = 375$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $f$. The correlation coefficient was 0.57.

6.4.2 Correlation Coefficients

Figure 6.4.6 shows the RE and SMSE correlation coefficients for the Reynolds numbers tested. In general the RE estimate produces higher correlation coefficients than the SMSE method. The first order RE estimate coefficients are much higher than for second order, with values close to 1. The reason for this is the unboundedness in the second order solutions. The very coarse mesh results are more susceptible to these effects and the effects will be transferred to the RE estimate, degrading the shape prediction. The RE estimate correlation coefficients are seen to decrease with increasing Reynolds number, while typically the SMSE correlation coefficients increase.
Figure 6.4.6: Correlation coefficients produced by the RE and SMSE plotted against \( Re \) for the impinging flow simulations and using two discretisation schemes; FO and SOC.

6.4.3 Summation Coefficients

Figure 6.4.7 shows the RE and SMSE summation coefficients for the Reynolds numbers tested. It shows that the RE estimate produces a scale prediction that is more consistently accurate than the SMSE method. The RE data sets produce close to linear relationships and there does not appear to be a breakdown where the scale prediction becomes a large under-estimate. This can be seen when comparing the first order upwind constant flux simulations. The RE scale prediction remains a slight over-estimate of the error scale.
Figure 6.4.7: Plots of the summation coefficients produced by the RE and SMSE for the ribbed channel simulations, using two discretisation schemes; FO and SOC. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

6.4.4 Richardson Extrapolation Error Estimate Discussion

The RE error estimate performs better than the SMSE by predicting both the shape and scale of the errors more accurately. This was expected because grid refinement and the RE estimate systematically converge on the error. The RE estimate is dependent on some of the same issues seen in the SMSE. Unboundedness in the solutions affects the RE estimate, and zero error bands are also not always predicted correctly. These have been seen to degrade the second order RE estimate’s correlation coefficients. The RE estimate does not appear to suffer from the issues seen in the SMSE relating to the second moment solution process, such as capturing the near wall boundary layer and the gradient approximation in near-wall cells. Although near wall effects were not visible, the RE estimate may
still be sensitive to them as a result of the calculation of the wall face values by the CFD software, and of the method’s interpolations of the very coarse mesh solutions onto the coarse mesh.

The SMSE correlation coefficients are lower than for the RE estimate, however, they are, in general, still high and the important features of the errors are discernible. The SMSE scale prediction is less reliable than for the RE estimate as a result of the observed SMSE scale breakdown, which not observed in the RE estimate. However, under sensible limitations the prediction is of the same order of magnitude, making the SMSE estimate of practical use. This highlights the balance that the proposed method intends to strike between systematic rigour and computation expense. The RE estimate requires an additional mesh, secondary simulation time, and an interpolation procedure that may become complex if a non-regular refinement is used that may also degrade the accuracy of the estimate. The SMSE requires a single mesh, with a slight increase in computation time instead of an additional simulation. Also, the final estimate calculation is done inside the main simulation without an interpolation, which means that the SMSE is, in theory, applicable to any grid structure. The expense for this procedural simplification is the degradation of the accuracy of the prediction that has been observed in the results.

6.5 Chapter 6 Summary

Chapter 6 presented the results from the investigation of the scalar error estimate. The first section examined the use of the constant flux boundary condition and the second section examined the use of the constant value boundary. Both sections were laid out to present general qualitative descriptions and comparisons of the errors and estimates before making quantitative comparisons using the summation and correlation coefficients. The third section was a discussion of the results and compares to the Richardson extrapolation procedure. In general, the error estimate was successful in predicting the shape of the errors, producing high correlation coefficients, although in the impinging flow test case the values were lower than those found when using Richardson extrapolation. However, some limitations presented themselves, for example unboundedness, and they have possibly caused the accuracy of the shape predictions to degrade. The scale predictions were also good under certain limitations. The results demonstrated an upper limit on the cell Peclet number in order to achieve a reliable scale prediction. Above the limitation the scale prediction became a large underestimate. This was not observed when using Richardson extrapolation. The additional computation time required was low, between 10% – 20%.

It was noted that some of the limitations observed may be the result of the simulation of the velocity that was required to find the flow field for the scalar investigations. Therefore a complete discussion of the scalar error estimate could not be made until the velocity error estimate has also been investigated. The next chapter will present the investigation of the velocity error estimate and complete the discussion of both the scalar and velocity error estimates together.
Chapter 7

Velocity Error Estimate in Laminar Fluid Flow

Chapter 7 presents the results for the velocity error estimate in laminar fluid flow. The real error distributions were found by the difference between the coarse Code_Saturne solutions and the refined mesh Code_Saturne solution. In the following sections it is the velocity vector error that is considered which is calculated using Equation 3.2.29. The value of the velocity components in each coarser mesh cell were subtracted from the values of the refined solution at the location of the coarser mesh cell centre, and the absolute value was taken. The estimate was calculated according to Equation 3.2.30.

The solving of the additional equations will increase the computation time. This increase will be monitored by comparing the time it takes to complete the simulations with and without the simulation to find the error estimate. The comparisons will be made by expressing the increased computation time as a percentage of the computation time without the method.

This chapter is separated into two sections, the first presents the results, and the second discusses conclusions that can be drawn about how the method performs. The results section first presents examples of the $u$ error distributions alongside corresponding error estimate distributions. Qualitative descriptions of these are given. Next the correlation coefficients, calculated using Equation 3.2.34, are presented to judge how well the shape has been predicted. To give context to these numbers, examples of high and low correlation coefficient results are shown for qualitative comparison. Finally the averaged summation quantities for the error and estimate, calculated using Equations 3.2.32 and 3.2.33, are presented to judge the scale prediction.

The discussion section explains the trends seen in the results and discusses the limitations of the method when applied to the transport of momentum in a laminar fluid flow. It also makes comparisons to the results seen in the scalar transport investigation.
7.1 Laminar Flow Results

7.1.1 Laminar Flow f Error and Error Estimate

This section will present some examples of the velocity error distributions and their corresponding estimate distributions for the three test cases. In general, each discretisation scheme produces slightly different sets of features in the error and estimate distributions. Typically, the results that use the first order scheme produced distributions different from the more similar results that use either of the two second order schemes. For each test case two sets of error and estimate distributions are presented, giving an example from the first order simulations and an example from the second order simulations. The physical characteristics observed in the distributions using the two orders of discretisation are discussed in general, referring to the given examples only to illustrate the comparisons.

7.1.1.1 Ribbed Channel

Figure 7.1.1 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 22\% computation time after 2000 iterations. The highest error is found in a small number of cells adjacent to the corner of the downstream rib. There is also lower, but still significant, error spread across the centre of the channel as well as around the corner of the upstream rib. There is a zero error band that outlines the recirculation bubble that is behind the upstream rib. In the estimate distributions the high errors at the corners are not visible. The error spread across the centre of the channel is predicted, and the scale is adjusted to its maximum value. For this reason the estimate in the example appears very different to the error, but in actuality they are similar, except for the absent zero-band. As the Reynolds number is increased the peak in error at the corner becomes less significant compared to the error in the channel centre, and the distributions become more similar.

Figure 7.1.2 is an example from the simulations using the second order centred discretisation scheme. High error is seen adjacent to the corners of both ribs with low error found in-between them. This is reflected in the estimate distributions, although the location is not predicted correctly. In the example it is predicted along the top of the rib, two cells away from the corners.
Figure 7.1.1: An example of the $u$ error and estimate from the ribbed channel simulations. They were found with $Re = 175$, using the first order upwind scheme on the coarse mesh, expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.81.

Figure 7.1.2: An example of the $u$ error and estimate from the ribbed channel simulations. They were found with $Re = 175$, using the second order centred scheme on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.12.

7.1.1.2 Impinging Flow

Figure 7.1.3 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 17% computation.
time after 100 iterations. Regions of high error are found around the corner of the intersection between the inlet and impingement channel, and the highest error is seen in the cells adjacent to it. There is lower, but still significant, error seen moving downstream which decreases to very low error once the flow has returned to a normal channel flow. There are also two zero error bands outlining the path of the fluid. The estimate predicts error to be in the region close to the intersection but does not show the error in the cells adjacent to the corner, and no zero-band is seen. This is similar to the behaviour observed in the first order ribbed channel example above. The lack of prediction of this peak causes the scale to be normalised to the values in the channel, so the distributions appear different.

Figure 7.1.3: An example of the $u$ error and estimate from the impinging flow simulations. They were found with $Re = 375$, using the first order upwind scheme on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.54.

Figure 7.1.4 is an example from the simulations using the second order centred discretisation scheme. In the example the main location of high error is the cells adjacent to the intersection corner, but as $Re$ increases, the error along the impinged wall increases. The estimate identifies these features, however, it also predicts the error along the inlet channel wall that becomes more prominent as $Re$ increases.

Figure 7.1.4: An example of the $u$ error and estimate from the impinging flow simulations. They were found with $Re = 375$, using the second order centred scheme and on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.17.

7.1.1.3 Hexagonally Stacked Tube Bundles

Figure 7.1.5 is an example from the simulations using the first order upwind discretisation scheme. The calculation of the error estimate for these simulations required an additional 17% computation time after 500 iterations. High error is seen across the whole geometry, following the path of the fluid. The error goes to zero at the walls as well as along two zero error bands that outline the fluid path. The estimate predicts error across the whole geometry and goes to zero at the walls, however, the bands are not predicted.
Figure 7.1.5: An example of the $u$ error and estimate from the tube bundles simulations. They were found with $Re = 79.9$, using the first order upwind scheme and on the coarse mesh, and expressed as a percentage of the volume average of $u$. The correlation coefficient was 0.60.

Figure 7.1.6 is an example from the simulations using the second order centred discretisation scheme. There are two regions of high error which are along the upstream walls of the centre and right cylinders, where the fluid impinges. They remain close to the wall and low error spreads out from them. These same features have been predicted in the error estimate distribution.

7.1.2 Laminar Flow Correlation Coefficients

How well the shape has been predicted is judged by the correlation coefficient. This section will present the coefficients for the three test cases plotted as correlation coefficient against Reynolds number. For each test case two examples are given, one each for a high and a low correlation coefficient, to give insight into the meaning of these values.

7.1.2.1 Ribbed Channel

In Figure 7.1.7 the correlation coefficients are shown for the different simulations. On both meshes the first order correlation coefficients are similar. The coefficients are low at low Reynolds number and increase with $Re$ to a value of 0.8. The two second order schemes produce sets of coefficients that differ between the two meshes as well as between schemes. The second order centred scheme produces profiles that are similar in shape to the first order profiles, and level off at a low coefficient of 0.2 on the very coarse mesh, and are higher at 0.4 on the coarse mesh. The second order linear upwind profiles
decrease with increasing $Re$ and actually become negative. On the very coarse mesh the coefficients reach as low as $-0.3$, and are higher on the coarse mesh at $-0.1$.

Figure 7.1.7: Correlation coefficients plotted against $Re$ for the ribbed channel velocity simulations using three discretisation schemes; FO, SOC and SOLU.

Figure 7.1.8 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order linear upwind scheme at $Re = 344$, which produced a correlation coefficient of $-0.31$. The coefficient has become negative which indicates that there is a negative correlation between where there are peaks and where there are dips. A large contribution comes from the incorrect prediction of error along the top wall not seen in the error distribution, which is typical of the high $Re$ results using this scheme. At lower Reynolds number the presence of this feature is reduced and the coefficients are higher.

The second example is for the coarse mesh with the first order upwind scheme at $Re = 1640$, which
produced the highest correlation coefficient of 0.88. At high Reynolds numbers the first order estimates have a high coefficient. The first order example shown in Figure 7.1.1 shows a lower Reynolds number error/estimate set with correlation coefficient 0.81. The reason the coefficients drop is because of the peak in high error predicted adjacent to the downstream rib corner. The estimate does not predict the zero error band that is present across the range of \( Re \) tested.

![Figure 7.1.8](image)

Figure 7.1.8: Two examples of the error and estimate distributions for the ribbed channel velocity simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of \( f \).

### 7.1.2.2 Impinging Flow

In Figure 7.1.9 the correlation coefficients are shown for the different simulations. Comparing between the two meshes, the profiles of the three schemes are a similar shape, although on the very coarse mesh the values are 25\% lower. The first order profiles are level, between 0.5 and 0.6 on the coarse mesh
and between 0.4 and 0.5 on the very coarse mesh. Both the second order centred and linear upwind coefficients on the coarse mesh become level at 0.4. The second order centred coefficients on the very coarse mesh also become level, at a value of 0.2. However, the second order linear upwind coefficients on the very coarse mesh begin to decrease with Reynolds number and actually become negative down to a value of $-0.2$.

![Correlation Coefficients against Reynolds Number for the Very Coarse Mesh](image1)

![Correlation Coefficients against Reynolds Number for the Coarse Mesh](image2)

Figure 7.1.9: Correlation coefficients plotted against $Re$ for the impinging flow velocity simulations using three discretisation schemes; FO, SOC and SOLU.

Figure 7.1.10 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order linear upwind scheme at $Re = 2500$, which produced a correlation coefficient of $-0.16$. In this example there is not a great deal of similarity at this Reynolds number. At low $Re$ the main source of error is adjacent to the intersection corner similar to what is seen in 7.1.4 and the estimate reflects this. As $Re$ increases a strip of high estimate develops down the centre of
the downstream channel, while the error develops high error with a zero error band down the centre. This is seen in the example. It is these features that lead to a negative correlation coefficient.

The second example is for the coarse mesh with the first order upwind scheme at $Re = 2500$, which produced the highest correlation coefficient of $0.66$. The first order errors show high error in the centre of the channel starting where the inlet fluid impinges, which spreads downstream. How far it spreads increases with Reynolds number. There is a zero error band cutting through the error distribution along the centre of the fluid flow that is not seen in the estimate, however the correlations are consistently high.

![Figure 7.1.10: Two examples of the error and estimate distributions for the impinging flow velocity simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.](image)

### 7.1.2.3 Hexagonally Stacked Tube Bundles

In Figure 7.1.11 the correlation coefficients are shown for the different simulations. The results across all schemes and the two meshes are similar. The correlation coefficients decrease with increasing Reynolds number. The first order coefficients are slightly lower at 0.6 compared to the second order coefficients at 0.7.

Figure 7.1.12 shows two examples of the error and estimate distribution. The first is for the very coarse mesh with the second order centred scheme at $Re = 164$, which produced a correlation coefficient of 0.48. Across the range of Reynolds numbers tested the correlation coefficients are high enough to be confident the shape is being predicted well. The error is found around the upstream sides of the centre and right cylinder walls where the fluid impinges. In the example given the coefficients are lower because the location of the peaks in the estimate are predicted ten cells further downstream around the cylinder walls.

The second example is for the coarse mesh with the second order linear upwind scheme at $Re = 79.9$, which produced the highest correlation coefficient of 0.75. The simulation is similar to the second order centred distributions and the coefficients are higher because the predicted peaks in the estimate are closer to the location of the peaks in the error.
Figure 7.1.11: Correlation coefficients plotted against $Re$ for the tube bundles velocity simulations using three discretisation schemes; FO, SOC and SOLU.
Figure 7.1.12: Two examples of the error and estimate distributions for the tube bundles velocity simulations, one for a low correlation coefficient (left) and one for a high coefficient (right). The scales are expressed as a percentage of the volume average of $f$.

7.1.3 Laminar Flow Summation Coefficients

The prediction of the scale is assessed using the summation coefficients. In this section the summation coefficients are presented as graphs of average error against average estimate for each test case.

7.1.3.1 Ribbed Channel

Figure 7.1.13 shows the average error against average estimate for the three discretisation schemes and the two meshes. Two separate behaviours are seen between the first and second order scheme profiles. The first order profiles are linear with the estimate scale over-predicting the error scale by a factor of 3. There is also evidence of the breakdown after the seventh data point, where the estimate slows its increase with increasing Reynolds number.

However, the second order errors do not increase with Reynolds number. The estimate scale over-predicts the error scale by a factor between 1 and 2. For the coarse mesh all of data points are located together. For the very coarse mesh only the estimate varies with Reynolds number, so the data points spread out at a constant value of the error.
Figure 7.1.13: Plots of the summation coefficients for the ribbed channel velocity simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

7.1.3.2 Impinging Flow

Figure 7.1.14 shows the averaged error against the averaged estimate for the three discretisation schemes and the two meshes. The first order profiles are linear, with the error and estimate scales increasing with increasing Reynolds number. The estimate scale over-predicts the error scale by a factor of 3. The second order profiles do not show a clear relationship between the two. The error coefficients increase while the estimate coefficients are scattered. Again the breakdown is visible.
Figure 7.1.14: Plots of the summation coefficients for the impinging flow velocity simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$.

### 7.1.3.3 Hexagonally Stacked Tube Bundles

Figure 7.1.15 shows the averaged error against the averaged estimate for the three discretisation schemes and the two meshes. A similar situation to the ribbed channel is seen. The first order profiles are linear, error and estimate scale increases with the Reynolds number. The estimate scale over-predicts the error scale by a factor between 4 and 5. The second order profiles change very little over the Reynolds number range tested with all of the data points located together.
Figure 7.1.15: Plots of the summation coefficients for the tube bundles velocity simulations, using three discretisation schemes; FO, SOC and SOLU. They are displayed as error against estimate and expressed as a percentage of the volume average of $f$. 
7.2 Discussion of the Velocity Error Estimate in Laminar Fluid Flow

In general, the errors found using the first order upwind scheme produce distributions that spread across the entire fluid domain, while the second order schemes produce error distributions that are reserved to small regions of high stress such as corners of the flow or impingement points. The estimate distributions successfully reflect this, although the second order shape predictions degrade with increasing Reynolds number, producing lower correlation coefficients. The explanation refers back to the discussion of the scalar error estimate in section 6.3 because the same issues apply to the velocity error estimate.

Zero error bands are seen in the error distributions that are not predicted in the error estimate, similar to the scalar error estimate investigation. For example the first order upwind tube bundles simulation, using the coarse mesh and $Re = 79.9$, seen in Figure 7.1.5. These will have the effect of reducing the correlation coefficients despite providing useful information about error generating regions of the flow.

Another type of feature seen in the error distributions, that were not predicted in the estimate, is error generated at sharp corners, such as in the impinging flow intersection between the two channels (for example Figure 7.1.3), and the top of the ribs in the ribbed channel flow (for example Figure 7.1.1). These were not seen in the error estimate because either the estimate levels are generally higher and over-shadow them, or the differences between the velocity and kinetic energy solutions do not develop to become visible.

Extraneous features close to walls, similar to those seen in the scalar error estimates, are also seen in the error estimate distributions. This is expected because the calculation of the kinetic energy will be influenced by the effects of solution unboundedness, non-resolution of the boundary layer and near-wall gradient approximation. There are also additional sources of first moment error included in the velocity SME source term, as explained in section 3.2.2. There will be error from the numerical pressure solution, or possible interference between the velocity components, etc. This will lead to an increased difference when the velocity and kinetic energy solutions are compared via mechanisms already discussed. The additional sources will contribute to the generally lower correlation coefficients seen using the velocity error estimate, when compared to the success of the scalar error estimate.

The $K$ simulations use the constant value boundary conditions to match the velocity boundary conditions of zero at the walls. This has lead to the same over-shadowing predictions close to walls when using the second order schemes. For example, the second order linear upwind ribbed channel flow simulation, using the very coarse mesh and $Re = 344$, seen in Figure 7.1.8 (left). Here, high estimate is seen along the upper wall. Recall that the issue is caused by a drop in order of gradient approximation combined with boundary layer non-resolution. This explains why features can appear in the estimate distributions along any of the walls in the geometry, since all walls are involved in the velocity and kinetic energy calculation.

All of these will contribute to enable the scale breakdown through a similar process to what was discussed in section 6.3.5, the explanation of the scalar estimate scale breakdown. However, the occurrence of the breakdown is not as obvious in the velocity summation coefficient profiles. For the
first order upwind profiles there was only a slight effect. This is unlike the scalar summation coefficient profiles where large divergences are seen. For both the second order schemes the error and estimate scales show little variation with Reynolds number. The velocity is predicted so accurately in the main regions of the flow that the error does not develop over the range of Reynolds numbers tested. The error that is seen generates at the corners and remains at a constant level. As a result, not enough data is available to discern a breakdown cell Peclet number.

7.2.1 Effect of Second Order Centred Scheme Velocity Errors on the Scalar Simulations

A conclusion drawn from section 6.3 was that the velocity flow field must also be influencing the scalar solution and its error estimate. The flow fields in which the scalar is transported are found using the second order centred scheme. As seen in the SOC scheme velocity error distributions, the error peaks are at sharp corners in the geometry and do not spread far into the fluid flow. The velocity errors will produce error in the scalar simulations in these locations. As shown previously, solution errors are able to magnifying during the estimate calculation process. Therefore the SOC velocity error is able to influence the scalar error estimate.

Increasing the Reynolds number causes the SOC velocity error scale to increase negligibly, and the location of error remains close to the sharp corners in the geometry. Therefore it contributes a small fairly consistent addition to the scalar solution errors. This explains certain features seen in the scalar simulation errors that become visible when the surrounding error is low enough. For example, consider the second order centred impinging flow simulations using the coarse mesh and constant value boundary condition. Refer to Figure 6.2.10 (left) which used a low Reynolds number, \( Re = 125 \), and Figure 6.2.4 which used a higher Reynolds number, \( Re = 375 \). As the Reynolds number was decreased the general error level throughout the domain decreased. This allowed a feature next to the sharp intersection corner to become significant relative to the rest of the distribution. This is the same location of the velocity error, seen in Figure 7.1.4.

For the impinging flow test case, these features did not appear in the estimate distributions. This is because there was very little transport of the scalar in this region to allow the error to magnify in the estimate calculation. This effect is the reason the correlation coefficients were observed to significantly decrease at low Reynolds numbers. Further investigation into the interdependence is required to fully understand the influence of the velocity solution on the scalar first and second moments.

7.2.2 Scale Estimate Breakdown: Velocity and Scalar

As previously described in section 6.3.5, the cell Peclet number at which the breakdown occurred in the main test cases showed a dependence on the mesh refinement. This was a different result to the point source in a cross flow test case in which no mesh dependence was observed. The difference between the test cases was that the velocity was not prescribed in the main test cases, and so included a velocity error. This error has the potential to contribute to the estimate breakdown similarly to the other effects discussed.

It has been shown that the SOC velocity error adds a constant source of error into the scalar
simulations. If the velocity error influences the overall scale prediction then this provides an explanation for the mesh dependency. Assuming that the breakdown depends on the influences already presented, each region of high error and estimate will act independently. If, as suggested, it is dependent on the cell Peclet number then each of these regions will experience a different $Pe_L$ because the cell Peclet number is dependent on the velocity in the cell. Therefore each region can experience the breakdown individually. This allows successive regions to experience the breakdown at different Reynolds number as it is increased. The constant addition to the scalar simulation may contribute a stabilising effect to the scale estimation that is different for each of the meshes, thereby creating a dependency on the refinement.

This effect is still not fully understood. The mechanisms enabling the breakdown require further investigation. It is the main concern observed in the investigation of the method, and requires a full explanation.
7.3 Chapter 7 Summary

Chapter 7 presented the results from the investigation of the scalar error estimate, thereby completing the investigations of the proposed error estimate in this thesis. The chapter was laid out similar to chapter 6 and presented a general qualitative descriptions and comparisons of the errors and estimates before making quantitative comparisons using the summation and correlation coefficients. The second section was a discussion of the results that also drew conclusions from the previous chapter. Much like the scalar error estimate results limitations were observed in the methods ability to predict the errors. The correlation coefficients were in general lower, indicating a less accurate shape prediction. However, the scale prediction appeared to be less affected by the breakdown and remained a slight over estimate in most cases. The additional computation time required was higher than for the scalar simulations, but remained low at about 20%. The conclusions from the previous two chapters will be discussed in the following conclusions chapter, which is the final chapter in this thesis.
Chapter 8

Conclusions and Further Work

8.1 Conclusions

The aim of this work was to meet the requirements of today's industry with respect to error analysis in CFD. In industry there are time and computer power restraints and full error analysis is not always possible due to its time consuming nature. The systematic methods required for accurate error analysis, such as multiple grid generation or the use of higher order schemes, are therefore impractical. The additional simulation time for error analysis could be greatly reduced by instead using certain non-systematic error estimation methods, such as those that examine auxiliary algebraic evaluations on the same grid using the solutions as input. However, these do not meet the accuracy requirements of industry because the methods will often not have a direct relation to any error measure of engineering or scientific interest. This leaves an opening in the field of error estimation methods for one that balances the two. With this in mind, the second moment solution method was proposed to fill this gap, and has been successfully applied to six laminar test cases. Under certain limitations the estimate is able to give reliable information about the scale and location of the errors and is relatively cheap to compute, increasing computation time by between 10% - 20%, thereby meeting industry's requirements.

The proposed error estimate is a mixture of the two classes of systematic and non-systematic methods. It is based on mathematical justifications derived from combinations of the first and second moment solutions. Therefore it is expected to make accurate predictions of the error distributions, clearly identifying high and low regions of error and their scale. The increase in predictive accuracy upon non-systematic methods comes at the expense of solving an additional transport equation for each considered variable. This places the computational cost of the method above that of simple algebraic evaluations, but below that of the solution of using higher order schemes or higher resolution grids. This was reflected in the increase in computation time observed. The decrease of the computational cost upon systematic methods comes at the expense of the accuracy of the estimate prediction. This is reflected in the comparisons between the proposed method and an implementation of the Richardson extrapolation method. However, the results in the work so far have shown the error estimate is capable of identifying the scale and location of numerical errors, under certain limitations, with good reliability.

The accuracy was judged by two sets of coefficients, the correlation coefficient for the shape predic-
tion and the averaged summation coefficients for the scale prediction. The testing on the six test cases showed that when using the first order upwind scheme the error estimate produced consistently high correlation coefficients indicating accurate prediction of error location. When using the two second order schemes, for some simulations there was degradation of the prediction with increasing Reynolds number, indicated by the decrease in correlation coefficient. The cause for this is from a number of contributions such as solution unboundedness, non-resolution of the boundary layer, near-wall gradient approximation, and pressure solution error introducing extraneous features.

The contributions to the shape prediction degradation also influenced the scale prediction. As the Reynolds number was increased the estimate scale prediction experienced a breakdown. The estimate scale slowed its increase with increasing Reynolds number, while the error scale continued unaffected. The relationship between the error and estimate scales was initially a linear increase, and over-predicted the error by a factor of 3. After cell Peclet number 4 the estimate scale became an under-prediction by up to a factor of 10. This was the one major concern of the method.

The work here has shown the method has great potential for industrial use due to the balance it strikes between accuracy and relatively cheap computation time. Since it is based on mathematical justification before a CFD discretisation is applied, the method is applicable to the any of the solver methods such as finite differences, finite element, and finite volume. The method could be easily incorporated into a CFD solver and would require no input from the user beyond selecting it for use. Therefore it is a strong candidate for adoption by the CFD community as a whole.

However, further development of the method is required before this could be done. A greater understanding is needed of the limitations that are related to the second moment solution process. While no particular anomalies remain unexplained, further investigation is needed of the mechanisms that allow the degradation of the shape and scale. Investigation of how the velocity solution influences the scalar solution must also be conducted in greater detail.

8.2 Further Work

8.2.1 Extension to Turbulence

Currently, testing has been carried out only on laminar fluid flow and the transport of scalars within it. Before the method can be put into practical use it needs to be extended to include turbulent flows. This increases the range of fluid flows to which the estimate is applicable to those most commonly seen in industrial circumstances. This would be achieved using the same procedure in this thesis with the equations that include turbulent effects. For example, when using the Reynolds-averaged approach, the Reynolds-averaged Navier-Stokes (RANS) equations can be formed as

\[
\rho \frac{\partial \overline{\mu}'}{\partial x_j} = \rho \overline{b_i} + \frac{\partial}{\partial x_j} \left[ -\mathbf{P} \delta_{ij} + 2\mu D_{ij} - \rho \overline{u_i'u_j'} \right]
\]

(8.2.1)

where an overbar represents a time averaged quantity, \( \overline{D_{ij}} = \frac{1}{2} \left( \frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i} \right) \) is the mean rate of strain tensor, and \(-\rho \overline{u_i'u_j'}\) is the nonlinear Reynolds stress term that contains the information about the turbulence \([14, 41 & 43]\). This term requires additional estimation to close the RANS equation, which
requires the use of turbulence modelling to allow a solution to be obtained. One method to do this employs the use of the Boussinesq eddy viscosity approximation \cite{10}. The approximation assumes that the components of the Reynolds stresses are proportional to the components in $D_{ij}$, giving

$$-\rho u_i' u_j' = 2\mu_t D_{ij} - \frac{2}{3} \rho k \delta_{ij}$$

(8.2.2)

where $\mu_t$ is the turbulent viscosity and can be modelled using turbulence models such as the $k - \epsilon$ model.

The first step in the continued research would be deciding on the correct formulation for the second moment equation. The simplest way would be to transform Equation 8.2.1 through the same procedure used in Section 3.1.3. However, the total kinetic energy will also be subjected to turbulent effects and the transformed equation may not reflect this. Another possibility would be to start from Equation 3.1.6, the momentum second moment equation, and apply the Reynolds-averaged approach. Consideration must then be taken into how to include the turbulence model that was used for the momentum equation. From here any additional sources can be approximated using the velocity solution.

The error estimate can be calculated following the same formulation from the first and second moment solutions. A complication that may arise is due to uncertainties related to turbulence modelling. How this interferes with the estimate calculation should be investigated. The full exploration of the mechanisms that caused the laminar limitations will also provide a basis to understand the influences of the turbulent uncertainties.

### 8.2.2 Extension to Unsteady Flows

The next step in the development of the method is the extension to unsteady flows. Alongside the inclusion of turbulence, this would ensure that the method is applicable across most types of fluid flow problems encountered in industrial CFD. The steps to investigate this are the same as what has been stated previously. Consideration must be made towards the correct formulation of the second moment equation before testing on simple unsteady test cases, for example the tube bundle test case at higher Reynolds numbers.

### 8.2.3 Adoption by Industrial CFD Codes

Implementing the method into an industrial CFD code does not require extensive coding. In Code\_Saturne the additional lines of code can be written into a user subroutine that is able to alter the source terms and boundary conditions of a user scalar. Once the error estimate subroutine has been chosen for inclusion into the simulation, through the use of a control coefficient in \textit{usini1}, and a user scalar included for its use, no other input from the user is required. This allows the use of the error estimate to be selected on or off, and during the final iteration the estimate is produced on the same mesh for inspection. With regards to other CFD solvers the method could be adopted in the same way.
References


[36] T. Oliver et al., "Phenomena Identification and Ranking Table", Office of Nuclear Regulatory Research 2008


[40] L. Prandtl, "Z. Angew", Mathematical Mechecahnics 1925


