

THE UNIVERSITY OF
SYDNEY

Novel Program for Particle Image Velocimetry (PIV) Pressure Field Reconstruction

A Thesis submitted to the University of Sydney for the degree of
Bachelor of Engineering (Honours) (Aeronautical)
School of Aerospace, Mechanical & Mechatronic Engineering

Alexander Le Poer Trench

SID: 470427349

2021

The work in this thesis is subject to the copyright laws of Australia, as specified in the Copyright Act 1968, and cannot be reproduced without the sole permission of the Author.

Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the Author. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the Author.

Copyright
Alexander Le Poer Trench, 2021
All rights reserved.

Abstract

Particle Image Velocimetry (PIV) is a method of quantitatively measuring planar or volumetric velocity fields in a non-intrusive manner, that makes it desirable for fundamental turbulence research, including boundary layers, transonic flow, and rotary flow. This thesis aims to expand the use of PIV velocity data by designing and implementing a freely available program to reconstruct pressure fields from subsonic velocity fields, thus allowing direct aerodynamic design optimisation informed by load determination, fatigue-life, and acoustic spectra.

Using velocity fields derived from a Wall-Modelled Large Eddy Simulation (WMLES) of a wall-mounted hemisphere or hemispherical protuberance, this thesis reveals that the Lagrangian approach to pressure field reconstruction yields lower pressure error than the Eulerian approach, and is more robust to noise and advection dominated flow. However, both approaches suffer from spatial filtering due to the Successive-over-relaxation algorithm used to determine the pressure from its Poisson equation. This filtering results in the pressure field from an empirical jet flow being smoothed, such that its "tube-like" structure observed in the velocity field is not reflected in the pressure field. Finally, the settings designed for user customisability, were found to have little effect on the mean error found in the pressure fields of the Eulerian or Lagrangian approaches for the WMLES of the hemispherical protuberance.

Declaration

This thesis was completed between March and November of 2021 under the supervision of Dr Gareth Vio, from the School of Aerospace, Mechanical and Mechatronic Engineering (AMME) at the University of Sydney.

The following areas of original work were carried out for this thesis:

- Comprehensive literature review pertaining to PIV, hemispherical protuberances, and pressure field reconstruction
- Design and implementation of a pressure field reconstruction program in the MATLAB programming language
- Verification of pressure field reconstruction program implementation using velocity fields from analytical potential flows, a CFD simulation, and empirical PIV measurements

The above represents an accurate summary of my contribution to this thesis.



Alexander Le Poer Trench

.....
.....
.....
.....
.....

Dr. Gareth Vio

Acknowledgements

I would like to thank Dr. Gareth Vio for supervising this thesis, being quite generous with his time, and always being patient with me. I always enjoyed our fortnightly thesis meetings.

I would also like to thank him for teaching me how to operate the University of Sydney low speed wind tunnel during the first semester of this thesis, and Derrick Ho for setting up the infrastructure necessary to perform PIV experiments in that laboratory. Even with the eventual shut-down of those facilities in the second semester due to the COVID-19 pandemic, I loved the practical work.

Thank you to Dr. Nicholas Giannelis for providing the data from his CFD simulations of a hemispherical protuberance and an explanation for the setup of the simulation, which helped immensely with making this thesis more rigorous.

Finally, thank you to Peter Manovski and Matteo Giacobello from DSTG for providing requirements of what they wanted from a pressure field reconstruction program, it helped to have that from the beginning and understand what I needed the final product to be.

Contents

Abstract	i
Declaration	ii
Acknowledgements	iii
List of Figures	vii
List of Tables	x
List of Abbreviations	xi
List of Symbols	xii
1 Introduction	1
2 Literature Review & Background	3
2.1 Particle Image Velocimetry (PIV)	3
2.1.1 History of PIV	3
2.1.2 Modern PIV Techniques and Sources of Uncertainty	6
2.1.2.1 Seeding Uncertainty	7
2.1.2.2 Illumination Uncertainty	8
2.1.2.3 Recording and Calibration Uncertainty	9
2.1.2.4 Evaluation and Post-Processing Uncertainty	10
2.1.3 Current Applications of PIV	11
2.2 Hemispherical Protuberances	12
2.3 Pressure Field Reconstruction	14
2.3.1 Material Acceleration Estimation	15
2.3.1.1 Eulerian Approach	15

2.3.1.2	Lagrangian Approach	16
2.3.1.3	Single Measurement Approaches	19
2.3.2	Pressure Estimation	20
2.3.2.1	Direct Integration	20
2.3.2.2	Poisson Solution	22
3	Objectives & Program Design	27
3.1	Research Objectives	27
3.2	Program Design	28
3.2.1	Program Overview and Architecture	28
3.2.2	Non-dimensionalisation	30
3.2.3	k Nearest Neighbour Interpolation	31
3.2.4	Least Squares Polynomial Regression	32
3.2.5	Domain Boundary	33
3.2.6	Eulerian Source Term	34
3.2.7	Lagrangian Approach	35
3.2.7.1	Particle Track Construction	36
3.2.7.2	Material Acceleration	37
3.2.7.3	Source Term	38
3.2.8	SOR Poisson Solver	39
3.2.9	User-defined Settings	41
4	Program Results & Verification	43
4.1	Potential Flow	43
4.1.1	Impinging Jet Case	44
4.1.2	Hemispherical Protuberance Case	46
4.1.3	Summary	47
4.2	CFD Simulation	49
4.2.1	Material Acceleration Error Comparison	51
4.2.2	Pressure Coefficient Error Comparison	53
4.3	Effect of Program Settings	57
4.4	PIV Experiment	62
4.4.1	Hemispherical Protuberance	62
4.4.2	Unsteady Inviscid Jet Core	64
5	Conclusion & Recommendations	67

List of Figures

2.1	Early illustration of turbulence: <i>Studies of Turbulent Water</i>	4
2.2	Separated Flow over an Aerofoil	5
2.3	Tracer particle density determines the velocimetry method: (a) PTV (b) PIV (c) LSV	6
2.4	General PIV Experimental Design and Associated Subsystems	7
2.5	Structures of a Flow around a Hemispherical Protuberance	13
2.6	(a) Water Flow Visualisation of Hemispherical Protuberance (b) CFD Large-Eddy Simulation of a Hemispherical Protuberance	13
2.7	Earliest Results of Pressure Reconstruction from Particle Streak Images .	14
2.8	Propagation of Imaginary Particles A and B across an Example Velocity Field	17
2.9	RMS Error in Streamwise Velocity for the Particle Track Construction Schemes	18
2.10	(a) Noise Amplification for Material Acceleration Approaches (b) RMS Error in Material Acceleration for Material Acceleration Approaches and Track Lengths	19
2.11	Two Point Average Direct Pressure Integration Scheme	21
2.12	(a) Four Point Average Direct Pressure Integration Scheme (b) Omni- directional Direct Pressure Integration Scheme	22
2.13	Successive-over-relaxation Iterative Method for Pressure Estimation across a 2D Computational Domain	25
3.1	Program Architecture	28
4.1	(a) Total Velocity Contour and Streamlines of Impinging Jet Potential Flow (b) Reconstructed Pressure Coefficient Contour (c) Pressure Coeffi- cient Error Contour	45

4.2	(a) Total Velocity Contour and Streamlines of Hemispherical Protuberance Potential Flow (b) Reconstructed Pressure Coefficient Contour (c) Pressure Coefficient Error Contour	48
4.3	Distinct Regions of Flow around a Hemispherical Protuberance	51
4.4	(a) WMLES Total Velocity Contour for Flow around a Hemispherical Protuberance at its Centre-plane (b) Eulerian Approach Material Acceleration Error (c) Lagrangian Approach Material Acceleration Error	52
4.5	(a) Reconstructed Pressure Coefficient Field from Eulerian Approach (b) Pressure Coefficient Error Field from Eulerian Approach	54
4.6	(a) Reconstructed Pressure Coefficient Field from Lagrangian Approach (b) Pressure Coefficient Error Field from Lagrangian Approach	55
4.7	(a) Effect of Number of Particles on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Number of Particles on Computational Time	57
4.8	(a) Effect of Maximum Particle Track Length on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Maximum Particle Track Length on Computational Time	58
4.9	(a) Effect of Polynomial Fit Order on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Polynomial Fit Order on Computational Time	59
4.10	(a) Effect of Number of Nearest Neighbours for Interpolation on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Number of Nearest Neighbours for Interpolation on Computational Time	59
4.11	(a) Effect of Minimum Error for SOR Convergence for Interpolation on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Minimum Error for SOR Convergence for Interpolation on Computational Time	60
4.12	(a) Effect of SOR Over-relaxation Factor on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of SOR Over-relaxation Factor on Computational Time	60
4.13	(a) Total Velocity Field around a Hemispherical Protuberance from SPIV Measurements (b) Reconstructed Pressure Coefficient Field from Velocity Fields using 2D Eulerian Approach	63
4.14	Distribution of Velocity Fluctuations over a Jet Flow and its Key Structures	64

4.15 **(a)** Total Velocity Field of an Unsteady Inviscid Jet Core from SPIV Measurements **(b)** Pressure Coefficient Field Reconstructed from Velocity Field using Time-Dependent Eulerian Approach **(c)** Pressure Coefficient Field Reconstructed from Velocity Field using Lagrangian Approach 66

List of Tables

3.1	Program Constants and Scale Factors used for Non-Dimensionalisation	30
4.1	Program Settings for 2D Eulerian Pressure Field Reconstruction of an Impinging Jet Potential Flow	46
4.2	Program Settings for Lagrangian Approach and SOR Poisson Solver	54
4.3	Mean Pressure Coefficient Error across Regions of a Flow around a Hemispherical Protuberance for the Eulerian and Lagrangian Approaches	56
4.4	Default Program Settings for Lagrangian Approach and SOR Poisson Solver	61

List of Abbreviations

ADF	Australian Defence Force
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lowy
DEHS	Di-ethyl-hexyl-sebacate
DSTG	Defence Science and Technology Group
FSR	Free-stream Region
GUI	Graphical User Interface
LES	Large Eddy Simulation
LSV	Laser Speckle Velocimetry
NS	Navier-Stokes
PIV	Particle Image Velocimetry
PPIV	Planar Particle Image Velocimetry
PTV	Particle Tracking Velocimetry
RAAF	Royal Australian Air Force
RAN	Royal Australian Navy
RANS	Reynolds Averaged Navier-Stokes
SOR	Successive-over-relaxation
SPIV	Stereoscopic Particle Image Velocimetry
SVR	Standing Vortex Region
UAV	Unmanned Aerial Vehicle
VPIV	Volumetric Particle Image Velocimetry
WMLES	Wall-Modelled Large Eddy Simulation
WR	Wake Region

List of Symbols

N_s	Source Density	
N_I	Image Density	
p_∞	Free-stream Static Pressure	Pa
U_∞	Free-Stream Velocity	m/s
U	Total Velocity	m/s
ρ	Density	kg/m^3
μ	Dynamic Viscosity	$Pa.s$
t	Time	s
\mathbf{u}	Velocity Vector	m/s
u	Horizontal (x) Velocity Component	m/s
v	Vertical (y) Velocity Component	m/s
w	Out-of-plane (z) Velocity Component	m/s
Re	Reynolds Number	
Δt	Integration and PIV Measurement Time-step	s
Δx	Spatial Step	m
ϵ_u	Velocity Measurement RMS Error	m/s
$\epsilon_{D\mathbf{u}/Dt,eul}$	Eulerian Material Acceleration RMS Error	m/s^2
$\epsilon_{D\mathbf{u}/Dt,lag}$	Lagrangian Material Acceleration RMS Error	m/s^2
$\boldsymbol{\Omega}$	Vorticity Vector	$1/s$
$\bar{\mathbf{u}}$	Local Mean Velocity Vector	m/s
\mathbf{u}'	Local Velocity Fluctuation Vector	m/s

p	Pressure	Pa
ω	SOR Over-Relaxation Parameter	
C_p	Pressure Coefficient	
k	Number of Nearest Neighbours	
N_t	Maximum Length of Lagrangian Particle Tracks	
N_p	Number of Lagrangian Particles	
m	Order of Polynomial Fit	
N_{max}	Maximum Number of SOR Iterations	
ϵ_{min}	Minimum Error for SOR Convergence	
ϕ	Velocity Potential Function	m^2/s
ψ	Velocity Streamline Function	m^2/s
ϵ_{C_p}	Pressure Coefficient Relative Error	

Chapter 1

Introduction

Advances in quantitative flow visualisation in the last few decades building upon high-speed camera and laser technology have led to a plethora of commercial solutions for Particle Image Velocimetry (PIV): a non-intrusive, planar or volumetric method of quantitatively measuring velocity fields in a desired flow. Raffel et al. (2018) argues that this measurement technique is invaluable for fundamental turbulence research and aerodynamic design optimisation. Particularly for resolving boundary layer profiles on aircraft and motor vehicles, as well as understanding transonic flow fields for passenger aircraft.

Modern PIV techniques have been used to measure flows around many different fundamental bodies, however the Australian Defence Science and Technology Group (DSTG) has invested time in studying flows over hemispherical protuberances, or wall-mounted hemispheres due to their prevalence as optical turret bodies on Unmanned Aerial Vehicles (UAV). This thesis is motivated by this DSTG objective, and builds upon PIV studies of this flow by Franck Bauer (2019) at the University of Sydney, by developing a program that can be used to determine the planar pressure field from planar velocity measurements by PIV techniques.

Pressure field reconstruction is important for surface load determination, allowing direct design optimisation of propellers, and wings in complex flow fields, and for estimation of fatigue-life and acoustic spectra. Planar reconstruction is achieved by solving the Navier-Stokes (NS) equations for the pressure field using planar velocity fields as initial and boundary conditions. Furthermore, there are many methods of achieving this as described by van Gent et al. (2017), however the main approaches; Eulerian and Lagrangian concern estimation of the velocity dependent material acceleration term in the NS equations, and

therefore the pressure gradient. The pressure term itself can also be solved in two distinct ways: direct spatial integration of its gradient or through solution of a Poisson equation.

This thesis investigates the design and implementation of a pressure field reconstruction program in the MATLAB programming language, looking at the Eulerian and Lagrangian approaches detailed in Section 2, and using a Poisson solver to determine the pressure from its gradient. The design of the program is detailed in Section 3 with descriptions and pseudo-code of the necessary functions to enable the reader to reproduce the program if desired. The program implementation is verified in Section 4 using velocity fields and pressure fields derived from analytical potential flows, and a CFD simulation. Furthermore, an analysis of the reconstructed pressure fields of empirical velocity fields from PIV measurements is performed.

In the interest of the reader and scientific contribution, the program is freely available for download at github.com, and it is the intention of the author to continue to update the program for the [OpenPIV](#) community.

Chapter 2

Literature Review & Background

2.1 Particle Image Velocimetry (PIV)

Particle Image Velocimetry or PIV, is an experimental method of non-intrusive quantitative flow visualisation used to obtain information about the velocity content of a particular flow field. This method has become very popular in the last few decades, having originated in the mid-twentieth century with the advent of advanced optical measurement techniques and synchronised flash-tube photography as postured by Grant (1997). However, this method is founded upon decades of flow visualisation research, which can be traced to the origins of humanity.

2.1.1 History of PIV

Flow visualisation is most commonly achieved by the addition of 'tracer' particles to a bulk fluid undergoing a form of motion, whether the tracer is a wooden log hurtling down a stream or smoke in a wind tunnel, the effect is the same: the structure of the flow field can be discerned. Qualitative flow visualisation, a familiar experience for university undergraduate students, has dominated most of human history. With prominent historical figures such as Leonardo da Vinci having recorded his perception of the fluid pathlines through flowing water in Figure 2.1. Even without understanding the quantitative content of a flow, qualitative flow visualisation was important during the evolution of fluid mechanics, with illustrations and eventually photographs of visualised flows being shared amongst the scientific community as described by Raffel et al. (2018). This drove research in the pursuit of understanding coherent fluid structures inside of chaotic turbulence.



Figure 2.1: Early illustration of turbulence: *Studies of Turbulent Water*. da Vinci (1510)

Velocimetry or the measurement of velocity in fluids before non-intrusive methods, was first achieved by pressure tapping surfaces or placing pitot tubes inside of a moving fluid. While pressure taps and pitot tubes are still used today as effective and accurate single point velocimetry devices along with modern hot wire or hot film probes, they contain errors that need to be corrected and impose additional dynamics to a flow field that would otherwise be absent as described in Barlow et al. (1999) for the case of wind tunnels. Therefore, it is difficult to measure velocity in regions with turbulence present, such as within a boundary layer or a wake, as there may be a significant contribution to the flow dynamics due to the measurement device itself. This limitation of intrusive velocimetry methods justifies the use of non-intrusive methods such as PIV.

Raffel et al. (2018) and Willert et al. (2019) reflect on the growth of photographic technology in the early twentieth century, that allowed prominent aerodynamicists such as Ludwig Prandtl to create non-intrusive 'particle streak images', essentially capturing the pathlines of a flow seeded with tracer particles such as aluminium powder by exposing film for a relatively long period of time (0.3 s). Willert et al. (2019) recounts the natural evolution of qualitative flow visualisation as progressing to motion pictures, demonstrating complex phenomena such as boundary layer formation and flow separation. A still image from Ludwig Prandtl's 1927 film: C1 is presented in Figure 2.2.

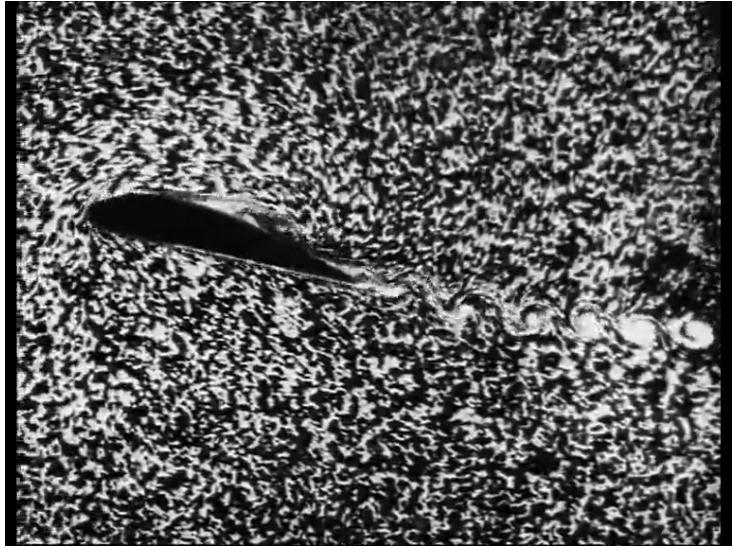


Figure 2.2: Separated Flow over an Aerofoil. Prandtl (1927)

Following the invention of the laser in 1960, a coherent light source could now be used to predictably interact with objects. As a result, new non-intrusive single-point velocimetry techniques such as Laser Doppler Velocimetry became possible. A technique described by Wang (1988) which relies on measuring the Doppler change in optical wavelength of light reflected from a moving target, in this case being a tracer particle. Furthermore, Grant (1997) recognises that the precursor to PIV, Laser Speckle Velocimetry (LSV) was enabled by this new technology.

Grant (1997) describes LSV as having derived from Speckle Metrology, a method that formed interference patterns from coherent light upon reflection from a solid surface. This method was used to measure surface roughness, and was applied to highly concentrated seeded flows, where the interference pattern would follow the particle motion and the flow velocity could be measured. Grant (1997) and Westerweel (1993) both agree that LSV is limited by its high tracer particle densities, which is difficult to realistically seed, and it introduces unintended dynamics, such as particle scattering indicative of two-phase flows. Furthermore, the flow sensitive is out-of-plane motion, common in turbulent flows that are inherently three-dimensional.

The growth of computer technology, and digital image processing in the 1980s enabled lower tracer particle density flows to be practically analysed according to Raffel et al. (2018), in contrast to the analogue Young's Fringe Analysis used for LSV described in Grant (1997). In essence, digital processing enabled measurements of many discrete particles, compared to analogue processing of an almost continuous particle field.

Adrian and Yao (1984) differentiates the different types of seeded flows by their source density, N_s : the average number of particles in the resolution cell or pixel, and image density, N_I : the average number of particles in the interrogation area or the sub-domains of the image that will be analysed using correlation methods. Westerweel (1993) describes LSV as a technique used for flows where $N_s \gg 1$ and $N_I \gg 1$, where individual particles cannot be discerned, and the images of these particles will overlap. PIV was the next evolution of the technique that measures lower tracer particle density flows with $N_s \ll 1$ and $N_I \gg 1$. Therefore, particles can be discerned under illumination from coherent laser light, however their motion is ambiguous and must be calculated using bulk statistical methods, such as auto-correlation and cross-correlation methods according to Grant (1997). Finally, Particle Tracking Velocimetry (PTV) measures low tracer particle density flows where $N_s \ll 1$ and $N_I \ll 1$. Individual particles that can be discerned and tracked randomly sample the flow and cannot provide velocity estimates at every pixel in the measurement plane. The observable differences between each type of seeded flow, and their respective measurement methods are summarised in Figure 2.3.

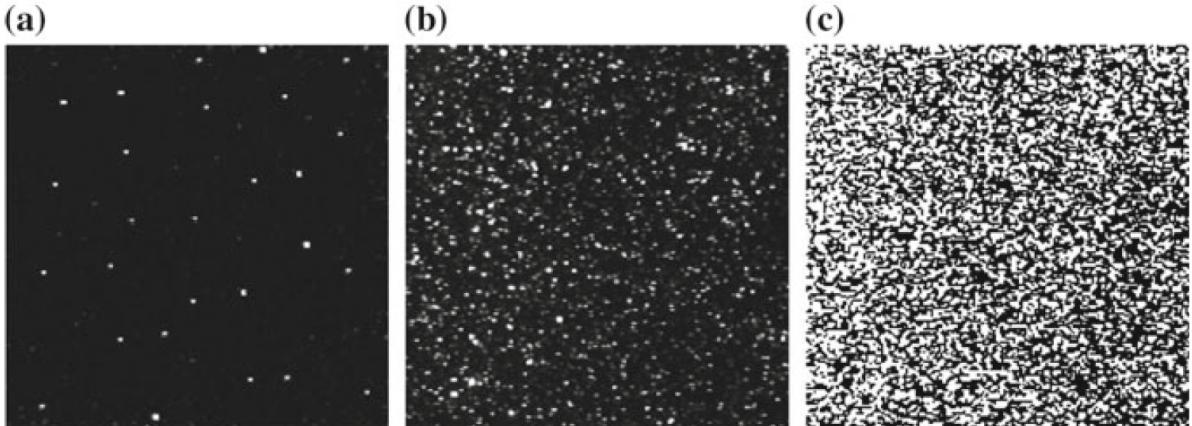


Figure 2.3: Tracer particle density determines the velocimetry method: (a) PTV (b) PIV (c) LSV. Raffel et al. (2018)

2.1.2 Modern PIV Techniques and Sources of Uncertainty

Adrian and Westerweel (2010) and Westerweel (1993) summarise PIV as a method that determines the instantaneous velocity field of a flow by statistically calculating the most probable displacement of tracer particles that follow motion in the flow. Yet, Adrian and Westerweel (2010) emphasises that PIV is a generic method of indirectly and non-intrusively measuring velocity, and it has many different forms, such as Planar PIV (PPIV), that measures the two-dimensional projection of velocities on a two-dimensional (2D) plane. Additionally, there is Stereoscopic PIV (SPIV) which measures all three ve-

locity components in a 2D plane, and Volumetric PIV (VPIV) that measures all three velocity components in a three-dimensional (3D) volume. Schroeder and Willert (2008) makes it clear that most PIV experiments are performed using PPIV methods, and that it has only recently been technologically and financially viable for academia and industry to use VPIV methods due to continued development of pulsed lasers, and high-speed cameras.

No matter the type of PIV used to perform experiments, Raffel et al. (2018) highlights the general experimental design and the multiple subsystems in Figure 2.4 that ultimately determine the uncertainty of the velocity measurements with respect to the true flow field, and the error associated with any properties that can be calculated from the field such as pressure. The following sections are a summary of the collated work of Raffel et al. (2018) and Adrian and Westerweel (2010).

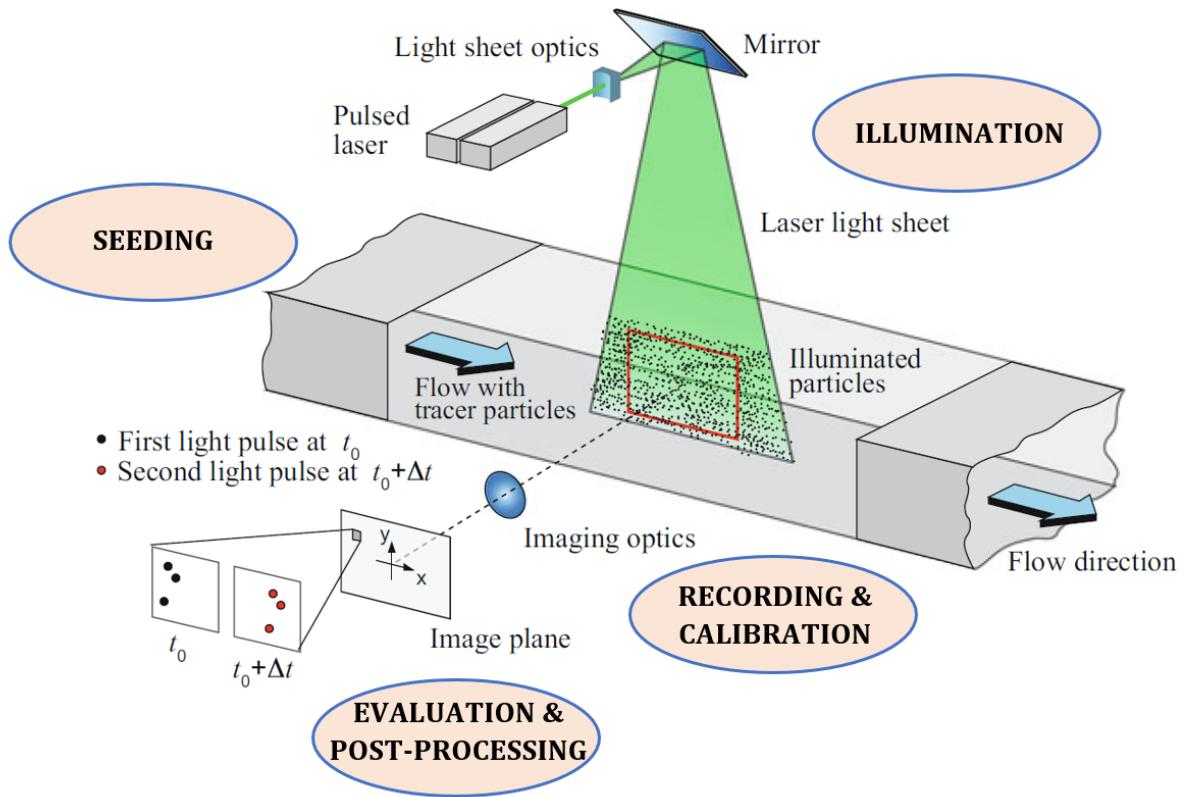


Figure 2.4: General PIV Experimental Design and Associated Subsystems. Raffel et al. (2018)

2.1.2.1 Seeding Uncertainty

Seeding is the process of filling a test section intended for PIV measurement with tracer particles. The distribution, density, and properties of the tracer particles are all important contributors to the uncertainty of PIV measurements. Firstly, the distribution of the

particles must be homogeneous, which in practice makes qualitative visualisation of the flow field impossible to the eye, however this is important for uniformly sampling the field when the velocity vectors are evaluated in software after the experiment has taken place. To achieve this homogeneity, the mechanism for seeding must be examined. Specifically, a small profile rake may be set up downstream of the test section and near the diffuser in a closed-loop wind tunnel to ensure that there is minimal disruption to the flow field.

The particle density is important to ensure that the velocity evaluation algorithms, specifically the cross-correlation or auto-correlation methods can work effectively by discerning individual particles and having sufficient surrounding particles for correlating bulk motion of the fluid. These requirements are equivalent to the $N_s \ll 1$ and $N_I \gg 1$ requirements described previously.

Finally, the physical properties of the particles determine their own dynamics inside the bulk fluid. The diameter of the particles determine their Mie scattering cross section for incident light and therefore the extent that they expose the camera sensor. This is important to tune using different atomisers to ensure that the chosen CMOS camera sensor can resolve an individual particle given the intensity of the chosen light source. The diameter and the density of the particle compound also determine the velocity lag or response time of the particles, which exists due to non-neutral buoyancy in the bulk fluid, and is particularly important for gas flows where the particles are usually denser than air. This will contribute significant uncertainty especially in strong velocity gradients, where the particles have delayed responses to the rapidly changing flow field. For most gas flow applications, oil or di-ethyl-hexyl-sebacate (DEHS) are suitable in density and can be atomised to the optimum droplet or particle diameter of $1 \mu\text{m}$. Furthermore, oil particles compared to solid metallic powders pose less of a safety hazard due to their organic nature, stay in the flow field for longer periods of time, and are stable in size under most conditions. Additionally, centrifugal motion related to vortex structures is important to consider when deciding on tracer particles, as dense particles can accumulate along the core of streamwise vortices and produce unfavourable inhomogeneities in the particle field.

2.1.2.2 Illumination Uncertainty

The illumination or light source must be chosen to ensure that the particles can be resolved by the camera within a certain illumination period without streaking. Nd:YAG solid-state lasers are usually chosen as light sources for PIV experiments due to their high

maximum peak power, and strong mechanical and thermal properties aiding their reliability and durability. They can also be designed as Q-switched double oscillator systems meaning that they can distribute their power over two short pulses, useful for high temporal resolution of the particle image pairs, and therefore the flow field. The time delay between these laser pulses must be tuned to enable sufficient displacement of the particles can be resolved by the camera, and short enough to avoid too many particles leaving the measurement plane. The double oscillator system also ensures that the laser optics do not absorb too much energy compared to a single oscillator system where thermal lensing would occur and defocus the beam.

Furthermore, lasers can be tuned to have an approximately Gaussian or top-hat beam profile, required for illuminating the measurement plane homogeneously without gaps. However, this is dependent on the manufacturing of the internal laser optics, which may introduce mirror imperfections or lens aberrations that can defocus the beam and change its beam profile. Ambient conditions also affect the optics, with temperature acting to distort the cylindrical lens used for creating the laser sheet, and the internal laser optics if they are not cooled.

Finally, the laser sheet must be aligned to the desired measurement plane, especially needed if the measurements are to be verified by Computational Fluid Dynamics (CFD) Large Eddy Simulations (LES). The laser must also be focused and aligned along the guiding optics it will follow to ensure sufficient intensity where it is shaped into a laser sheet.

2.1.2.3 Recording and Calibration Uncertainty

The digital camera must be chosen primarily based on its resolution and its frame rate frequency, which determine the required magnification during the PIV experiment and the maximum speed that the wind tunnel can be run as the camera must be able to sample particle motion at twice its frequency, satisfying the Nyquist sampling theorem. Significant uncertainty of PIV measurements is dependent on the particle image diameter, an effective measure of how large particles appear in the interrogation areas and the importance of the resolution limit or Rayleigh criterion. This is a function of the camera magnification and the size of the measurement plane.

For single camera set-ups or PPIV methods, the camera must be aligned and focused 90° to the measurement plane so that projected velocities can be accurately measured,

however misalignment is solved using SPIV methods. SPIV is also useful for accurately measuring turbulent flows with significant out-of-plane motion where PPIV would fail. Camera lens aberrations due to manufacturing errors, glare, and sensor noise due to multiple scattered photons and thermal electron generation also add uncertainty to PIV measurements.

Installation errors such as misalignment, lens aberrations, and wind tunnel window aberrations must be calibrated for, so that experiments can be repeatable. This is achieved using a standardised calibration plate, however if any parts of the experiment change in its duration such as deformation of wind tunnel windows due to pressure differences, or the calibration plate has manufacturing errors, calibration will contribute an uncertainty to the measurements themselves.

2.1.2.4 Evaluation and Post-Processing Uncertainty

The correlation methods used to calculate the displacement and velocity of particles from particle image pairs contributes to the total uncertainty of the velocity measurements. If single frame/double exposure images are captured from the experiment, auto-correlation methods must be used to determine displacement, however the direction is ambiguous. Double frame/single exposure images solve this problem by using cross-correlation methods. These methods statistically measure the similarity between two interrogation areas for a given shift, the most probable displacement for each particle is determined from this, and it has intrinsic statistical error associated with it. The evaluation algorithms determine the size of the interrogation areas based on the optimum particle image diameter of 3 pixels, and this size controls the range of dynamic scales that can be resolved. This is important to consider in turbulent flows that have a characteristic Kolmogorov length scale where turbulent eddies dissipate. Finally, the particle image diameter controls whether peak-locking occurs, a statistical phenomenon where the displacements take inaccurate integer values due to the nature of interpolation algorithms, observed as defined peaks in a displacement histogram. This will occur if the particle image diameter is too small.

Post-processing is used to remove poorly resolved velocity vectors that are commonly found on the border of the measurement plane. These are removed based on established relationships that correlate the number of particle images within an interrogation area and the probability of invalid velocity vectors. Particle image displacement is also used as a metric for measurement invalidity, where the displacement is larger than the interrogation area or the light sheet thickness. These invalidity algorithms are capable of producing

false positives and thus removing otherwise accurate velocity vectors and contributing to measurement uncertainty with reduced data.

2.1.3 Current Applications of PIV

PIV is an important non-intrusive research tool for complex unsteady flow phenomena, and has been used by academia for fundamental turbulence research since the 1980s. Within the last 20 years it has been feasible for use by industry as a research tool, and now that VPIV methods are available, resolving 3D unsteady structures grants industry the ability to validate CFD codes, and optimise aerodynamic design to a new standard. Raffel et al. (2018) and Schroeder and Willert (2008) detail important applications of PIV that are likely to become ubiquitous with decreasing costs of PIV equipment.

Study of the boundary layer has been leveraged by the aerospace and motor vehicle industry for predicting and controlling turbulent boundary layer transition and thus controlling lift and drag characteristics of these vehicles. PIV has been used to study the evolution of boundary layer instabilities beginning from 2D Tollmien-Schlichting waves, which form downstream vortices upon amplification by 3D spanwise disturbances. This transition is tied to surface roughness, macroscopic surface shape, Reynolds number, and Mach number, all of which can be controlled to optimise lift and drag characteristics for better performance and handling. Shock wave turbulent boundary layer interactions have also been studied using PIV, where traditional intrusive methods or single point non-intrusive methods are either too coarse or disrupt the flow field. These interactions are important at Mach numbers where the oblique shock angle is low enough where the shock front begins to interact with the boundary layer, which expands with heat.

Transonic flows known for their complex and sensitive flow field have been researched using PIV, as this regime is shared by commercial airliners whose manufacturers aim to delay normal shock formation, which acts to create unfavourable heating and pressure losses. Gas-turbine engine and propeller manufacturers also share this design goal.

Propellers and rotors suffer from blade vortex interactions that create significant noise pollution. The underlying flow field generating this noise, and the measurement of the acoustic spectrum can both be determined using PIV. The pressure field can be reconstructed from the velocity field by the Navier-Stokes equations under some fundamental assumptions of dimension, fluid properties, and Reynolds number. This pressure field under a transformation reveals the spectrum of pressure fluctuations and therefore the

acoustic spectrum. Furthermore, the pressure field allows for load determination, useful for design optimisation of propellers and rotors immersed in complex rotary flow fields, and for structural fatigue life estimation.

2.2 Hemispherical Protuberances

Hemispherical protuberances are hemispheres protruding from a flat plane, which may be the shape of optical turrets on aircraft, rivet heads on aircraft structures, and hemispherical stadiums or civilian buildings. They are becoming more common-place with significant investment from international governments and specifically the Australian government and Australian Defence Force (ADF) in Unmanned Aerial Vehicles (UAVs), which use optical turrets for surveillance and reconnaissance. Examples from Royal Australian Navy (2021) and Royal Australian Air Force (2021) include the ScanEagle used by the Royal Australian Navy (RAN), and the MQ-4C Triton operated by the Royal Australian Air Force (RAAF). These investments have led the Australian Defence Science and Technology Group (DSTG) to conduct studies investigating the flow phenomena around a hemispherical protuberance, and to encourage similar studies from Australian universities including this thesis.

Franck Bauer et al. (2018) is a study from DSTG describing flow around a hemispherical protuberance as having significant unsteady flow structures that could negatively impact the flight envelope and performance of aircraft using such a structure. The structures of a junction flow or flow around a hemispherical protuberance are presented in Figure 2.5. The follow-up study from Franck Bauer (2019) supports the use of SPIV in fundamental research of the flow features discussed in Franck Bauer et al. (2018), and it experimentally confirmed the hemispherical protuberance velocity flow field only previously described by CFD studies. Furthermore, the unsteady flow structures are linked to pressure fluctuations that cause aero-elastic vibration in optical turrets impacting image quality and laser tracking accuracy according to Gordeyev and Jumper (2010). Savory and Toy (1986) supports the importance of measuring the pressure field around hemispherical buildings to assess occupant discomfort, ventilation, and predicting effects on downstream buildings.

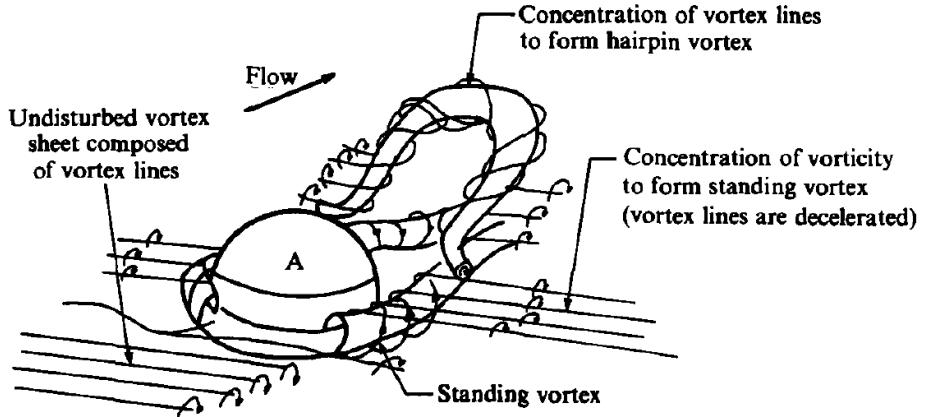


Figure 2.5: Structures of a Flow around a Hemispherical Protuberance. Acarlar and Smith (1987)

Acarlar and Smith (1987) describes the unsteady structures around a hemispherical protuberance as consisting of a stable standing vortex that forms upstream of the stagnation point and evolves into a horseshoe vortex that wraps around the sides of the hemisphere observed in Figure 2.5. This double vortex begins to stretch as it moves out of the velocity gradient in the boundary layer increasing its vorticity concentration and attracting the two vortices together, analogous to two opposing current carrying wires. The attraction eventually leads to viscous interaction and dissipation as the vortices expand and fade.

The 'hairpin' vortex system sheds periodically due to an interaction between the separated shear layer from the wake and the freestream flow, and moves downstream, observed in Figure 2.6. Franck Bauer et al. (2018) reveals that this causes significant oscillations in the pressure distribution of the ground plane and the hemisphere. This complex system is dependent on boundary layer thickness, laminar or turbulent state, and freestream turbulent intensity.

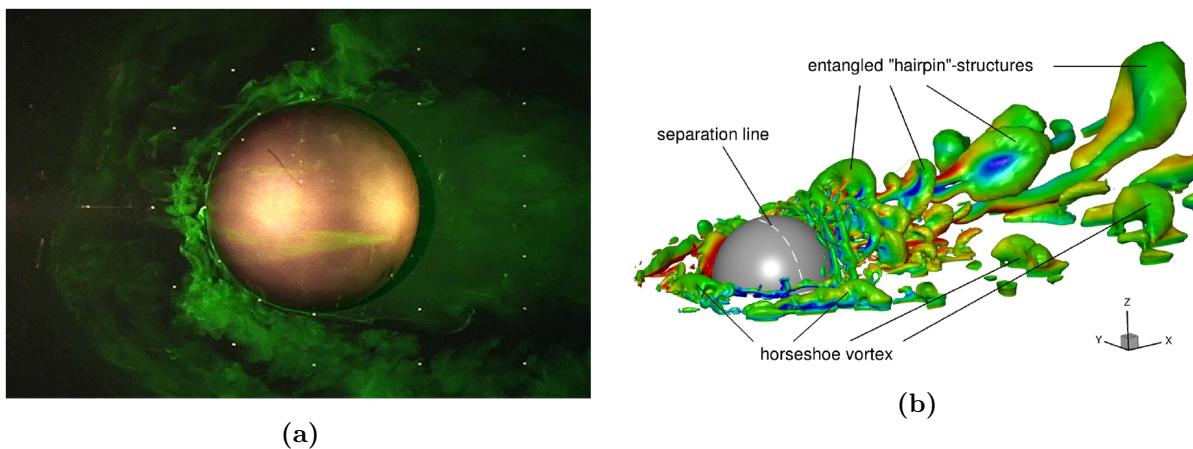


Figure 2.6: (a) Water Flow Visualisation of Hemispherical Protuberance. Franck Bauer et al. (2018) (b) CFD Large-Eddy Simulation of a Hemispherical Protuberance. Wood et al. (2016)

2.3 Pressure Field Reconstruction

It has been established that pressure measurements derived from PIV data is extremely useful for modern research in academia and industry, and this transformation of data has its roots in Bernoulli's principle.

$$p = p_\infty + \frac{1}{2} \rho (U_\infty^2 - U^2) \quad (2.1)$$

Eq. 2.1 shows that there is a non-linear relationship between pressure and velocity. In fact, this is the ideal case for an incompressible, irrotational, steady and inviscid flow. Pressure-velocity relationships only become more complex for realistic flows, and the first documented paper to use such a pressure reconstruction method is Schwabe (1935), where the author calculated by hand the pressure field using an unsteady Bernoulli relation, from photographic images of tracer particles in a flow around a circular cylinder. The author's results are shown in Figure 2.7.

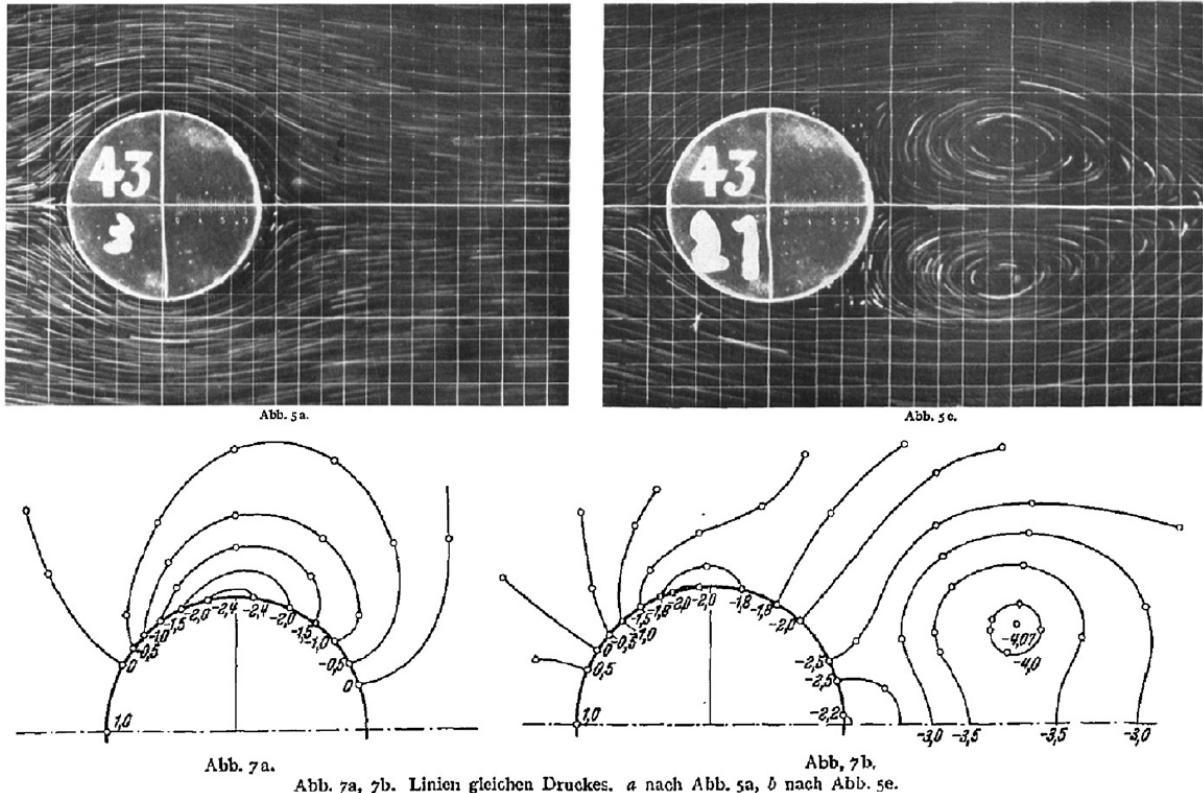


Figure 2.7: Earliest Results of Pressure Reconstruction from Particle Streak Images. Schwabe (1935)

Realistic pressure calculations are derived from solutions of the Navier-Stokes (NS) mo-

mentum equations for a continuum fluid with known velocity fields and pressure boundary conditions. The NS equations are summarised in Eq. 2.2, and describe the pressure gradient in a flow as having contributions (in order) from the material acceleration and the viscous stresses.

$$\nabla p = -\rho \frac{D\mathbf{u}}{Dt} + \mu \nabla^2 \mathbf{u} \quad (2.2)$$

2.3.1 Material Acceleration Estimation

van Oudheusden (2013) reveals that the material acceleration can be determined using three primary approaches: direct determination by tracking tracer particles in the flow field (applicable when the data is derived from PTV methods), Lagrangian pseudo-tracking by reconstructing trajectories of imaginary particles in the velocity flow fields over time, and the Eulerian approach, which is to decompose the material acceleration into temporal and spatial derivatives, and calculate it directly. The viscous stresses are simpler to analyse, and can be calculated directly from second order finite difference schemes.

The Lagrangian, Eulerian and additional approaches to calculating the material acceleration term for PIV measurements will be discussed in the following sections.

2.3.1.1 Eulerian Approach

This approach assumes the fluid volume to be a continuous field with evolving macroscopic field properties such as density, temperature, velocity, and pressure. This assumption is computes the material acceleration in a stationary reference frame, and effectively simplifies it by separating the temporal local acceleration and the spatial convective accelerations in Eq. 2.3. This approach is common for CFD solvers.

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = \left(\frac{\partial}{\partial t} + u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y} + w \frac{\partial}{\partial z} \right) \mathbf{u} \quad (2.3)$$

Imaichi and Ohmi (1983) uses this approach in one of the earliest examples of digital PPIV, where the velocities of aluminium powder tracer particles flowing around a circular cylinder were measured from particle streak photographs taken over 0.9 s. 2000-4000 particle paths per photograph were interpolated onto a rectangular computational grid,

and analysis was performed by a crude "digitising tablet". Under the assumption of steady 2D flow for the circular cylinder, the out of plane velocity component and the unsteady term velocity are neglected.

Unfortunately, these simplifications led to significant error in the final pressure solution attributed to the least-squares interpolation of the measurements to the computational grid around the cylinder itself, and due to the neglect of the unsteady term. This particularly affected the circular cylinder in the $Re = 100$ and $Re = 200$ regimes where turbulent Kármán vortex streets and twin vortex systems formed in the wake of the cylinder. Gurka et al. (1999) improved upon this digital PPIV method by incorporating the unsteady term in the 2D formulation. However, for real 3D flows measured by SPIV methods, the out of plane velocity component significantly contributes to the pressure gradient as supported by van Oudheusden (2013), and provides a field-dependent $w \frac{\partial}{\partial z}$ error term that cannot be modelled without VPIV methods which resolve velocity gradients across the out of plane direction, z .

The Eulerian approach is limited by the truncation error associated with the finite difference schemes that approximate the temporal and spatial derivatives of the velocity field. The temporal derivative is constrained when convection dominated phenomena is present as described by van Oudheusden (2013), which has a smaller Eulerian timescale than its Lagrangian counterpart. Thus, the temporal resolution must be higher for the Eulerian approach. Furthermore, the precision error due to the propagation of velocity measurement uncertainty (estimated RMS), ϵ_u contributes to the total error of the Eulerian material acceleration. This total error comprising the truncation and precision errors is calculated using a first order forward time and second order central space scheme in Eq. 2.4. It is derived from the linear error propagation procedure used by van Oudheusden (2013), where Δt is the time-step, $\Delta x = \Delta y$ is the spatial step of the scheme, and $U = |\mathbf{u}|$ is the total velocity.

$$\epsilon_{D\mathbf{u}/Dt,eul} \approx \epsilon_u \sqrt{\frac{2}{\Delta t^2} + \frac{1}{2} \frac{U^2}{\Delta x^2} + |\nabla U|^2} \quad (2.4)$$

2.3.1.2 Lagrangian Approach

The Lagrangian approach treats the fluid as an ensemble of individual particles, each with position, \mathbf{x} and velocity, \mathbf{u} states. The acceleration of a particle along its unique track

represents a portion of the material acceleration information for the entire fluid volume. van Gent et al. (2018) proposes propagating the imaginary particles that are initially located at the grid points of the computational domain across the measured velocity field both forwards and backwards in time depicted in Figure 2.8. The material acceleration at that initial time-step is calculated as the first temporal derivative of the particle velocity.

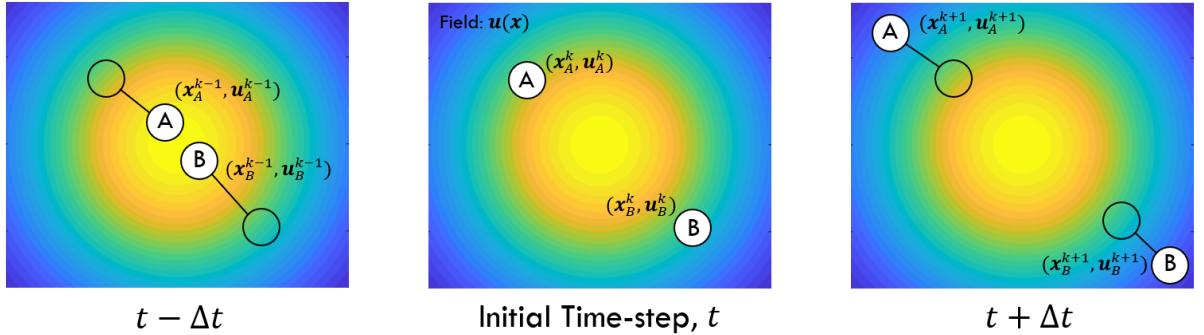


Figure 2.8: Propagation of Imaginary Particles A and B across an Example Velocity Field

For particle track construction van Gent et al. (2018) suggests four methods. Firstly, a simple forward Euler scheme with one integration per measurement time-step presented in Eq. 2.6 (Low-order B). One integration will not satisfy the Courant-Friedrichs-Lowy (CFL) condition in Eq. 2.5, potentially required for numerical stability of the underlying non-linear NS equations.

$$\Delta t \leq \frac{\Delta x}{u + v + w} \quad (2.5)$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{u}^k \Delta t \quad (2.6)$$

An improvement is two integrations per measurement time-step for the forward Euler scheme (Low-order A), which satisfies the CFL condition and results in lower RMS error in the pressure solution. Furthermore, a second order Heuns integration scheme is proposed with 2 integrations per measurement time-step (2nd-order), and finally a Runge-Kutta fourth order integration scheme in combination with cubic spline interpolation is proposed (High-order). Figure 2.9 from van Gent et al. (2018) reveals that all the track construction schemes performed similarly in regions of the flow field where there was a shear-layer, reattachment, and a freestream flow, meaning that the RMS error was dominated by the underlying velocity measurement uncertainty due to the SPIV method.

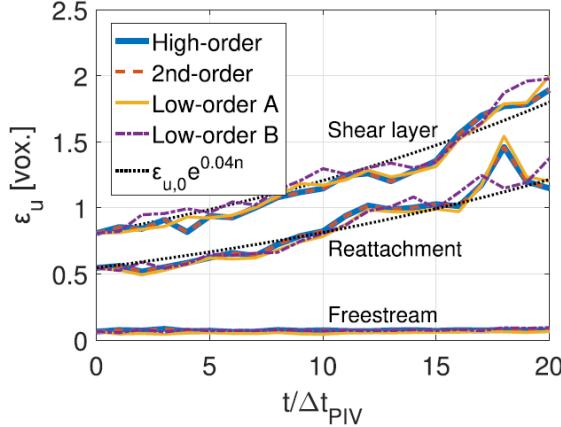


Figure 2.9: RMS Error in Streamwise Velocity for the Particle Track Construction Schemes.
van Gent et al. (2018)

With the track constructed van Gent et al. (2018) offers two methods for calculating the first temporal derivative: centred differencing (CD) and least-squares polynomial regression (LS). Upon analysis by van Gent et al. (2018) in Figure 2.10, least-squares linear regression is found to be most advantageous as it has meaningfully less noise amplification than centred differencing for more than three PIV measurement time-steps, and it has a lower material acceleration RMS error than centred-differencing for any number of measurement time-steps, and cubic polynomial regression for less than 16 time-steps (for shear-layer, reattachment, and freestream regions). The material acceleration error follows a non-linear trend first rapidly decreasing as the velocity measurement error has reduced influence with more time-steps considered, then a minimum error is reached, and the error slowly rises due to the accumulation of truncation errors and track construction errors which manifests as excessive smoothing. The optimal track length or number of measurement time-steps for the linear regression scheme is 11.

van Oudheusden (2013) determines that the RMS error of the material acceleration calculated by the Lagrangian approach can be approximated as Eq. 2.7, where the U term in the Eulerian counterpart is now absent. Therefore, the Lagrangian approach is less sensitive to advection (and convection) dominated phenomena. Furthermore, the Lagrangian approach was found to be the most accurate approach for noisy velocity fields when compared to Eulerian and single measurement approaches such as Taylor's Hypothesis and Instantaneous Vortex-in-Cell in van Gent et al. (2017). This study used a Zonal Detached Eddy Simulation (ZDES) to verify the pressure solutions of these approaches by creating synthetic particle images with ideal and noisy particles. The Lagrangian approach was found to be more tolerant to noise, and had a pressure error field that evolved gradually

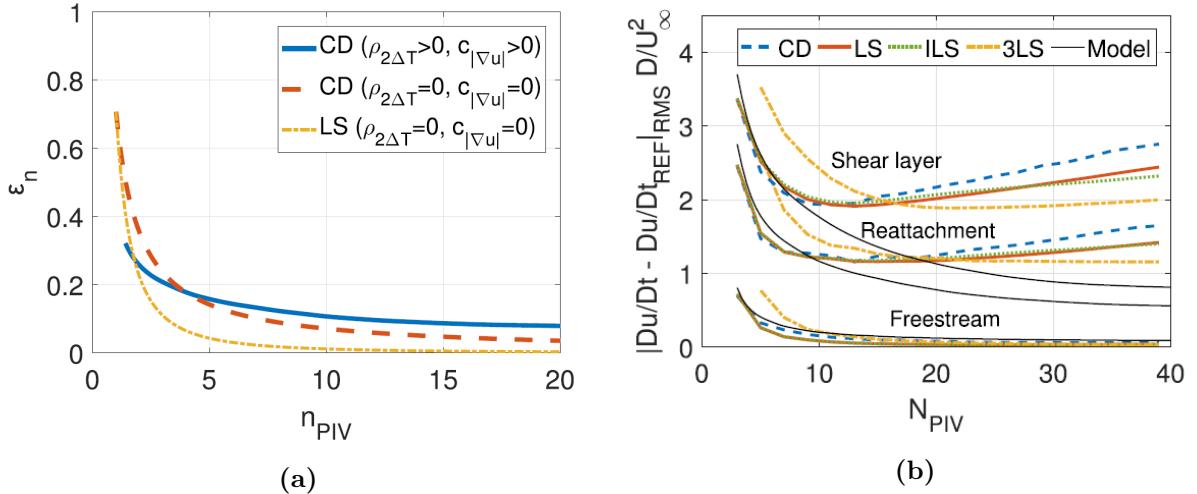


Figure 2.10: (a) Noise Amplification for Material Acceleration Approaches (b) RMS Error in Material Acceleration for Material Acceleration Approaches and Track Lengths. van Gent et al. (2018)

over time in comparison to the Eulerian approach which had a highly fluctuating pressure error field. These results are due to the Lagrangian approach having more access to temporal information, compared to the Eulerian's two time-steps. However, overall all of the methods were not critically sensitive to noise in the PIV measurements.

$$\epsilon_{Du/Dt, \text{lag}} \approx \epsilon_u \sqrt{\frac{1}{2\Delta t^2} + \frac{1}{2} |\nabla U|^2} \quad (2.7)$$

2.3.1.3 Single Measurement Approaches

These approaches only require a single velocity field from PIV measurements to determine the material acceleration, as they replace temporal information with spatial information.

der Kindere et al. (2019) describes the Taylor's Hypothesis approach as using Reynolds averaging (Eq. 2.8) to assume that turbulent fluctuations, \mathbf{u}' advect with the mean flow, $\bar{\mathbf{u}}$. This is not true for strongly turbulent flows where the turbulent fluctuations are on the same order of magnitude as the local mean velocity and Reynolds averaging does not hold. The Eulerian approach is implicit to this method, however as the mean velocity is steady the resultant material acceleration is purely dependent on spatial information as in Eq. 2.9.

$$\mathbf{u}(t) = \bar{\mathbf{u}} + \mathbf{u}'(t) \quad (2.8)$$

$$\frac{\partial \mathbf{u}'}{\partial t} = -(\bar{\mathbf{u}} \cdot \nabla) \mathbf{u}' \rightarrow \frac{D\mathbf{u}}{Dt} = -(\bar{\mathbf{u}} \cdot \nabla) \mathbf{u}' + (\mathbf{u} \cdot \nabla) \mathbf{u} \quad (2.9)$$

The other approach is Instantaneous Vortex-in-Cell described by Schneiders et al. (2016) which calculates the velocity temporal derivative from the solution of a Poisson equation involving the curl of the inviscid and incompressible vorticity transport equation (Eq. 2.10), which itself is approximated to be spatially dependent. The result is a material acceleration that is purely spatially dependent shown in Eq. 2.11.

$$\frac{\partial \boldsymbol{\Omega}}{\partial t} = (\boldsymbol{\Omega} \cdot \nabla) \mathbf{u} - (\mathbf{u} \cdot \nabla) \boldsymbol{\Omega} \quad (2.10)$$

$$\nabla^2 \frac{\partial \mathbf{u}}{\partial t} = -\nabla \times \frac{\partial \boldsymbol{\Omega}}{\partial t} \rightarrow \frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \quad (2.11)$$

2.3.2 Pressure Estimation

Upon processing of the velocity measurements to obtain the material acceleration, the pressure field can be estimated from its gradient in the NS equations. van Oudheusden (2013) presents two methods for obtaining the pressure field from its gradient, firstly direct integration, and secondly solution of a Poisson equation.

2.3.2.1 Direct Integration

As one of the earliest pressure reconstruction methods, Imaichi and Ohmi (1983) used a direct integration scheme under the assumptions of steady 2D flow, incompressibility, and constant viscosity. This Eulerian scheme represented by Figure 2.11 involved numerically integrating along the two orthogonal directions (x and y) in the computational domain using an Euler integration scheme starting from an initial point whose pressure value was estimated using Bernoulli's principle in Eq. 2.1.

The scheme requires the 2D steady incompressible pressure gradients shown in Eq. 2.13 and 2.14 and the associated second order finite first and second derivatives of velocity components exemplified by Eq. 2.12. To calculate the pressure at points (i,j) diagonal to the initial point their pressure values were estimated from the average of the two

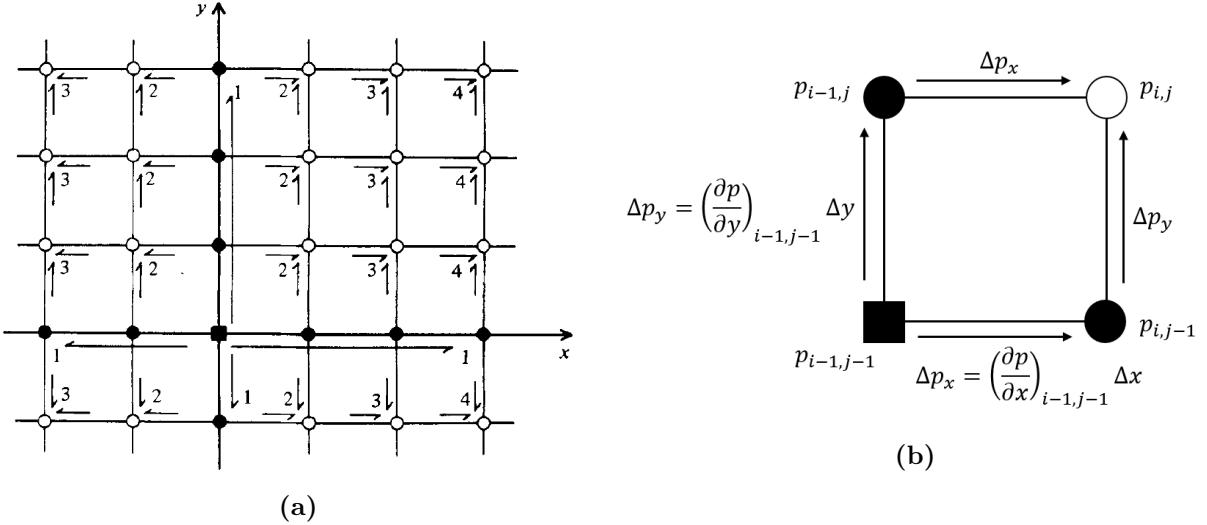


Figure 2.11: Two Point Average Direct Pressure Integration Scheme. Imaichi and Ohmi (1983)

integration directions in Eq. 2.15.

$$\frac{\partial u}{\partial x} \approx \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} \quad \frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} \quad (2.12)$$

$$\frac{\partial p}{\partial x} = -\rho \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (2.13)$$

$$\frac{\partial p}{\partial y} = -\rho \left(u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (2.14)$$

$$p_{i,j} = \frac{1}{2} (p_{i-1,j} + \Delta p_x + p_{i,j-1} + \Delta p_y) \quad (2.15)$$

Baur and Kongeter (1999) improves this scheme with a four point average for the incompressible unsteady 3D NS equations shown in Figure 2.12a. This method produces a solution that is dependent on the direction of integration, usually chosen to be the direction of the mean flow.

To improve this, Liu and Katz (2006) proposes a spatial-marching scheme depicted in Figure 2.12b that integrates along paths radiating from a point chosen on a virtual boundary outside of the computational domain, creating an omni-directional integration scheme. This scheme is loosely summarised by Eq. 2.18 for diagonally marching across a 2D domain with incompressible, and constant viscosity assumptions. Furthermore, it is suitable for the Eulerian or Lagrangian approaches.

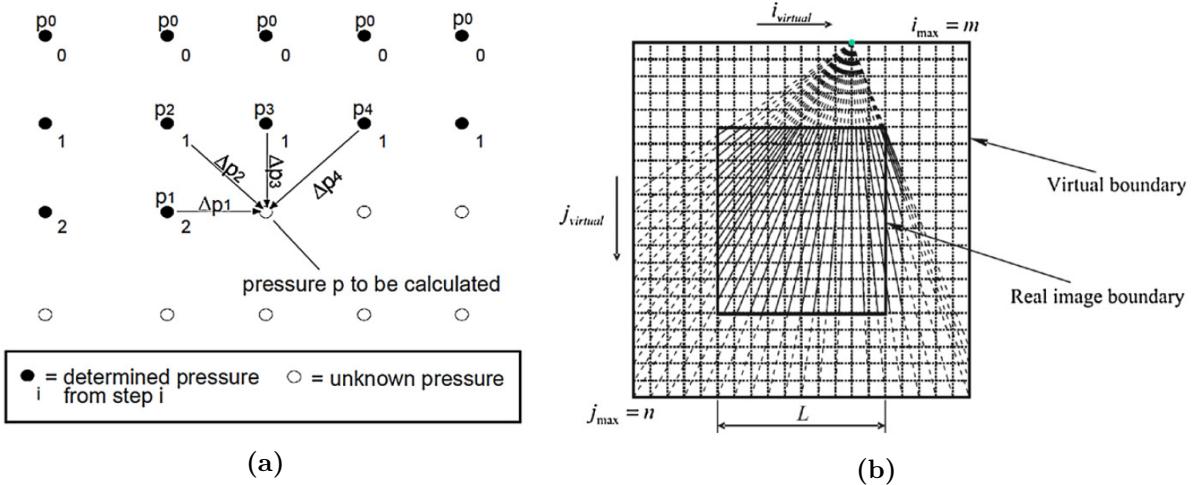


Figure 2.12: (a) Four Point Average Direct Pressure Integration Scheme. Baur and Kongeter (1999) (b) Omni-directional Direct Pressure Integration Scheme. Liu and Katz (2006)

$$\frac{\partial p}{\partial x} = -\rho \frac{Du}{Dt} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (2.16)$$

$$\frac{\partial p}{\partial y} = -\rho \frac{Dv}{Dt} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (2.17)$$

$$p_{i,j} = p_{i-1,j-1} + \left(\frac{\partial p}{\partial x} \right)_{i-1,j-1} \Delta x + \left(\frac{\partial p}{\partial y} \right)_{i-1,j-1} \Delta y \quad (2.18)$$

van Oudheusden (2013) describes the direct integration method of pressure estimation as having the benefit of prescribing an initial pressure point that can be placed in a relatively undisturbed portion of the velocity field compared to the Poisson method that requires true boundary conditions that will significantly contribute to the error of the pressure solution. However, van Oudheusden (2013) reveals that direct integration is more sensitive to noise, and overall the omni-directional and Poisson methods perform equally well.

2.3.2.2 Poisson Solution

By taking the divergence of the pressure gradient vector, a single Poisson equation is generated as in Eq. 2.19. This is beneficial as the Poisson equation is a general second order partial differential equation that has a variety of finite difference algorithms for its solution.

$$\nabla \cdot \nabla p = \nabla^2 p = \nabla \cdot \left(-\rho \frac{D\mathbf{u}}{Dt} + \mu \nabla^2 \mathbf{u} \right) \quad (2.19)$$

Gurka et al. (1999) uses this method to significantly simplify the pressure estimation for a 2D incompressible Eulerian approach for a laminar constricted pipe flow, and an air jet impinging on a surface measured using PPIV methods. Using the continuity equation in Eq. 2.20 for an incompressible 2D fluid, the temporal velocity derivative and the viscous stresses can be neglected, leaving the pressure Poisson equation purely dependent on spatial information in Eq. 2.21.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \Rightarrow \nabla \cdot \mathbf{u} = 0 \quad (2.20)$$

$$\begin{aligned} \nabla^2 p &= \nabla \cdot \left(-\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \mu \nabla^2 \mathbf{u} \right) \\ &= -\rho \left(\frac{\partial}{\partial t} (\nabla \cdot \mathbf{u}) + \nabla \cdot (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \mu \nabla^2 (\nabla \cdot \mathbf{u}) \\ &= -\rho \nabla \cdot ((\mathbf{u} \cdot \nabla) \mathbf{u}) \end{aligned}$$

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = -\rho \left(\left(\frac{\partial u}{\partial x} \right)^2 + 2 \frac{\partial u}{\partial y} \frac{\partial v}{\partial x} + \left(\frac{\partial v}{\partial y} \right)^2 \right) \quad (2.21)$$

For a Eulerian or Lagrangian incompressible approach to velocity fields measured using SPIV methods, where the in-plane divergence of the velocity will be non-zero (mass is not conserved across a plane in a fluid volume), the resultant Poisson equation cannot be simplified as before. van Oudheusden (2013) shows that the equation will contain temporal information and the viscous stresses. The Eulerian approach is detailed in Eq. 2.22, and the Lagrangian approach in Eq. 2.23. The in-plane gradient operator is represented as ∇_{xy} , and in-plane divergence is $\text{div}_{xy} = \nabla_{xy} \cdot \mathbf{u}$.

$$\begin{aligned}
\nabla_{xy}^2 p_{eul} &= \nabla_{xy} \cdot \left(-\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \mu \nabla^2 \mathbf{u} \right) \\
&= -\rho \left(\left(\frac{\partial u}{\partial x} \right)^2 + 2 \frac{\partial u}{\partial y} \frac{\partial v}{\partial x} + \left(\frac{\partial v}{\partial y} \right)^2 \right) \\
&\quad - \rho \left(\frac{\partial \text{div}_{xy}}{\partial t} + u \frac{\partial \text{div}_{xy}}{\partial x} + v \frac{\partial \text{div}_{xy}}{\partial y} \right) \\
&\quad + \mu \left(\frac{\partial^2 \text{div}_{xy}}{\partial x^2} + \frac{\partial^2 \text{div}_{xy}}{\partial y^2} \right) \\
&\quad - \rho \left(\frac{\partial w}{\partial x} \frac{\partial u}{\partial z} + \frac{\partial w}{\partial y} \frac{\partial v}{\partial z} + w \frac{\partial \text{div}_{xy}}{\partial z} \right)
\end{aligned} \tag{2.22}$$

$$\begin{aligned}
\nabla_{xy}^2 p_{lag} &= \nabla_{xy} \cdot \left(-\rho \frac{D\mathbf{u}}{Dt} + \mu \nabla^2 \mathbf{u} \right) \\
&= -\rho \left(\frac{\partial}{\partial x} \left(\frac{Du}{Dt} \right) + \frac{\partial}{\partial y} \left(\frac{Dv}{Dt} \right) \right) + \mu \left(\frac{\partial^2 \text{div}_{xy}}{\partial x^2} + \frac{\partial^2 \text{div}_{xy}}{\partial y^2} \right)
\end{aligned} \tag{2.23}$$

These equations reveal that for 3D flows, the pressure field in a plane will have intrinsic error that cannot be resolved due to the absence of out-of-plane information, which is also true for the direct integration method. However, van Oudheusden (2013) argues that a moderate degree of out-of-plane motion will not significantly affect the accuracy of the pressure solution.

The Poisson equation is an elliptic partial differential equation, and Hoffman (2001) offers three iterative methods: Jacobi iteration, Gauss-Seidel iteration, and Successive-over-relaxation (SOR). SOR is the general method, where Gauss-Seidel is the case where the over-relaxation factor, ω equals one. Gurka et al. (1999) uses the SOR method for pressure estimation, which results in relatively accurate pressure results when compared to the inviscid solution for a constricted tube flow. The Poisson equation can also be constructed as a matrix equation and solved via a least-squares method as in van Gent et al. (2018) and Jeon et al. (2015).

The SOR method is constructed using a second-order central difference scheme according to Hoffman (2001) in Eq. 2.24, where $f(x, y)$ is the source term containing the velocity field information. Iteration happens concurrently, where already updated points denoted by the superscript $k+1$ are used to update the next point in the domain as shown in Figure 2.13, in comparison to Jacobi iteration where the entire field is updated using unchanged point values before an iteration is complete. The optimal over-relaxation parameter that forces the pressure solution towards convergence within the least number of iterations is

also presented in Eq. 2.25 from Hoffman (2001). Here, the horizontal and vertical spatial steps are equivalent, $\Delta x = \Delta y$, and the number of spatial steps in each direction are L_x and L_y .

$$\begin{aligned}\nabla_{xy}^2 p &= \frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \approx \frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{\Delta x^2} + \frac{p_{i,j+1} - 2p_{i,j} + p_{i,j-1}}{\Delta y^2} = f(x, y) \\ p_{i,j}^{(1)} &= \frac{1}{4}(p_{i+1,j}^k + p_{i-1,j}^k + p_{i,j-1}^k + p_{i,j+1}^k - \Delta x^2 f(x, y)) \\ p_{i,j}^{k+1} &= p_{i,j}^k + \omega(p_{i,j}^{(1)} - p_{i,j}^k) \\ &= (1 - \omega)p_{i,j}^k + \frac{\omega}{4}(p_{i+1,j}^k + p_{i-1,j}^k + p_{i,j-1}^k + p_{i,j+1}^k - \Delta x^2 f(x, y))\end{aligned}\quad (2.24)$$

$$\omega_{opt} = 2 \left(\frac{1 - \sqrt{1 - \xi}}{\xi} \right) \quad \xi = \left(\frac{\cos(\pi/L_x) + \cos(\pi/L_y)}{2} \right)^2 \quad (2.25)$$

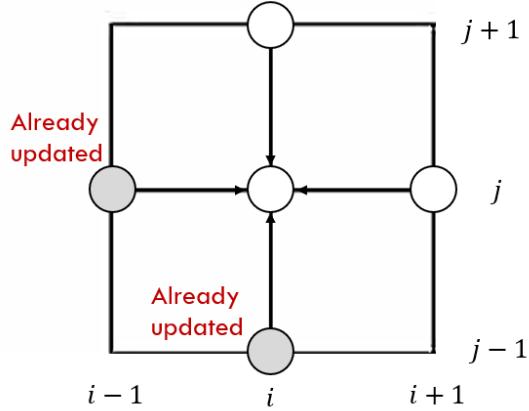


Figure 2.13: Successive-over-relaxation Iterative Method for Pressure Estimation across a 2D Computational Domain

The analysis performed by Pan et al. (2016) reveals the propagation of error within the solution of the Poisson equation specifically for PIV pressure reconstruction purposes. It is found that the greatest error after solution of the Poisson equations occurs for a square computational domain ($L_x = L_y$), and rectangular grids have lower error using Dirichlet boundary conditions, where the pressure along the boundary is explicitly set. Furthermore, error on the boundary dominates the pressure solution for small grids, whereas the error within the field dominates the solution for larger grids. This corresponds to the error associated with the source term. Finally, Dirichlet boundary conditions tend to yield less error than Neumann boundary conditions, which need to satisfy compatibility conditions, which is difficult to achieve.

The Poisson solution method of obtaining the pressure field, and thus reconstructing the pressure from PIV measurements is described by van Oudheusden (2013) as a global optimisation formulation of the direct integration method. Thus, it is found that this method averages over noise from the PIV measurements and is in fact invariant to noise in the data, making it particularly useful for general PPIV and SPIV methods and experimental assemblies.

Chapter 3

Objectives & Program Design

3.1 Research Objectives

This thesis is aimed to educate the reader on how to create a software program that quantitatively determines the planar pressure fields associated with planar velocity fields measured by PPIV and SPIV methods described in Section 2. Specifically, the objectives of this thesis are listed below:

- Design a pressure field reconstruction program that is generic to any type of flow, including a junction flow, like a flow around a hemispherical protuberance
- Design a program that is accessible and readable by engineers and interested individuals with knowledge and understanding of fluid mechanics
- Design a program with a breadth of user-defined settings for altering the behaviour of the numerical methods used to reconstruct the pressure field from a velocity field
- Verify that the pressure field reconstruction program performs to a standard observed in literature described in Section 2
- Compare and contrast the Eulerian and Lagrangian material acceleration estimation approaches detailed in Section 2
- Investigate the effect of user-defined settings on the error of the pressure field reconstruction
- Determine the optimal settings configuration for the lowest pressure field reconstruction error

These objectives will be achieved by writing the program in the MATLAB programming language due to the author's familiarity with the language, and it being ubiquitous at the University of Sydney. The author also has the intention of rewriting the program in Python with a Graphical User Interface (GUI) for wider accessibility by the general public. The program will comprise of multiple functions with a single script to execute them in sequence and reconstruct a pressure field from velocity field data from PIV measurements. The error of the pressure field reconstruction will be determined from a comparison to a pressure field generated from an analytical potential flow and from a more realistic CFD simulation. This error will be used as the metric for the effect of different user-defined settings on the program.

3.2 Program Design

This section follows the structure of the pressure field reconstruction program, and the content of the functions that together form the body of the program. Each function will have its purpose described, and its content explained with supporting pseudocode.

3.2.1 Program Overview and Architecture

The program is designed with the architecture presented in Figure 2.12.

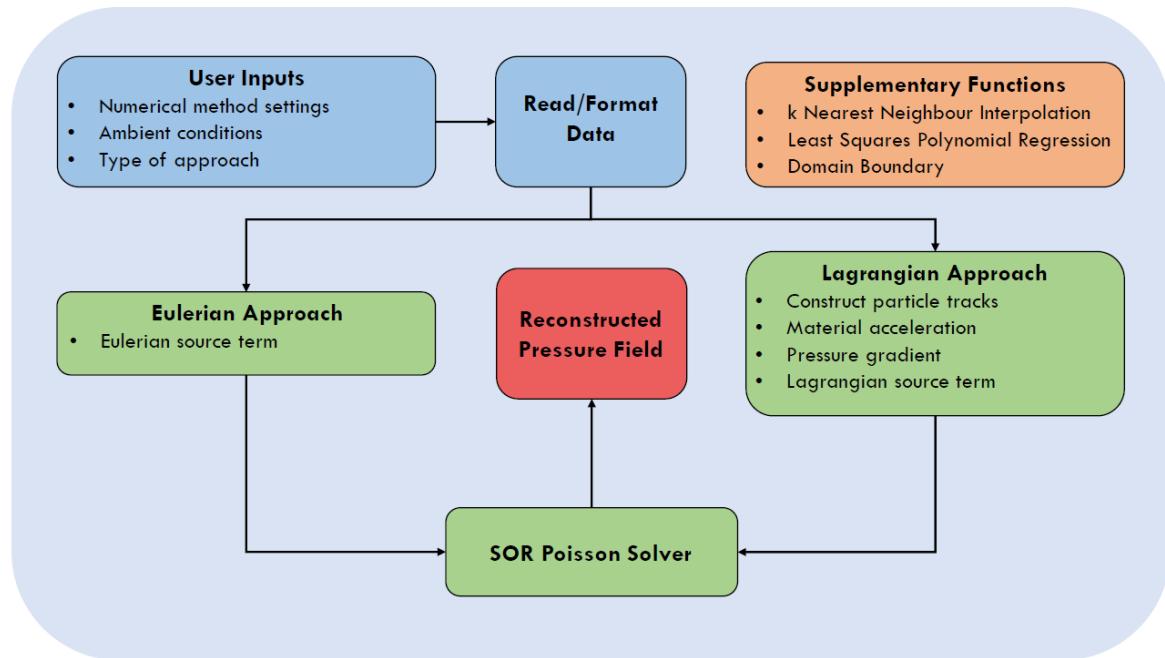


Figure 3.1: Program Architecture

Initially, the program needs inputs by the user. Specifically, the numerical method settings could include the number of velocity fields or time-steps to be investigated, the number of time-steps that a Lagrangian track will cover, the order of the particle track polynomial fit, the number of nearest neighbours to be considered for interpolation on the 2D computational domain and so on. The folder for reading the input velocity fields or data, which could be derived from PIV measurements, CFD simulations, or potential flow equations must be designated. Furthermore, the folder for saving the reconstructed pressure fields must chosen. Lastly, the ambient conditions associated with the to-be investigated velocity field, and the type of approach with which to reconstruct the pressure field (Eulerian or Lagrangian) must be entered by the user.

The user inputs will be fed to the rest of the program, which starts by reading and formatting the desired discrete velocity fields that are saved as text files or MATLAB data files. Upon formatting and reading the data into memory, the chosen approach will determine how these velocity fields are analysed. In the simpler Eulerian approach, the velocity fields across time-steps are used to calculate the temporal and spatial derivatives required for the source term described in Section 2. The more complex Lagrangian approach, requires a number of virtual particles to be propagated from the domain grid points across time-steps. A polynomial is then fit to every particle track and the material acceleration is extracted for the initial time-step or velocity field. Using the material acceleration and the initial velocity field, the viscous stresses are calculated and the pressure gradient vector is extracted. The source term for the Lagrangian approach is then the divergence of this pressure gradient vector.

Feeding the source term from either the Eulerian or Lagrangian approach into the next function allows the iterative Successive-over-relaxation method to solve the defined 2D, incompressible, and constant viscosity Poisson equation, with specified boundary conditions. The result is a non-dimensionalised pressure field on a planar grid, that can be saved or compared to pressure fields generated from potential flows or CFD simulation.

It is important to note that the planar velocity fields are read into the program containing at least five variables for every point in the field: horizontal coordinate, x , vertical coordinate, y , horizontal velocity, u , vertical velocity, v , and the measurement validity. With these five variables, the field can be considered 2D, allowing significant numerical method simplifications if the Eulerian approach is chosen. However, if the out-of-plane velocity, w is present, which is the case for SPIV measurements, the field is considered planar 3D, and

these simplifications cannot be made. The measurement validity determines the domain over which a pressure field reconstruction can occur within the observation plane of the original PIV measurement.

3.2.2 Non-dimensionalisation

To ensure generality with the results produced by this program, and minimise numerical error associated with round-off error, the governing variables of: pressure, displacement, velocity, time, density, and viscosity are non-dimensionalised, ensuring that their order of magnitude can be managed based on scale factors and constants. The scale factors are defined to ensure that the NS equations are non-dimensionalised as in Eq. 3.1, and they are described in Table 3.1.

Constant or Scale Factor	Symbol	Unit
Ambient Fluid Density	ρ	kg/m ³
Ambient Fluid Dynamic Viscosity	μ	Pa.s
Ambient Static Pressure	p_∞	Pa
Free-stream Fluid Total Velocity	U_∞	m/s
Scale length (Distance per camera pixel)	L_0	m
Scale Pressure	$p_0 = \rho U_\infty^2$	Pa
Scale Time	$t_0 = L_0/U_\infty$	s
Scale Reynolds Number	$Re_0 = \rho U_\infty L_0 / \mu$	

Table 3.1: Program Constants and Scale Factors used for Non-Dimensionalisation

$$\begin{aligned}
\nabla p &= -\rho \frac{D\mathbf{u}}{Dt} + \mu \nabla^2 \mathbf{u} \rightarrow \frac{\partial p}{\partial x} = -\rho \frac{Du}{Dt} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \\
\frac{\partial \hat{p}}{\partial \hat{x}} \frac{p_0}{L_0} &= -\rho \frac{D\hat{u}}{D\hat{t}} \frac{U_\infty}{t_0} + \mu \left(\frac{\partial^2 \hat{u}}{\partial \hat{x}^2} + \frac{\partial^2 \hat{v}}{\partial \hat{y}^2} \right) \frac{U_\infty}{L_0^2} \\
\frac{\partial \hat{p}}{\partial \hat{x}} &= -\frac{D\hat{u}}{D\hat{t}} + \frac{1}{Re_0} \left(\frac{\partial^2 \hat{u}}{\partial \hat{x}^2} + \frac{\partial^2 \hat{v}}{\partial \hat{y}^2} \right) \rightarrow \hat{\nabla} \hat{p} = -\frac{D\hat{u}}{D\hat{t}} + \frac{1}{Re_0} \hat{\nabla}^2 \hat{\mathbf{u}}
\end{aligned} \tag{3.1}$$

3.2.3 k Nearest Neighbour Interpolation

This function is used to interpolate a discrete 2D field onto a point with spatial coordinates within or close to the boundary of the field using the k Nearest Neighbour algorithm. Therefore, points between the grid points can be assigned values. The algorithm uses a certain number of points within the field (set by the user) that have the closest straight line (Euclidean) distance to the target point to calculate a distance weighted average for the target point value. According to Russell and Norvig (2021), the k Nearest Neighbour algorithm performs well in low-dimensional space (2D), unlike in higher dimensions, where the notion of "distance" breaks down. The pseudocode for the function is provided below.

```

1 % FUNCTION INPUT: Field Coordinates, Field Values, Target Point ...
    Coordinates, Number of Nearest Neighbours (k)
2
3 CALCULATE array of Euclidean distances between target point and all ...
    points of field
4 SORT distance array in ascending fashion
5 SET new array with k first elements of distance array
6 SET array of field values corresponding to the array indices of the ...
    minimum distances in the original distance array
7 SET small error variable to prevent infinities
8 CALCULATE distance weights using the reciprocal of the sum of the new ...
    distance array and the error
9 CALCULATE target point value with sum of the product of the weights ...
    and field values divided by the sum of the weights
10
11 % FUNCTION OUTPUT: Target Point Value

```

For clarity, the Euclidean distance is calculated by Eq. 3.2, with the field coordinates denoted by vectors: \mathbf{x}_f and \mathbf{y}_f , and the target coordinates by x , and y . The weights, α , and the target point value are calculated by Eq. 3.3 with the field values denoted by \mathbf{f} .

$$d = \sqrt{(\mathbf{x}_f - x)^2 + (\mathbf{y}_f - y)^2} \quad (3.2)$$

$$\alpha = \frac{1}{d_{min} + \epsilon} \quad g = \frac{\sum \alpha \mathbf{f}}{\sum \alpha} \quad (3.3)$$

3.2.4 Least Squares Polynomial Regression

This function is used to fit a polynomial function to discrete values of an independent variable, y which is correlated to a dependent variable x . Specifically, it is used to fit a polynomial of order, m to the x component of instantaneous particle velocity along its track with the dependent variable being time. The y component is also fit to a polynomial, both with the end goal of obtaining the material acceleration vector or the first derivative of the velocity vector. To achieve a polynomial fit, a least squares regression technique is used, following the method of van Gent et al. (2018).

A polynomial of order, m is described by Eq. 3.4. Given N x and y values, the $m + 1$ coefficients of the polynomial can be optimised with minimum error between the fit and the real values of y . Eq. 3.5 describes the vector of y values as a summation of matrices: \mathbf{X} is the x power matrix, \mathbf{a} is the coefficient vector, and ϵ is a vector of random error. Eq. 3.6 describes the estimated coefficient vector using matrix multiplication.

$$y(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m \quad (3.4)$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} 1 & x_1 & \cdot & t_1^m \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 1 & x_N & \cdot & x_N^m \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix} + \begin{bmatrix} \epsilon_0 \\ \cdot \\ \cdot \\ \epsilon_m \end{bmatrix} \rightarrow \mathbf{y} = \mathbf{X}\mathbf{a} + \boldsymbol{\epsilon} \quad (3.5)$$

$$\hat{\mathbf{a}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (3.6)$$

The pseudocode for this procedure is presented below, the R-squared value can be determined according to Eq. 3.7, with y_i representing the fit values, and \bar{y} is the mean.

$$SS_{res} = \sum (y - y_i)^2 \quad SS_{tot} = \sum (y - \bar{y})^2 \quad R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (3.7)$$

```

1 % FUNCTION INPUT: X Array, Y Array, Polynomial Order (m)
2
3 SET LENGTH(X Array) by m+1 matrix as X power matrix
4 SET first column of matrix as column vector of ones

```

```

5
6 FOR LENGTH(X Array)
7     FOR m
8         SET matrix current row and next column element as current X ...
9             value to the power of current column
10    ENDFOR
11
12   ENDFOR
13
14 CALCULATE estimated polynomial coefficients using matrix multiplication
15
16 % FUNCTION OUTPUT: Fit Y Array, Polynomial Coefficients, R-squared Value

```

3.2.5 Domain Boundary

This function takes the computational domain defined by the measurement validity variable, and returns a matrix of the same size as the domain defining the internal edge of the domain with logical True values, and everywhere else is False. This is required to initialise the boundary conditions for the SOR Poisson solver.

To achieve this, the validity domain is translated into 2D grayscale image, and binarised. A MATLAB specific function, **bwboundaries** is then used to extract the (x,y) coordinates of the internal boundary of every closed loop in the image (that being the closed loop of the domain). These coordinates are used to set the True values inside a matrix of the same size as the domain. Finally, the domain boundary and internal domain (without the boundary) are extracted. The pseudocode is provided below.

```

1 % FUNCTION INPUT: Measurement Validity Matrix
2
3 SET measurement validity matrix as grayscale image
4 EXTRACT domain boundary as coordinates of only closed loop in image
5 SET domain boundary matrix
6 FOR LENGTH(domain boundary coordinate vector)
7     SET domain boundary matrix element corresponding to coordinate as ...
        TRUE
8 ENDFOR
9 SET internal domain as measurement validity matrix minus internal ...
    domain boundary

```

```

10
11 % FUNCTION OUT: Domain Boundary Matrix, Internal Domain Matrix

```

3.2.6 Eulerian Source Term

As described in Section 2.3.2.2, the source term which consists of spatial (and temporal) velocity derivatives and therefore velocity information, is required for pressure field reconstruction using a Poisson solver. The source term is described by Eq. 2.21 for the 2D implementation, and Eq. 2.22 for the 3D implementation.

If the 2D implementation is chosen (for velocity field data that does not contain the out-of-plane velocity component), the source term calculation relies on one velocity field (at the current time-step), and therefore only spatial information. The resulting calculation of the first velocity derivatives uses the second order central difference scheme according to Eq. 2.12. These derivatives are only calculated within the internal domain, ignoring the boundary conditions which will be set by the Poisson solver.

If the 3D implementation is chosen, two velocity fields must be initialised, and the scheme of the first temporal derivative of the planar divergence is chosen based on whether a previous time-step is available (upwind scheme) or whether the next time-step is available (downwind scheme). The planar divergence is calculated using the first velocity derivatives calculated for the 2D implementation, saved as a matrix with values at every point in the domain, and then used to calculate the second order first and second spatial derivatives of the planar divergence, as well as the previously mentioned first temporal derivative. The resultant source term can then be fed to the Poisson solver for pressure field reconstruction.

The pseudocode of this function is presented below.

```

1 % FUNCTION INPUT: User Inputs, Velocity Field Coordinates, Velocity ...
   Field Values, Current Time-step, Measurement Validity Matrix
2
3 CALL DOMAIN_BOUNDARY function to obtain internal domain
4 SET domain column length
5 SET domain row length
6 SET current time-step x velocity
7 SET current time-step y velocity
8 INITIALISE first velocity derivative terms
9

```

```

10 FOR domain column length
11     FOR domain row length
12         IF matrix element is in internal domain THEN
13             CALCULATE first velocity derivatives using central ...
14                 difference scheme
15         ENDIF
16     ENDFOR
17
18 IF data is 2D THEN
19     CALCULATE source term
20 ELSE
21     CALCULATE planar divergence
22     INITIALISE next or previous time-step velocity field depending on ...
23                 available time-steps
24     INITIALISE derivatives of planar divergence
25     INITIALISE first derivatives of velocity for other time-step
26     FOR domain column length
27         FOR domain row length
28             IF matrix element is in internal domain THEN
29                 CALCULATE first derivatives of other time-step
30                 CALCULATE planar divergence derivatives using central ...
31                     difference scheme
32             ENDIF
33         ENDFOR
34     ENDFOR
35     CALCULATE planar divergence from other time-step
36     CALCULATE temporal derivative of planar divergence
37     CALCULATE source term
38 ENDIF
39 % FUNCTION OUTPUT: Eulerian Source Term

```

3.2.7 Lagrangian Approach

The following sections describe the functions that contribute to the calculation of the Lagrangian source term for input to the Poisson solver for pressure field reconstruction.

3.2.7.1 Particle Track Construction

The first function randomly fills the computational domain with a certain number of particles at its grid points. This ensures that any shaped domain can on average be evenly populated with virtual particles, without regions that are under resolved. The virtual particles are propagated forward and backward in time if possible given the availability of velocity fields. For example, if the user wants the particle track to span four time-steps or velocity fields, however there is only one velocity field available behind the current time-step: the particles will propagate from the current time, once backward in time and twice forward in time.

The particle track is constructed using the first order Euler integration method defined by Eq. 2.6, however for backwards construction, the time-step, Δt is negative. After a single step, k Nearest Neighbour interpolation is used to determine the particle's new x and y velocity components using its new position (x, y) in the field. If the particle leaves the boundaries of the domain then track construction stops. The function's pseudocode is described below.

```
1 % FUNCTION INPUT: User Inputs, Velocity Field Coordinates, Velocity ...
   Field Values, Current Time-step, Measurement Validity Matrix
2
3 INITIALISE velocity field
4 SET variable for matrix element indices of Measurement Validity ...
   Matrix that are TRUE
5 CALL function for random sample of matrix indices with number of ...
   samples equal to number of particles
6 SET variables for arrays containing x,y coordinates and x,y ...
   velocities of randomly sampled points (element indices) in domain
7 INITIALISE Particle Tracks data structure containing x,y coordinates ...
   and x,y velocities for every particle across desired time-steps
8
9 CALCULATE array of integer steps required to traverse desired number ...
   of time-steps from the current time given available time-steps ...
   backwards and forwards
10 SET variable for array of integer steps + current time (with ...
   reference to the total number of time-steps that will have their ...
   pressure field reconstructed)
11 SET time array by multiplying the new step array by the step size in ...
   seconds
```

```

12
13 FOR Number of Particles
14     INITIALISE current time and current particle coordinates and ...
15         velocities
16     INITIALISE temporary coordinate and velocity arrays for current ...
17         particle with current time values
18     FOR desired time-steps forward in time
19         CALCULATE next particle positions using Euler method
20         CALL k Nearest Neighbour to determine next particle ...
21             velocities from next velocity field
22         SET temporary coordinate and velocity arrays with values ...
23             corresponding to specific time-step
24     IF particle is not inside domain THEN
25         CALL break loop
26     ENDIF
27 ENDFOR
28 FOR desired time-steps backward in time
29     CALCULATE next particle positions using reverse Euler method
30     CALL k Nearest Neighbour to determine next particle ...
31         velocities from next (backwards) velocity field
32     SET temporary coordinate and velocity arrays with values ...
33         corresponding to specific time-step
34     IF particle is not inside domain THEN
35         CALL break loop
36     ENDIF
37 ENDFOR
38 APPEND x,y coordinates and x,y velocities of particle across ...
39         desired time-steps to Particle Tracks
40 ENDFOR
41
42 % FUNCTION OUTPUT: Particle Tracks

```

3.2.7.2 Material Acceleration

This function loops over every particle track, fits polynomials of a desired order to the x and y velocity versus time trends, and extracts the first derivatives of these curves to obtain the x and y material accelerations of the particle at the current time. Due to the nature of the time array used for polynomial regression, where the current time is set at $t = 0$, the first derivative of the curve is simply the a_1 coefficient in Eq. 3.8. The pseudocode for the function is provided below.

$$u(t) = a_0 + a_1 t + a_2 t^2 + \dots + a_m t^m$$

$$\frac{Du(t)}{Dt} = a_1 + 2a_2 t + \dots + m a_m t^{m-1} \rightarrow \frac{Du(t=0)}{Dt} = a_1 \quad (3.8)$$

```

1 % FUNCTION INPUT: User Inputs, Particle Tracks
2
3 INITIALISE time array
4 INITIALISE x,y material acceleration arrays for number of particles
5 FOR Number of Particles
6     CALL polynomial regression function for time array and x particle ...
         velocity
7     CALL polynomial regression function for time array and y particle ...
         velocity
8     SET x,y material acceleration for each particle at current time ...
         as second coefficient
9 ENDFOR
10
11 % FUNCTION OUTPUT: Material Acceleration Arrays

```

3.2.7.3 Source Term

Firstly the pressure gradient is computed according to the non-dimensional NS equations in Eq. 3.1. To achieve this, the material acceleration for a number of points (particles) smaller than the number of points in the valid domain is extrapolated onto all the domain using k Nearest Neighbour interpolation. Furthermore, the second velocity derivatives or viscous stresses are computed using the second order central difference scheme in Eq. 2.12 for the current time. Upon summation, the x and y components of the pressure gradient are determined. Finally, the source term is computed by the divergence of the pressure gradient through a second order central difference scheme. This is encapsulated in the pseudocode below.

```

1 % FUNCTION INPUT: User Inputs, Velocity Field Coordinates, Velocity ...
   Field Values, Current Time-step, Measurement Validity Matrix, ...
   Material Acceleration
2

```

```

3 INITIALISE domain row length
4 INITIALISE domain column length
5 CALL DOMAIN_BOUNDARY function to obtain internal domain
6
7 INITIALISE x,y velocity fields
8 INITIALISE x,y material acceleration field matrices
9 INITIALISE second velocity derivative matrices
10
11 FOR domain column length
12     FOR domain row length
13         IF point is in internal domain THEN
14             CALL k Nearest Neighbour to extrapolate x,y material ...
15                 acceleration to entire domain
16             CALCULATE second velocity derivatives using central ...
17                 difference scheme
18         ENDIF
19     ENDFOR
20 ENDFOR
21
22 CALCULATE x,y pressure gradient matrices by summation of material ...
23                 acceleration and second velocity derivatives
24
25 FOR domain column length
26     FOR domain row length
27         CALCULATE Source Term by divergence of pressure gradient ...
28             using central difference scheme
29     ENDFOR
30 ENDFOR
31
32 % FUNCTION OUTPUT: Source Term

```

3.2.8 SOR Poisson Solver

The SOR Poisson solver obtains a solution of the Poisson equation for the pressure field with a given source term, derived from the Eulerian or Lagrangian approaches. The solution at a specific time-step is a reconstructed pressure field from a velocity field generated from PIV measurements, CFD simulation or potential flow equations.

The solver begins by first setting an initial pressure field, and values determined by the user will drive the parameters associated with the numerical method. These are:

the maximum number of iterations, the minimum tolerated error between consecutive iterations, and the choice of over-relaxation parameter, which is set to the optimal value (Eq. 2.25) by default.

The pressure field is prepared using only Dirichlet boundary conditions determined by the non-dimensional steady Bernoulli equation in Eq. 3.9. Dirichlet boundary conditions are used as they are independent of the domain shape, whereas mixed boundary conditions would require the domain boundary to be split into different sections. This is important as the valid domains produced by PIV measurements are usually rectangular with random discontinuities due to the nature of the velocity calculations performed by the software used to process the PIV particle displacement measurements. Therefore, this method is user-friendly, domain independent, and has lower error for rectangular domains as described in Section 2.3.2.2 by Pan et al. (2016).

$$\hat{p}_{BC} = \frac{1}{2}(1 - (\hat{u}^2 + \hat{v}^2)) \quad (3.9)$$

Every point in the internal domain has its pressure updated according to the SOR scheme in Eq. 2.24, where the pressure solution rapidly converges due to concurrent updating described in Section 2.3.2.2. The convergence is checked based on a relative error metric, and the final pressure field is the gauge pressure relative to the ambient pressure, output as the established pressure coefficient defined by Eq. 3.10. The pseudocode for this solver is provided below.

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho U_\infty^2} = 2\Delta\hat{p} \quad (3.10)$$

```

1 % FUNCTION INPUT: User Inputs, Velocity Field Coordinates, Velocity ...
   Field Values, Current Time-step, Measurement Validity Matrix, ...
   Source Term
2
3 INITIALISE domain row length
4 INITIALISE domain column length
5 CALL DOMAIN_BOUNDARY function to obtain internal domain and domain ...
   boundary

```

```

6
7 INITIALISE Successive-over-relaxation parameters
8 INITIALISE gauge pressure field matrix
9 SET domain boundary x,y velocities
10 CALCULATE pressure boundary conditions using Bernoulli equation
11
12 FOR maximum iterations
13     SET previous pressure field
14     FOR domain column length
15         FOR domain row length
16             IF point is in internal domain THEN
17                 CALCULATE new pressure at point according to ...
18                 Successive-over-relaxation
19             ENDIF
20         ENDFOR
21     ENDFOR
22     CALCULATE maximum pressure field relative error
23     IF error is below minimum error THEN
24         CALL break loop
25     ENDIF
26 ENDFOR
27 CALCULATE pressure coefficient from pressure field
28
29 % FUNCTION OUTPUT: Pressure Field Matrix

```

3.2.9 User-defined Settings

This pressure field reconstruction program fulfils the objective of enabling high user control with numerical method settings. The general effects of the settings are detailed below, and the effects of these settings on the error of the pressure field reconstruction is explored in Section 4.

1. Number of Nearest Neighbours for Interpolation (k)

Affects the error associated with two-dimensional interpolation of a point in a known field, and it has the capacity to under- or over-fit data

2. Maximum Length of Lagrangian Particle Tracks (N_t)

Affects the error of the material acceleration dependent on the order of the polynomial fit described in Section 2.3.1.2

3. Number of Lagrangian Particles (N_p)

Directly influences the computational time of the program and the level of smoothing of the material acceleration across the domain

4. Order of Polynomial Fit to Lagrangian Particle Tracks (m)

Affects the error of the material acceleration according to Section 2.3.1.2

5. Maximum Number of SOR Iterations (N_{max})

Controls computational time associated with the SOR Poisson solver, it may act to stop the function if it is stuck in a local minima, and it may act to prevent convergence

6. Minimum Error for SOR Convergence (ϵ_{min})

Directly affects the pressure field error

7. SOR Over-Relaxation Parameter (ω)

Directly influences the computational time of the SOR Poisson solver, however if set outside of the range of $1 \leq \omega \leq 2$ it will contribute to numerical instability and prevent convergence

Chapter 4

Program Results & Verification

This section aims to establish the standard of accuracy that the program fulfils in the state designed in Section 3, and how this compares to literature. This will be achieved by tasking the program to reconstruct pressure fields from velocity fields derived from potential flow equations, and CFD simulations to a certain degree of accuracy. The accuracy will be defined by error metrics specific to the Eulerian and Lagrangian approaches from Eq. 2.4 and 2.7, and an error metric comparing the expected pressure fields to the reconstructed pressure fields. Furthermore, velocity fields from PIV experiments will be used to verify that the program implementation follows expected trends for real data. However, due to the lack of access to acoustic measurements of a specific PIV experiment, empirical pressure field data will not be used to validate the program implementation. To meet the objectives described in Section 3, the Eulerian and Lagrangian approaches will be compared using the error metrics previously described, and the program settings for least error will be analysed.

Potential flow and CFD simulations can be used as verification tools for this program as they are both solutions for the dynamics of a continuum fluid, which is governed by the continuity and Navier-Stokes equations. Therefore, reconstructing pressure fields from their respective velocity fields to a close degree of accuracy demonstrates that the program is solving an approximate form of the NS equations.

4.1 Potential Flow

John D. Anderson (2017) describes potential flows as solutions for the dynamics of a fluid that is incompressible, and irrotational. These two conditions are represented by Eq. 4.1

for a fluid with a continuous velocity field, \mathbf{u} . The vector identity in Eq. 4.2 for a scalar velocity potential field, ϕ implies that the velocity field can be represented by the gradient of this field. Finally, the divergence of this field yields the 2D Laplace equation in Eq. 4.3, which all potential flows are a solution of. The idealised solution to the NS equations for this type of flow is the Bernoulli equation in Eq. 2.1 or Eq. 3.9.

$$\nabla \cdot \mathbf{u} = 0 \quad \nabla \times \mathbf{u} = 0 \quad (4.1)$$

$$\nabla \times (\nabla \phi) = 0 \Rightarrow \mathbf{u} = \nabla \phi \quad (4.2)$$

$$\nabla \cdot (\nabla \phi) = \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (4.3)$$

By a similar argument in John D. Anderson (2017), the velocity streamline function, ψ (orthogonal to the velocity potential) is also a solution of the Laplace equation.

4.1.1 Impinging Jet Case

A 2D jet flow impinging on a flat surface is studied in Gurka et al. (1999) as a means to compare the reconstructed pressure field to the empirically measured surface stagnation pressure. Here, it will be used to compare the known pressure coefficient field of the 2D flow to the pressure coefficient field reconstructed from the known velocity field. This type of potential flow is useful as it represents a truly rectangular domain compatible with a Cartesian coordinate system without an object inside the domain like a cylinder or airfoil.

Gurka et al. (1999) reveals that the 2D impinging jet potential flow is governed by the velocity streamline function, and velocity fields in Eq. 4.4. C is chosen to ensure that the total velocity at the top of the domain at the centre-line is the free-stream velocity, U_∞ .

$$\psi = -Cxy \quad u = Cx \quad v = -Cy \quad C = \frac{U_\infty}{L_y} \quad (4.4)$$

The MATLAB programming language is used to perform the 2D Eulerian pressure field reconstruction from the analytical velocity fields described in Section 3, and this is compared to the analytical pressure field using the error metric in Eq. 4.5. The non-dimensional total velocity field is plotted in Figure 4.1a, the reconstructed pressure coefficient field is

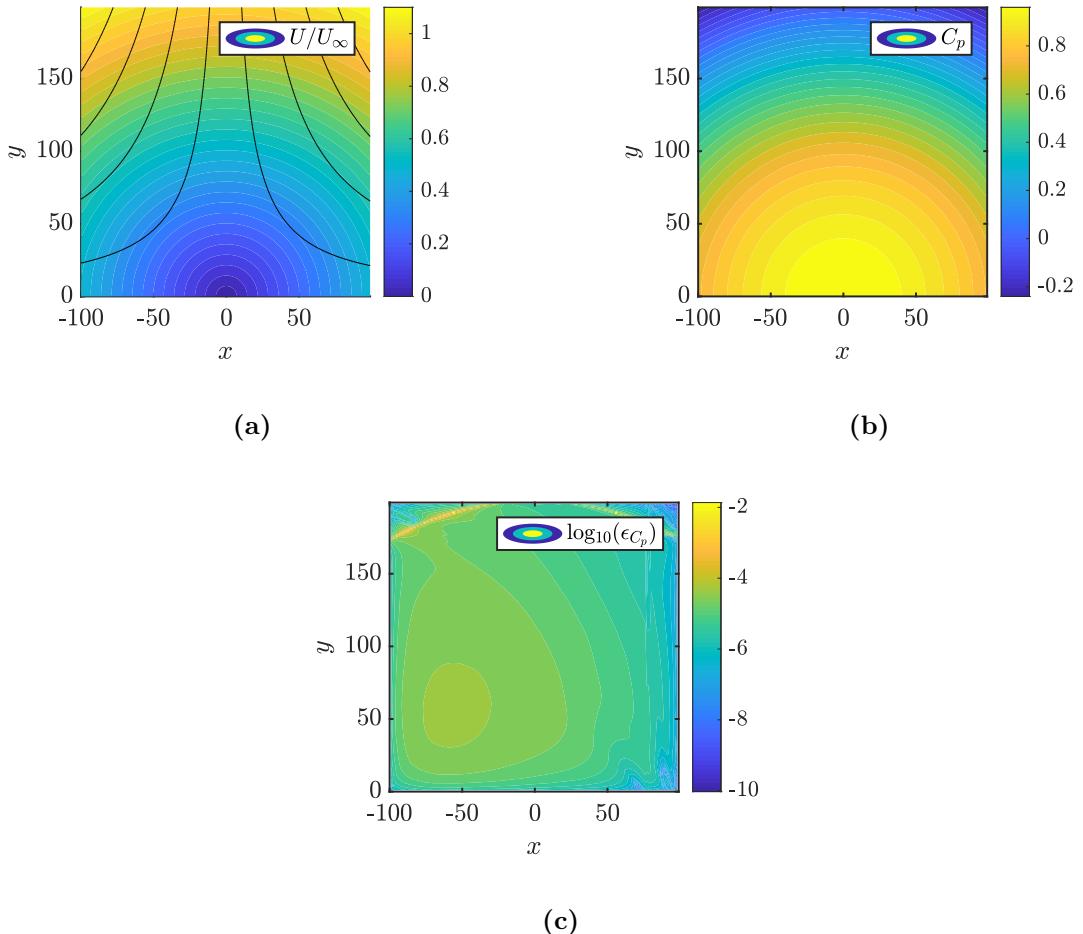


Figure 4.1: (a) Total Velocity Contour and Streamlines of Impinging Jet Potential Flow (b) Reconstructed Pressure Coefficient Contour (c) Pressure Coefficient Error Contour

Figure 4.1b, and the pressure coefficient error field is Figure 4.1c. The error field values are plotted in logarithmic scale (base 10) to enable the reader to observe the range of error magnitudes across the field. The domain size and program settings for this solution are presented in Table 4.1.

$$\epsilon_{C_p} = \left| \frac{C_{p,recon} - C_{p,anal}}{C_{p,anal}} \right| \quad (4.5)$$

Figure 4.1a shows the impinging jet flow decelerating as it approaches the wall surface or bottom edge of the domain, and expanding to follow its flat geometry. The reconstructed pressure coefficient field in Figure 4.1b reproduces the expected high pressure region of the stagnation point, and the radial distribution of the velocity field from the stagnation point. The behaviour of the analytical pressure coefficient field is similar, as it is purely

dependent on the total velocity field.

Figure 4.1c reveals a complex and discontinuous pressure coefficient error field. The SOR Poisson solver effectively updates the field from the bottom left vertex to the top right vertex of the domain every iteration. This procedure leaves a directional artifact in the error field, where significant error region is observed near the bottom left vertex. This region will have less new information than the top right region (upon fulfilling the convergence criterion), which has exponentially more new information about the pressure field due to the concurrently updating nature of the SOR iterative method. The lowest error in deep purple is observed at the boundaries of the domain, as the Dirichlet boundary conditions (initialised with the Bernoulli equation) will be approximately equivalent to the analytical pressure field with the exception of numerical round-off error. The greatest error in yellow is observed at the top of the domain following the edge of a pressure contour. This thin error region is the boundary where the analytical pressure field approaches zero, and the error in Eq. 4.5 approaches large values. In reality, the pressure coefficient error on either side of this thin boundary is approximately $10^{-4.5}$, therefore the boundary can be ignored as a small region of inaccuracy.

Setting	Symbol	Value
Horizontal Domain Length	L_x	200
Vertical Domain Length	L_y	200
Maximum Number of SOR Iterations	N_{max}	10^4
Minimum Error for SOR Convergence	ϵ_{min}	10^{-6}
SOR Over-Relaxation Parameter	$\omega = \omega_{opt}$	1.97

Table 4.1: Program Settings for 2D Eulerian Pressure Field Reconstruction of an Impinging Jet Potential Flow

4.1.2 Hemispherical Protuberance Case

A 2D potential flow around a hemispherical protuberance is an additional program verification case, specifically chosen as a direct comparison to the CFD simulation also concerning flow around a hemispherical protuberance. This flow contains an obstructing solid circular element that transforms the domain into a semi-rectangular domain.

John D. Anderson (2017) describes the potential flow over a cylinder as the linear combination of a uniform flow and a doublet flow. When this potential flow is shifted to the

bottom edge of the domain it is analogous to the 2D flow over a hemispherical protuberance or wall mounted hemisphere. This flow is governed by the streamline function and velocity fields in Eq. 4.7, with the doublet strength determined by the desired hemispherical radius, r (one eighth of the horizontal length of the domain) in Eq. 4.6.

$$r = \frac{L_x}{8} \quad M = 2\pi U_\infty r^2 \quad (4.6)$$

$$\psi = U_\infty y - \frac{My}{2\pi(x^2 + y^2)} \quad u = U_\infty - \frac{M(x^2 - y^2)}{2\pi(x^2 + y^2)^2} \quad v = -\frac{Mxy}{\pi(x^2 + y^2)^2} \quad (4.7)$$

The total velocity contour, its streamlines, the reconstructed pressure coefficient field, and the pressure coefficient error field are shown in Figure 4.2, and generated using the same program settings and 2D Eulerian approach for the impinging jet potential flow in Table 4.1.

Figure 4.2a shows the characteristic flow over a hemispherical protuberance, with a front stagnation point, flow accelerating over the front half of the hemisphere, and the upper free-stream being relatively uniform and unaffected by the hemisphere. The reconstructed pressure coefficient field in Figure 4.2b, like Figure 4.1b for the impinging jet potential flow shows a distribution similar to the related velocity field, due to the expected analytical dependency of the pressure coefficient on velocity. This corresponds to a high pressure front stagnation point, and low pressure upper hemisphere surface. However, the sparsest pressure contour and the lower pressure contours do not match the corresponding velocity field contour lines, which will contribute to the pressure coefficient error.

Figure 4.2c reveals an approximately symmetric pressure coefficient error field unlike the impinging jet error field in Figure 4.1c. However, the high error boundaries are present here, and do continue to correspond to the analytical pressure coefficient approaching zero. Furthermore, these thin boundaries contain on average an error of 10^2 that cannot be neglected, and are located between similarly high error contours. These boundaries that extend from the hemisphere as the flow is accelerated past the free-stream velocity, must be considered when analysing the CFD simulation pressure reconstruction.

4.1.3 Summary

The impinging jet potential flow reveals that the program is well suited for pressure field reconstruction of flows within unobstructed rectangular domains, with an average pressure

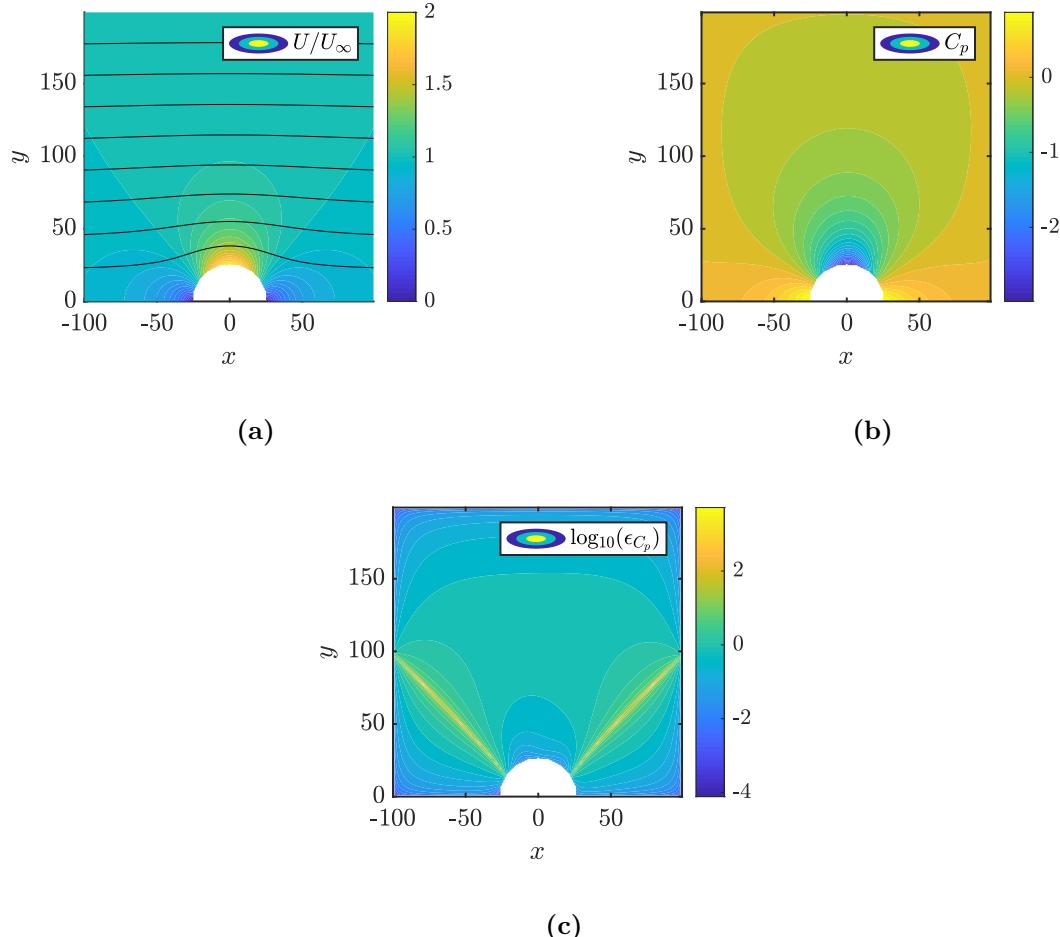


Figure 4.2: (a) Total Velocity Contour and Streamlines of Hemispherical Protuberance Potential Flow (b) Reconstructed Pressure Coefficient Contour (c) Pressure Coefficient Error Contour

coefficient error of $10^{-4.9}$, a minimum error of 0 (at the boundaries), and a maximum error of $10^{-1.6}$, all within 2.5% of the "true" analytical pressure field solution.

The hemispherical protuberance potential flow demonstrates the program's difficulty with handling flows within obstructed semi-rectangular domains with an average pressure coefficient error of $10^{-0.27}$ or 53.7 %, same minimum error at the boundaries, and a significant maximum error of $10^{4.0}$. This could be due to the Cartesian coordinate system having difficulty with inherently curved/polar flows that could benefit from a cylindrical formulation as postured by Gurka et al. (1999).

Therefore, on average the program can effectively solve the NS equations and reconstruct the pressure field of analytical potential flows using the 2D Eulerian approach, however it is more accurate for flows within unobstructed rectangular domains.

4.2 CFD Simulation

As previously explored in Section 2.2, hemispherical protuberances are important to the Australian defence industry as fundamental aerodynamic bodies that represent functional structures, such as optical turrets. Studies of these bodies are integral to understanding the performance, and optimising the design of modern and future UAVs. To align with these studies and the objectives of the Australian Defence Science and Technology Group (DSTG), this thesis aims to verify the pressure field reconstruction program detailed in Section 3 for PIV measurement of the pressure field surrounding hemispherical protuberances in subsonic conditions.

To achieve this, a CFD study performed by Dr. Nicholas Giannelis from the University of Newcastle is used as a verification case of the planar 3D Eulerian and Lagrangian approaches for pressure field reconstruction. Specifically, computational results for the three-component velocity and pressure fields have been generated for flow around a 100 mm diameter wall-mounted hemisphere at a Reynolds number of $Re \approx 64000$. The results explored in this thesis concern the centre-plane of the computational domain at $z = 0m$, corresponding to the centre-line of the hemisphere. This study is from a publication not yet available for reference, therefore a description of the simulation is provided below.

The simulation is performed with the commercial finite volume CFD code ANSYS Fluent 2021R1 detailed in ANSYS (2020). The incompressible, three-dimensional, implicit, pressure-based solver is employed, where the coupled set of momentum and pressure-based continuity equations are solved simultaneously. Second-order upwind differencing is used for pressure discretisation and a low-dissipation bounded central-differencing employed for the convective flow variables. The diffusive fluxes are treated with a second-order accurate central-difference scheme. Gradients for the convective and diffusive terms are computed at cell faces through a cell-based Least Squares reconstruction scheme and solved by Gram-Schmidt decomposition of the cell coefficient matrix.

The addition of equations that govern turbulent flow properties or turbulent closure of the Navier-Stokes equations, is achieved using Algebraic Wall-Modelled Large Eddy Simulation (WMLES) Shur et al. (2008); a hybrid Reynolds Averaged Navier-Stokes (RANS)/LES modelling approach that blends Prandtl's algebraic mixing length model with a modified Smagorinsky subgrid scale (SGS) model Smagorinsky (1963) and Piomelli's wall damping function Piomelli et al. (1988). In this formulation, the algebraic

RANS model is limited to the inner region of the logarithmic layer, with the outer region resolved through the modified Smagorinsky, permitting relaxation of the strict grid requirements necessitated by near-wall resolution in wall-resolved LES.

The transient calculations are performed using an implicit, second-order accurate backward Euler dual time-stepping scheme with a physical time-step of 5×10^{-5} s. This time-step was selected as the original experiments by Franck Bauer (2019) indicate frequency content for hemispherical protuberance flow oscillations lies primarily below 100 Hz. The selected temporal discretisation permits at least 200 physical time-steps for the highest frequency components. Temporal convergence within a physical time-step is achieved when the relative change in the aerodynamic coefficients of lift, drag and side-force between successive pseudo-time iterations falls below 10^{-6} .

A rectangular computational domain has been employed in the simulations; extending 4 diameters upstream, 10 diameters downstream and 4 diameters in both span-wise and normal directions from the hemisphere. A medium density hexa-hedral grid of approximately 12 million cells was developed in Pointwise R18.4. Isotropic near-field spacing of approximately 0.5 mm cubes is resolved within 1 diameter from the hemisphere, which are diffused to the far-field at a growth rate of 1.05. A subsonic velocity inlet with standard sea level conditions is employed upstream, along with an experimentally derived inlet boundary layer profile of magnitude 9.9 m/s in the free-stream. A subsonic pressure outlet diffuses the flow to atmosphere downstream of the hemisphere. Viscous walls are employed on the wind tunnel floor and hemisphere surfaces, with no-slip walls used on the remaining far-field domains.

The results are saved as files containing the 3D coordinates, three-components of velocity, pressure, and total pressure of every node in the computational domain, for a specific time-step in the simulation. A discrete time-step of 0.01 s is chosen to correspond to the 100 Hz frequency content limit of the hemispherical protuberance flow oscillations, and reduce the number of time-steps analysed in the program. Therefore, only computational results at multiples of 0.01 s are sampled.

To read the CFD data, a new computational domain inside MATLAB is generated with the same number of points and aspect ratio as the simulated domain. This is filled with velocity and pressure field values at every coordinate using k Nearest Neighbour interpolation ($k = 4$). This method was chosen due to the lack of organisation in the data files, and the presence of duplicate floating-point coordinates, making interpolation

the simplest approach. The area that would contain the solid hemisphere slice in the simulated plane is set to have False measurement validity values, to ignore this region during pressure field reconstruction.

The pressure fields reconstructed from the CFD velocity fields will be verified using a specific material acceleration error metric for the type of reconstruction approach defined by Eq. 2.4 or 2.7, and the general pressure coefficient error metric in Eq. 4.5. Figure 4.3 presents the three distinct regions in the CFD velocity fields that will be analysed. These regions are generally differentiated by the types of (or lack of) turbulent structures in the flow around a hemispherical protuberance as explained in Section 2.2. The average node distance of 0.001 m is used as the scale length, L_0 for the system.

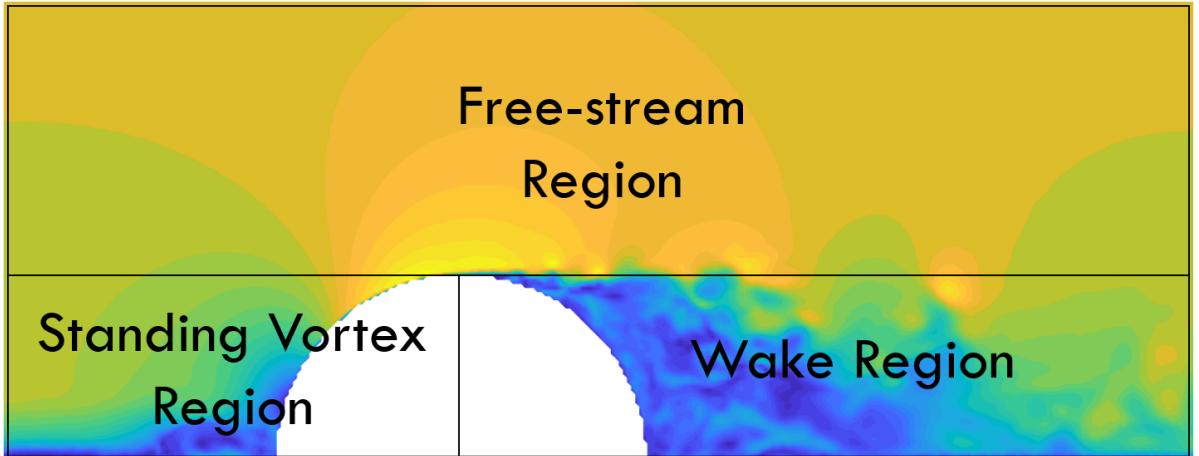


Figure 4.3: Distinct Regions of Flow around a Hemispherical Protuberance

4.2.1 Material Acceleration Error Comparison

The error associated with the material acceleration estimation approach of Eulerian or Lagrangian is compared for the described WMLES simulation of a flow around a hemispherical protuberance. This metric acts as a substitute for the pressure gradient error, as the calculation of viscous stresses is approximately equivalent between the two approaches. The error will be calculated according to Eq. 2.4 and 2.7, and the error associated with the velocity measurement uncertainty can be ignored as it acts linearly on the error as a scalar field and is equivalent between the two methods. The velocity field is plotted in Figure 4.4a, and the error associated with the Eulerian and Lagrangian approaches are plotted in Figures 4.4b and 4.4c.

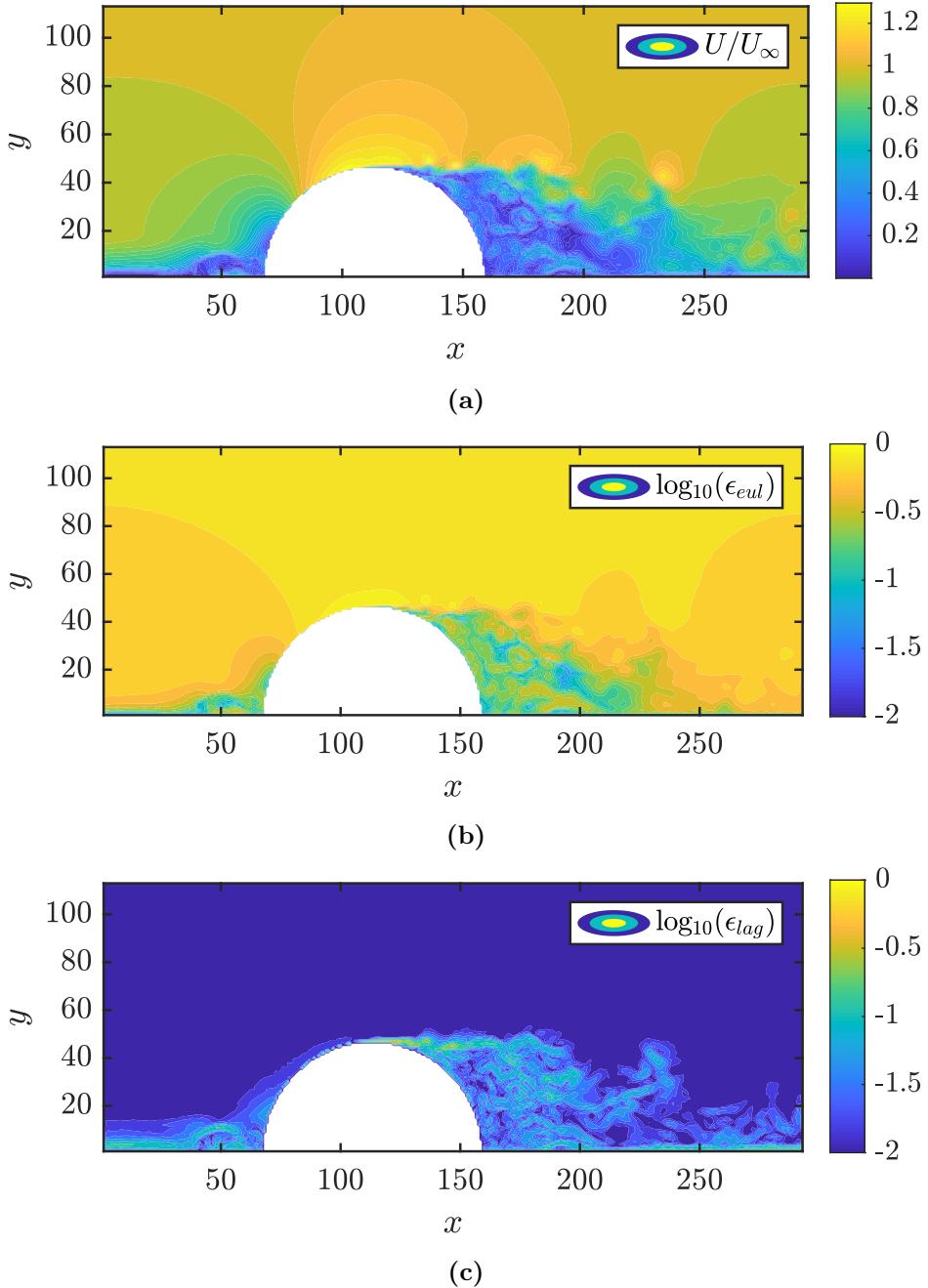


Figure 4.4: (a) WMLES Total Velocity Contour for Flow around a Hemispherical Protuberance at its Centre-plane (b) Eulerian Approach Material Acceleration Error (c) Lagrangian Approach Material Acceleration Error

Figure 4.4a shows the characteristic turbulent structures of the hemispherical protuberance with the standing vortex close to the front stagnation point, the acceleration of the attached flow around the front edge of the hemisphere leading to flow separation close to the top edge, and then a fully turbulent wake forming behind the hemisphere. This field is similar to the mean velocity field measured by Franck Bauer (2019) for a hemispherical protuberance at $Re = 69600$.

The material acceleration error generated through the Eulerian approach is sensitive to advection dominated phenomena as discussed in Section 2.3.1.2. Therefore, for this case where the free-stream velocity is 9.9 m/s, the advection term dominates the error in Figure 4.4b leading to a high error in the free-stream. This is in contrast to Figure 4.4c where the dominant error is the total velocity gradient term at the high shear separation point, standing vortex, and turbulent wake. It is clear that for this advection dominated case, the Eulerian approach is less suitable with more error on average than the Lagrangian approach, however it will become clear that the Eulerian approach is more computationally efficient.

4.2.2 Pressure Coefficient Error Comparison

The Eulerian and Lagrangian approaches are compared using the pressure coefficient error metric in Eq. 4.5, except the analytical pressure coefficient field is replaced with the pressure coefficient field from the WMLES simulation. The pressure coefficient field will be reconstructed from the velocity field in Figure 4.4a. To ensure a fair comparison of the two approaches, the Lagrangian approach will use two time-steps (maximum length of particle track) to estimate the material acceleration in agreement with the Eulerian's two time-step first order temporal derivative scheme. The number of valid domain points where particles could be placed is 29906, therefore 7500 particles are chosen to randomly populate the domain in the Lagrangian approach to ensure that every particle has at least four field points within one spatial step that it can use for k Nearest Neighbour interpolation with $k = 4$. Finally, the Lagrangian approach uses linear regression for estimating the material acceleration, as it the closest to the performance of the upwind or downwind scheme that the Eulerian approach will use. These settings, and the SOR Poisson solver settings that the approaches will share are presented in Table 4.2.

The Eulerian approach reconstructs the expected high pressure front stagnation point with $C_p \approx 1$, and the low pressure attached flow over the front portion of the hemisphere, $C_p < 1$ in Figure 4.5a. However, it creates a localised region of very low pressure, $C_p \ll 1$ behind the hemisphere within its wake. This "bubble" does not realistically correspond to the flow structures observed in Figure 4.4a, nor what would be expected for a hemispherical protuberance, as the lowest pressure should occur in the attached accelerated flow region at close to 75° from the horizontal according to Taniguchi et al. (1982). Furthermore, the pressure contours reveal a non-uniform free-stream, which is likely incorrect as Figure 4.4a shows a free-stream region with relatively consistent velocity contours. The

pressure contours downstream and close to the wall show high values, which is likely correct, as the flow inside the wall boundary layer is slower than the free-stream and higher pressure.

Setting	Symbol	Value
Number of Nearest Neighbours	k	4
Maximum Length of Particle Track	N_t	2
Number of Particles	N_p	7500
Order of Polynomial Fit	m	1
Maximum Number of SOR Iterations	N_{max}	10^4
Minimum Error for SOR Convergence	ϵ_{min}	10^{-6}
SOR Over-Relaxation Parameter	$\omega = \omega_{opt}$	1.96

Table 4.2: Program Settings for Lagrangian Approach and SOR Poisson Solver

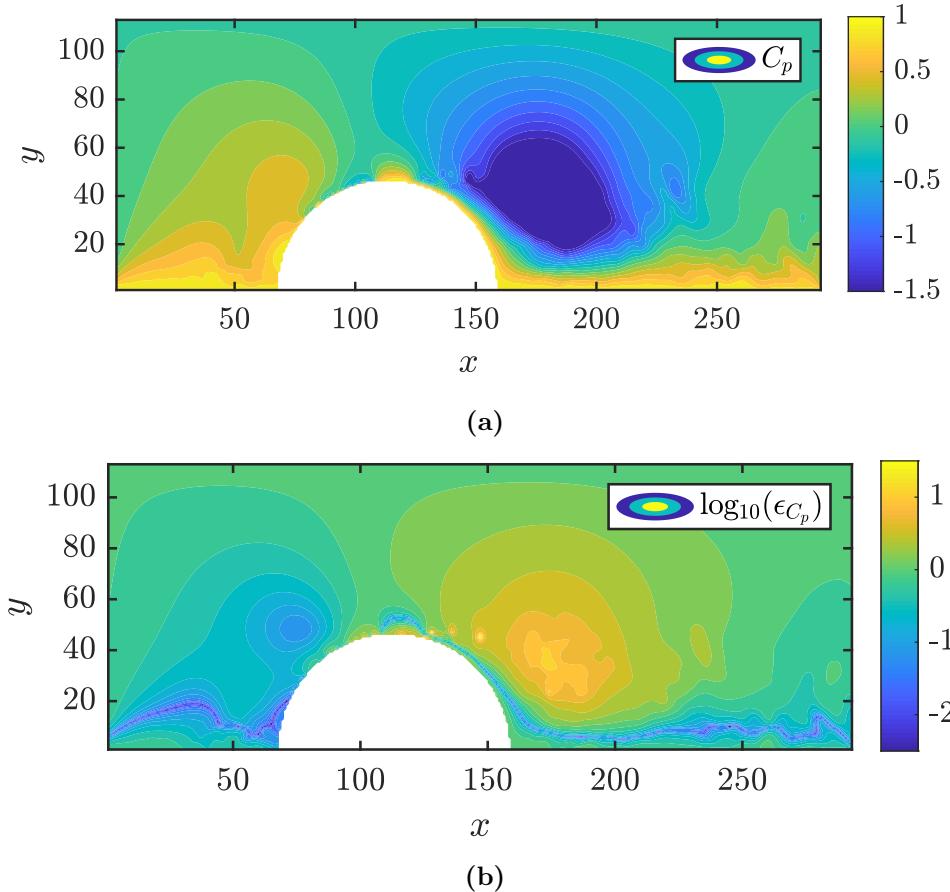


Figure 4.5: (a) Reconstructed Pressure Coefficient Field from Eulerian Approach (b) Pressure Coefficient Error Field from Eulerian Approach

The error contours in Figure 4.5b confirm that the low pressure "bubble" is in fact an artifact of the Eulerian approach pressure reconstruction, with pressure coefficient errors approaching $10^{1.2}$. This high error region is bounded by regions of positive pressure coefficient, similar to the effect found in the potential flow error fields, where the high error boundaries occur at the change of sign for the pressure coefficient. Across the top portion of the free-stream region, the error is also high at around 1 or 100%, indicating that the free-stream is not well resolved, likely due to the high material-acceleration error observed in the free-stream region. Fortunately, the standing vortex region is resolved well with generally low error indicated by blue. Furthermore, the error boundaries found in the potential flow error fields are observed again, however this time they reveal regions of low error. For this Eulerian approach, the lowest error is found near the domain boundaries corresponding to this thin error boundary which traverses the standing vortex, and wraps underneath the turbulent wake region.

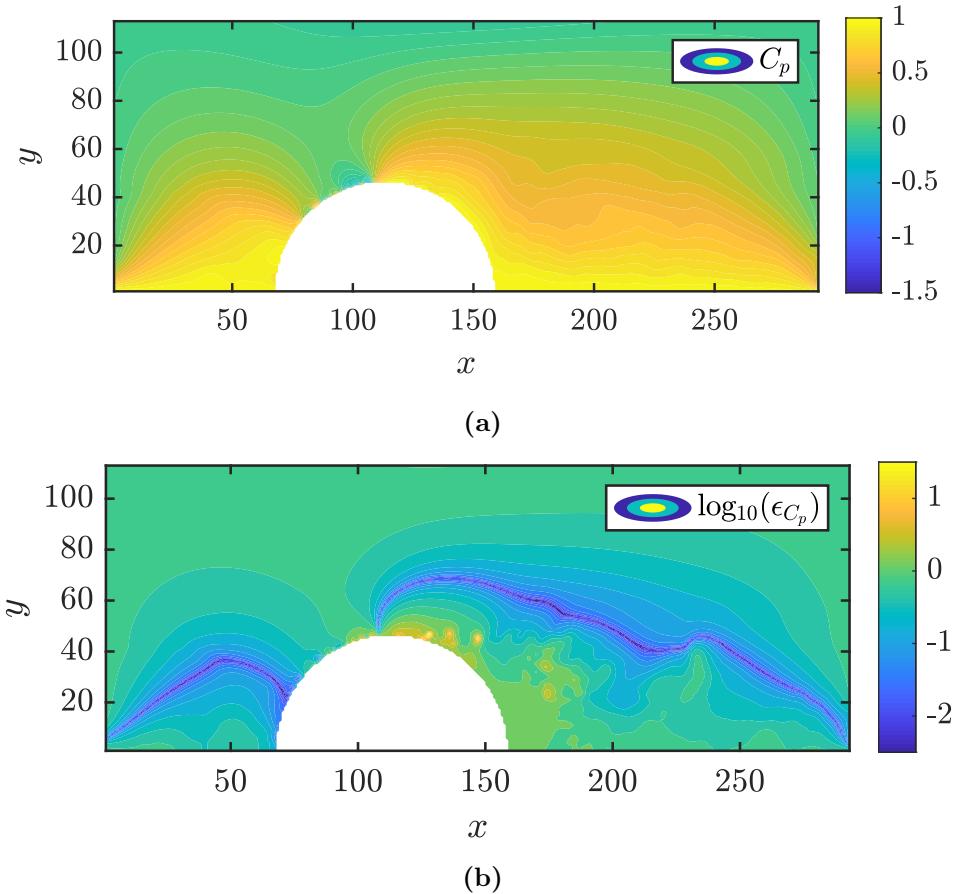


Figure 4.6: (a) Reconstructed Pressure Coefficient Field from Lagrangian Approach (b) Pressure Coefficient Error Field from Lagrangian Approach

The Lagrangian approach to pressure field reconstruction demonstrates a less erratic pressure field in Figure 4.6a compared to Figure 4.5a. The front stagnation point, attach

flow, turbulent wake, and free-stream can all be discerned with comparison to the velocity field in Figure 4.4a. The pressure coefficient contours follow the general trends of a high pressure rolled-up standing vortex, and a lower pressure turbulent wake that would result in significant pressure drag expected for the hemisphere acting as a bluff body. The free-stream region now looks more uniform and directional than the Eulerian approach in Figure 4.5a.

The results now satisfy the trends of Taniguchi et al. (1982), however the pressure coefficient error field in Figure 4.6b reveals high error of ≈ 10 along the back of the hemisphere, and this aligns to the negative pressure coefficients that should be observed according to Taniguchi et al. (1982). Generally, it is observed that the pressure coefficient error is lower across the domain than for the Eulerian approach in Figure 4.5b, expected as the material acceleration which dominates the pressure gradient calculation has lower error (Figure 4.4c). Surprisingly, the blue error boundaries now follow the outer contours of the standing vortex and turbulent wake in contrast to the boundaries in the Eulerian approach.

The mean pressure coefficient errors are compared between the two approaches across the distinct regions in Table 4.3. From this comparison, it is clear that both methods perform well in the standing vortex region, where the total velocity is low, and there is minimal complexity. However, the free-stream, and wake regions significantly differentiate the two approaches, where the Lagrangian approach more accurately models both regions, and the Eulerian approach suffers due to high sensitivity to turbulent structures, and noise as supported by van Gent et al. (2017), with the Lagrangian approach being more robust to noise with its linear regression approach to material acceleration estimation. However, with the current pressure coefficient error estimates, the Lagrangian approach is still inaccurate compared to de Kat and van Oudheusden (2012) with 5-10% mean pressure error. To rectify this issue, the effect of program settings will be investigated in the next section.

Region	Eulerian (%)	Lagrangian (%)
Standing Vortex (SVR)	37	35
Free-stream (FSR)	147	70
Wake (WR)	229	68

Table 4.3: Mean Pressure Coefficient Error across Regions of a Flow around a Hemispherical Protuberance for the Eulerian and Lagrangian Approaches

4.3 Effect of Program Settings

The effect of the program settings detailed in Section 3 on pressure coefficient error will be investigated using the more accurate Lagrangian approach. The program settings for the Lagrangian approach in Table 4.2 will be used as the default settings except with 1000 particles to ensure faster analysis. Only one setting will be varied at a time, and the mean pressure coefficient error across all three distinct regions for the hemispherical protuberance WMLES velocity field presented in Figure 4.4a will be presented. In the interest of the user, computational time will also be presented.

The effect of the number of particles is presented in Figure 4.7. Surprisingly, the number of particles used for the Lagrangian approach to pressure field reconstruction does not have a significant impact on the pressure coefficient error for any of the regions for the hemispherical protuberance case. It is expected that the greater number of particles should produce a more accurate representation of the velocity field with less smoothing, and therefore reduce the error. It is likely, that the consistent Dirichlet boundary conditions are having a significant impact on the reconstructed pressure field and affecting the mean error values in each region, therefore leaving the mean errors unchanged. However, the number of particles does linearly correlate with the computational time in Figure 4.7b, as the program must iterate over every particle to perform the material acceleration estimation. Therefore, every particle represents a finite amount of time that must be spent to obtain the material acceleration field. From these two figures it is clear, that it is desirable to use less particles for pressure field reconstruction within the range analysed, therefore the baseline of 1000 particles will be set as the default.

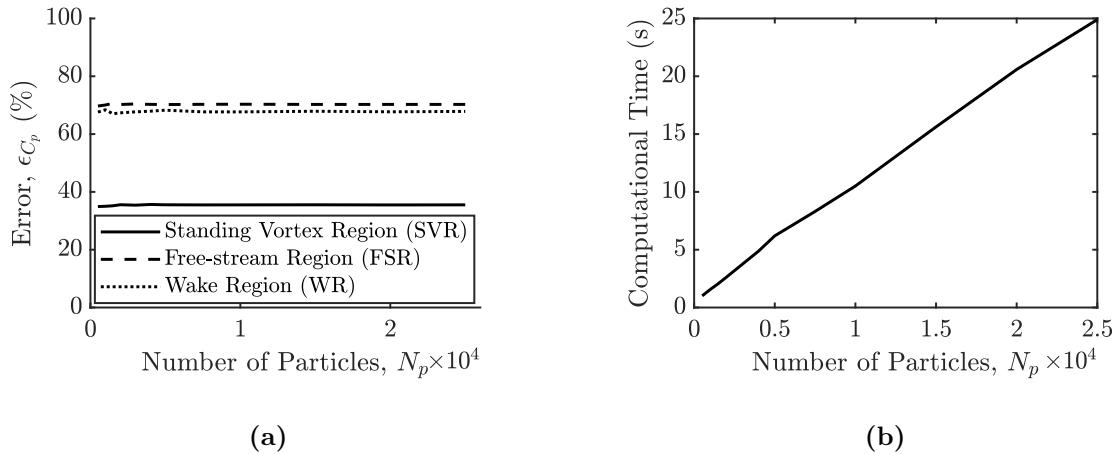


Figure 4.7: (a) Effect of Number of Particles on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Number of Particles on Computational Time

The effect of the maximum particle track length or the number of velocity fields that the particle can propagate over is considered in Figure 4.8a. SVR and FSR are barely affected by the change in track length, however WR increases in mean error with increasing track length, as long track could lead to excessive smoothing as postured by van Gent et al. (2018), which may be important for the highly fluctuating wake region. The lack of significant change in mean error is unusual, as it is expected that increasing track length leads to more temporal information in the pressure field, and thus a more accurate representation of the true pressure field as described in Section 2.3.1.2. Increasing track length does however have a non-linear correlation to computational time, as greater track lengths mean more particle coordinates and velocities must be analysed for material acceleration estimation. A maximum track length of 5 is chosen to be default as it has minimum error change, and it is likely to be more accurate for other flow cases.

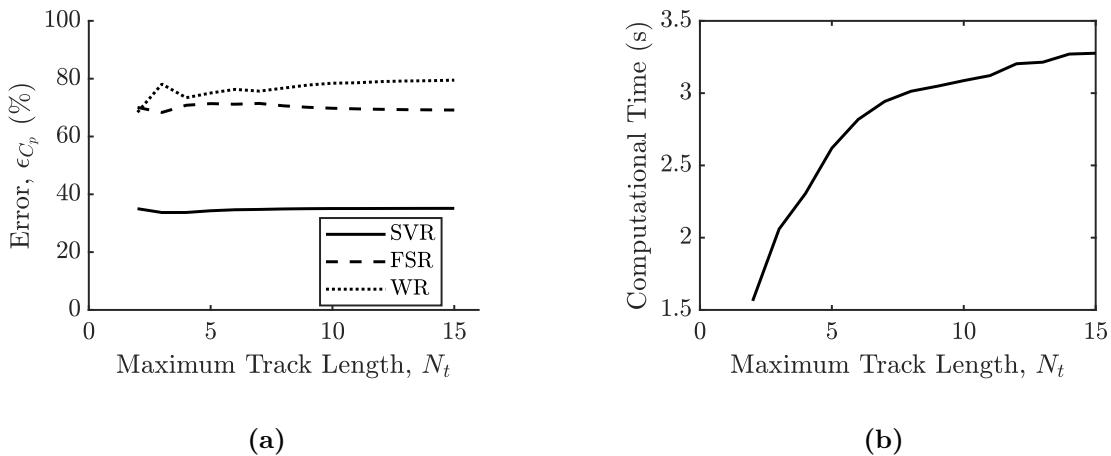


Figure 4.8: (a) Effect of Maximum Particle Track Length on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Maximum Particle Track Length on Computational Time

The order of polynomial fit is expected to increase the error for a baseline track length of 2, and Figure 4.9a does support this assertion from Section 2.3.1.2. Here the order was increased alongside the track length, as a proper fit to order m can only be made for $m+1$ points. Again, WR is subject to increasing error, as the particle tracks are being over-fit, where the highly fluctuating region requires more general material acceleration estimations. However, SVR and FSR do not show significant change in error. Additionally, due to the particles leaving the domain and having their track length reduced, proper polynomial fits may not have been possible resulting in increasing error for increasing order. The order does not have the same quality of correlation to computational time as the other program settings, however generally higher orders require more computational

time due to larger matrices in the least-squares polynomial regression algorithm. The recommended polynomial order is chosen as the default due to its minimum error and computational time.

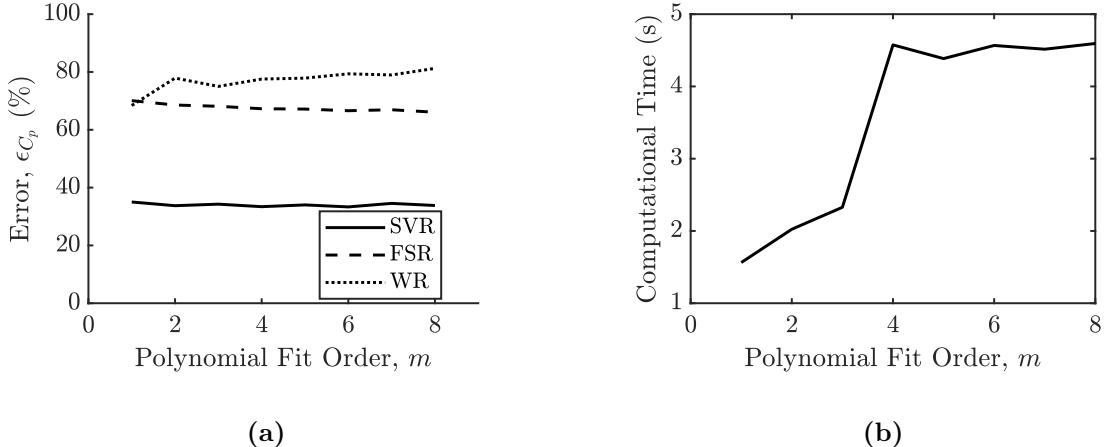


Figure 4.9: (a) Effect of Polynomial Fit Order on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Polynomial Fit Order on Computational Time

The effect of the number of Nearest Neighbours is insignificant as shown in Figure 4.10a, however it was expected that increasing the number would result in a more accurate interpolation due to points having greater spatial information. This setting also has an insignificant effect on computational time in Figure 4.10b, as it is a relatively fast algorithm, and is used sparingly in the program. The baseline number of nearest neighbours of 4 is used as default.

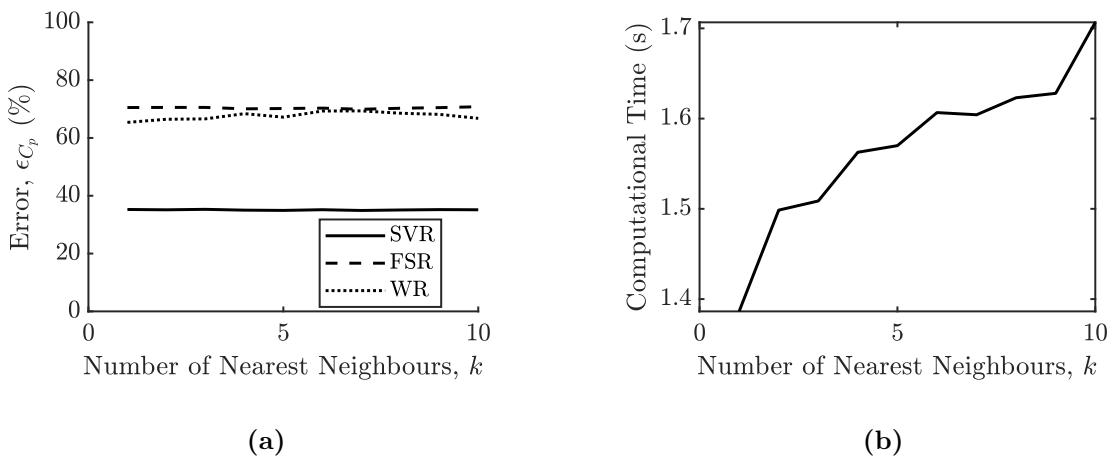


Figure 4.10: (a) Effect of Number of Nearest Neighbours for Interpolation on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Number of Nearest Neighbours for Interpolation on Computational Time

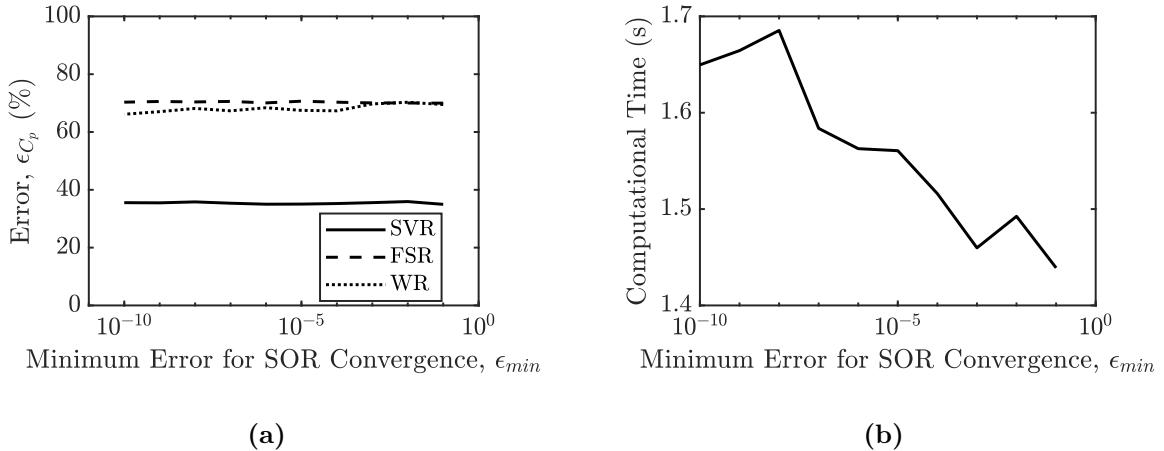


Figure 4.11: (a) Effect of Minimum Error for SOR Convergence for Interpolation on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of Minimum Error for SOR Convergence for Interpolation on Computational Time

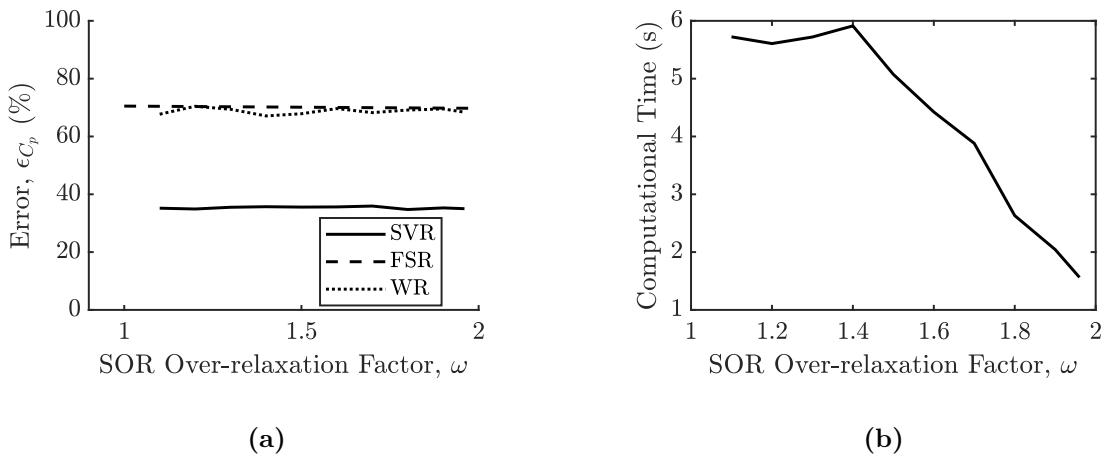


Figure 4.12: (a) Effect of SOR Over-relaxation Factor on Pressure Coefficient Error for WMLES Velocity Field (b) Effect of SOR Over-relaxation Factor on Computational Time

Both the minimum error for SOR convergence, and the SOR over-relaxation factor have similar trends for the pressure coefficient error across all regions in Figures 4.11a and 4.12a. That is that they do not contribute to the mean error across all regions. This is surprising as it is expected that the minimum error between SOR iterations should control the minimum error for the pressure field, as a low minimum error between iterations would likely result in a poorly converged and high error pressure field that is highly dependent on the boundary conditions. Fortunately the SOR Over-relaxation factor does not have an effect on error as expected. This setting should only affect the computational time as it controls the amount of change in the pressure field every iteration, and not the final pressure field. Both settings have the effect of increasing computation time, as they

both allow the solution to converge within less iterations, with the optimal SOR Over-relaxation factor having the computational time for the same error. The baseline values of both settings are used as the default for future cases.

In summary, this program produces unusual behaviour in regards to the effect of program settings on the reconstructed pressure field and therefore the mean pressure coefficient errors for the case of flow around a hemispherical protuberance. This program needs to be verified for more CFD simulated velocity fields with accompanying pressure field data to truly understand why the mean error is independent of the program settings. For now, the new default settings are presented in Table 4.4, and will be used for the following PIV measured velocity fields.

Setting	Symbol	Value
Number of Nearest Neighbours	k	4
Maximum Length of Particle Track	N_t	5
Number of Particles	N_p	1000
Order of Polynomial Fit	m	1
Maximum Number of SOR Iterations	N_{max}	10^4
Minimum Error for SOR Convergence	ϵ_{min}	10^{-6}
SOR Over-Relaxation Parameter	ω	ω_{opt}

Table 4.4: Default Program Settings for Lagrangian Approach and SOR Poisson Solver

4.4 PIV Experiment

Velocity fields on a plane measured by PIV experiments are analysed using the program to reconstruct their pressure coefficient fields on the measured plane. These PIV datasets will demonstrate that the program can reconstruct pressure fields from different empirical sources, compared to approximate computational simulations or analytical potential flows.

4.4.1 Hemispherical Protuberance

2D time-averaged velocity fields of flow around a 100 mm diameter hemispherical protuberance are provided by Franck Bauer (2019). These results were captured using a LaVision Flow Master PIV system with three Phantom Micro110 high speed cameras for 1280 by 720 pixel resolution stereoscopic image acquisition. A thin laser sheet generated by a class 4 Photonics Industries DM Series 527 nm dual cavity laser. This was positioned along the centre-line of the hemisphere mounted to a surface inside the University of Sydney 4 by 3 ft low speed closed circuit wind tunnel. The tunnel was seeded by a LaVision aerosol generator using $10 \mu\text{m}$ DEHS particles. The hemisphere was 3D-printed from acrylonitrile butadiene styrene (ABS) plastic, and coated with Rhodamine to reduce the intensity of reflected laser light. The experiment was conducted at a tunnel speed of 23.8 m/s, an ambient air density of 1.198 kg/m^3 , and an ambient temperature of 23.4°C , producing a free-stream Reynolds number of 1.67×10^5 .

Figure 4.13a presents the time-averaged velocity field (time-dependent fluctuations are absent) measured by Franck Bauer (2019) around the hemispherical protuberance. It reveals the expected structures of such a flow, with a front stagnation point, an extended low velocity region at the front (left) of the hemisphere that likely coincides with a standing vortex, attached flow acceleration from the front to the top of the hemisphere, before the flow separates and a large wake forms behind the hemisphere characterised by low velocity flow. Additionally, the free-stream with contours of 1 is relatively undisturbed.

Its reconstructed pressure field using the 2D Eulerian approach with default SOR Poisson solver settings from Table 4.4 is plotted in Figure 4.13b. The pressure field reproduces the field structures of the relatively high pressure front stagnation point and standing vortex, and the low pressure attached flow region, however it again fails to resolve the negative pressure coefficients expected over the back hemisphere surface as found in Taniguchi et al. (1982). This failing may however be due to the lack of boundary layer information in the

velocity field due to the low resolution SPIV method compared to the CFD simulation in Section 4.2. The pressure field is observably similar to the pressure field reconstructed from CFD data using the Lagrangian approach in Figure 4.6a. This could be due to the fact that the Lagrangian approach effectively time-averages the time-dependent material acceleration using linear regression. Therefore, it is likely that this reconstruction yielded a pressure field with error similar to the Lagrangian approach.

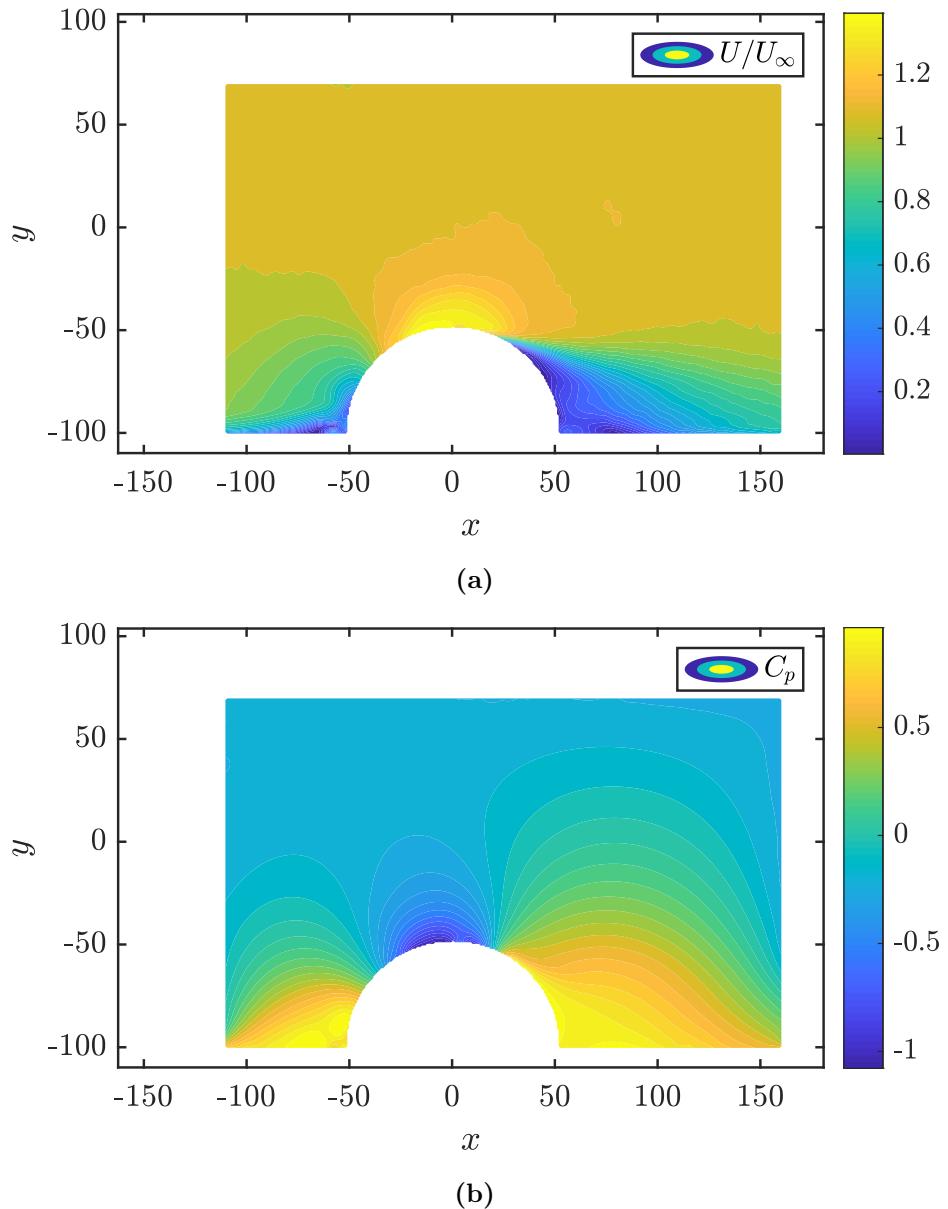


Figure 4.13: (a) Total Velocity Field around a Hemispherical Protuberance from SPIV Measurements. Franck Bauer (2019) (b) Reconstructed Pressure Field from Velocity Field using 2D Eulerian Approach

4.4.2 Unsteady Inviscid Jet Core

Three-component velocity fields of an unsteady inviscid jet core are extracted from the PIV Uncertainty Database created by Neal et al. (2015). According to Neal et al. (2015) an unsteady inviscid jet core is the region of a jet flow that begins to display low frequency velocity fluctuations due to the development Kelvin-Helmholtz vortices around the core as shown in Figure 4.14. The jet flow was captured using two LaVision HighSpeedStar 6 (5400 fps, 1024 by 1024 pixel) cameras in a stereoscopic configuration. The experiment was performed in the Experimental Fluid Dynamics Laboratory at Utah State University, where a Photonics Industries laser was used to illuminate the measurement plane, and the flow was seeded with a Rocket Portable Smoke System using $1 \mu\text{m}$ glycerine-water solution particles. The flow is characterised by a jet exit velocity of 5 m/s, a jet thickness of 10.2 mm, and a jet width of 72.8 mm, resulting in a thickness Reynolds number of 3000.

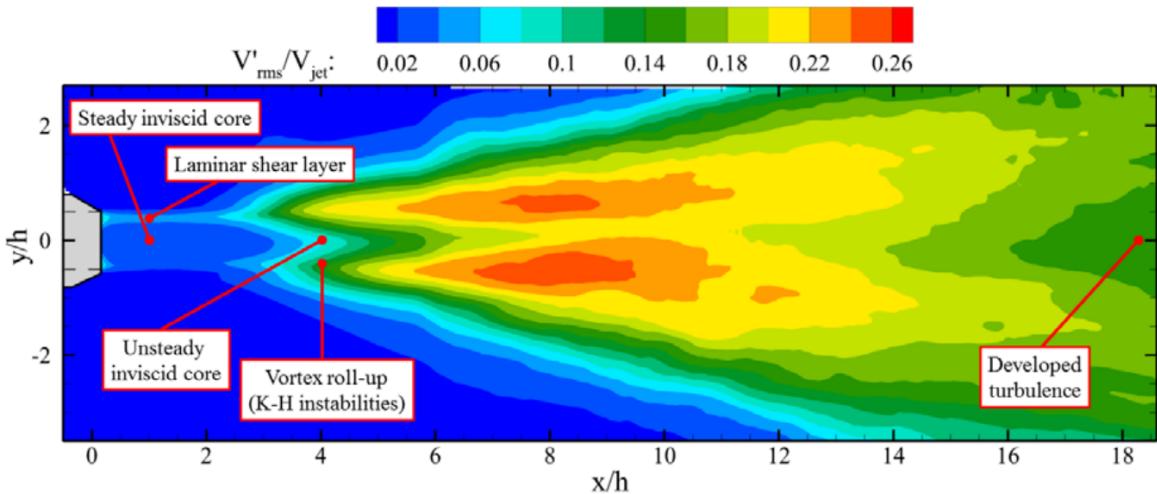


Figure 4.14: Distribution of Velocity Fluctuations over a Jet Flow and its Key Structures. Neal et al. (2015)

Figure 4.15a shows the unsteady jet beginning to axially oscillate as it travels along its path. It is expected that the jet flow is comprised of a laminar high velocity core, and a shell that interacts with the surrounding air in a viscous shear manner lowering the total velocity.

The jet flow pressure field in Figure 4.15b is reconstructed according to the time-dependent Eulerian approach using the default SOR Poisson solver settings in Table 4.4, and a 1/5400 s time-step. This pressure field reproduces the instantaneous oscillating nature of the velocity field, however it does not reproduce its axial symmetry. The pressure field has

two distinct regions of low pressure at the left and right with a higher pressure centre, and ambient pressure in the far-field. The central moderate pressure region is a false artifact of the Eulerian approach, and the low pressure zones from the inviscid Bernoulli equation on the sides should stretch across the domain following the "tube-like" distribution of the velocity field. This is likely due to spatial filtering from the SOR Poisson solver that averages the pressures between the low pressure and high pressure zones.

The Lagrangian approach to pressure reconstruction is presented in Figure 4.15c using all of the default program settings in Table 4.4. This shows a less oscillatory flow in contrast to the pressure field of the Eulerian approach, likely due to the time-averaging associated with the Lagrangian approach. However, the Lagrangian approach also suffers from the spatial filtering leading to a false pressure field like the Eulerian approach.

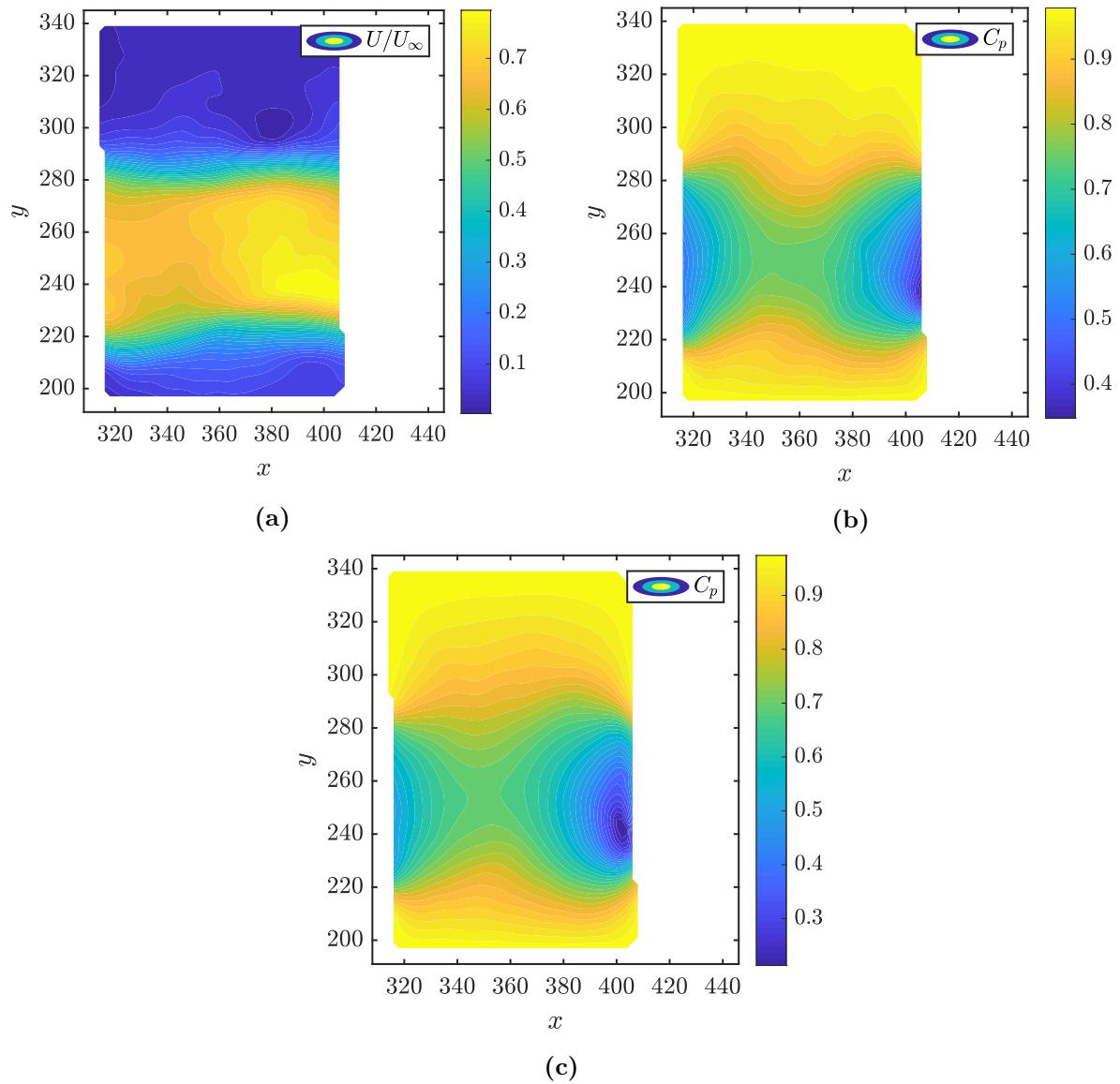


Figure 4.15: (a) Total Velocity Field of an Unsteady Inviscid Jet Core from SPIV Measurements. Neal et al. (2015) (b) Pressure Coefficient Field Reconstructed from Velocity Field using Time-Dependent Eulerian Approach (c) Pressure Coefficient Field Reconstructed from Velocity Field using Lagrangian Approach

Chapter 5

Conclusion & Recommendations

A pressure field reconstruction program has been implemented in the MATLAB programming language for intended use with planar velocity fields derived from PIV measurements. The program and numerical method implementation built upon solutions of the Navier-Stokes equations with initial conditions satisfied by velocity fields has been verified by comparing the known pressure fields of analytical impinging jet and hemispherical protuberance potential flows, and a numerical hemispherical protuberance WMLES CFD simulation. The program was verified to produce pressure fields from velocity fields measured from SPIV experiments.

Unfortunately, the program could not reconstruct pressure fields to the 5-10% mean pressure error that would satisfy standards in literature, with the minimum mean pressure error found to be 35% with the Lagrangian approach. However, the program was built to fulfill most of the research objectives set out in Section 3:

- It is generic to any type of flow: verified for junction and jet flows
- It is readable by engineers and interested individuals as the design is laid out in Section 3
- It is designed with a breadth of user-defined settings described in Section 3.2.9
- The Eulerian and Lagrangian material acceleration estimation approaches were compared and contrasted in Section 4.2.2
- The effect of the user-defined settings on the pressure field reconstruction error were investigated in Section 4.3

- The optimal settings configuration for lowest pressure field reconstruction error was determined
- It is freely available to download at github.com

Furthermore, task of design and implementing this pressure field reconstruction program revealed the following key findings:

- The Lagrangian approach is more robust to noise, and advection dominated flows compared to the Eulerian approach as supported by its low material acceleration error
- The Lagrangian approach is more accurate than the Eulerian approach for flow around a hemispherical protuberance when using the relative pressure coefficient error metric
- The Lagrangian approach produces time-averaged pressure fields
- The Eulerian approach produces approximately instantaneous pressure fields using either the 2D or planar 3D time-dependent forms
- Both approaches suffer from spatial-filtering in the SOR Poisson solver, which is particularly obvious for flows with defined shear layers
- The settings available for the Lagrangian approach and SOR Poisson solver do not have a significant impact on the relative pressure coefficient error for flow around a hemispherical protuberance

To rectify the issues of pressure field reconstruction accuracy, spatial-filtering from the SOR Poisson solver, and program setting effects, the author recommends that program verification be performed for another CFD simulation similar to the empirical jet flow case in Section 4.4 without a domain obstruction and all domain edges are outlets (not bounded by walls). Additionally, a second order particle track construction algorithm can be implemented as in van Gent et al. (2018), which requires the inclusion of the material acceleration and therefore iteration. An omni-directional direct integration algorithm can be implemented to reconstruct the pressure field from the pressure gradient, and mixed (Dirichlet and Neumann) boundary conditions could be used to improve reconstruction at domain outlets.

Chapter 6

References

- Acarlar, M. S. and Smith, C. R. (1987). A study of hairpin vortices in a laminar boundary layer. part 1. hairpin vortices generated by a hemisphere protuberance. *Journal of Fluid Mechanics*, 175:1–41.
- Adrian, R. and Yao, C. S. (1984). Development of pulsed laser velocimetry (plv) for measurement of turbulent flow. In *Proc. Symp. Turbul.*, pages 170–186.
- Adrian, R. J. and Westerweel, J. (2010). *Particle Image Velocimetry*. Cambridge University Press, 32 Avenue of the Americas, New York, NY USA.
- ANSYS (2020). *Fluent 2021 R1 Theory Guide*. ANSYS Inc.
- Barlow, J. B., William H. Rae, J., and Pope, A. (1999). *LOW-SPEED WIND TUNNEL TESTING*. John Wiley & Sons, 605 Third Avenue, New York, NY USA.
- Baur, T. and Kongeter, J. (1999). Piv with high temporal resolution for the determination of local pressure reductions from coherent turbulent phenomena. In *3rd Int. Workshop on Particle Image Velocimetry (Santa Barbara)*, pages 101–106.
- da Vinci, L. (1510). Studies of turbulent water. Manuscript, Royal Collection Trust.
- de Kat, R. and van Oudheusden, B. W. (2012). Instantaneous planar pressure determination from piv in turbulent flow. *Exp Fluids*, 52:1089–1106.
- der Kindere, J. W. V., Laskari, A., Ganapathisubramani, B., and de Kat, R. (2019). Pressure from 2d snapshot piv. *Exp Fluids*, 60(32).
- Franck Bauer, A. (2019). *Particle Image Velocimetry and its Applications to Flow Around a Hemispherical Protuberance*. Honours thesis.

Franck Bauer, A., McCarthy, J., and Giacobello, M. (2018). Flow visualisation around a hemispherical protuberance in the dst group water tunnel. Report DST-Group-TR-3482, Defence Science and Technology Group.

Gordeyev, S. and Jumper, E. (2010). Fluid dynamics and aero-optics of turrets. *Progress in Aerospace Sciences*, 46:388–400.

Grant, I. (1997). Particle image velocimetry: A review. In *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, volume 211, pages 55–76.

Gurka, R., Liberzon, A., Hefet, D., Rubinstein, D., and Shavit, U. (1999). Computation of pressure distribution using piv velocity data. In *3rd Int. Workshop on Particle Image Velocimetry*.

Hoffman, J. D. (2001). *Numerical Methods for Engineers and Scientists*. Marcel Dekker, 270 Madison Avenue, New York, NY USA.

Imaichi, K. and Ohmi, K. (1983). Numerical processing of flow-visualization pictures - measurement of two-dimensional vortex flow. *Journal of Fluid Mechanics*, 129:283–311.

Jeon, Y. J., Chatellier, L., Beaudoin, A., and David, L. (2015). Least-square reconstruction of instantaneous pressure field around a body based on a directly acquired material acceleration in time-resolved piv. In *11TH INTERNATIONAL SYMPOSIUM ON PARTICLE IMAGE VELOCIMETRY – PIV15*.

John D. Anderson, J. (2017). *Fundamentals of Aerodynamics*. McGraw-Hill Education, 2 Penn Plaza, New York, NY 10121.

Liu, X. and Katz, J. (2006). Instantaneous pressure and material acceleration measurements using a four-exposure piv system. *Experiments in Fluids*, 41:227–240.

Neal, D. R., Sciacchitano, A., Smith, B. L., and Scarano, F. (2015). Collaborative framework for piv uncertainty quantification: the experimental database. *Meas. Sci. Technol.*, 26(7).

Pan, Z., Whitehead, J., Thomson, S., and Truscott, T. (2016). Error propagation dynamics of piv-based pressure field calculations: How well does the pressure poisson solver perform inherently? *Meas. Sci. Technol.*, 27(8).

Piomelli, U., Moin, P., and Ferziger, J. H. (1988). Model consistency in large eddy simulation of turbulent channel flows. *The Physics of fluids*, 31(7):1884–1891.

- Prandtl, L. (1927). Entstehung von wirbeln bei wasserströmungen - 1. entstehung von wirbeln und künstliche beeinflussung der wirbelbildung. Film, Reichsanstalt für Film und Bild in Wissenschaft und Unterricht (RWU).
- Raffel, M., Willert, C. E., Scarano, F., Kähler, C. J., Wereley, S. T., and Kompenhans, J. (2018). *Particle Image Velocimetry: A Practical Guide*. Springer, Gewerbestrasse 11, 6330 Cham, Switzerland.
- Royal Australian Air Force (2021). MQ-4C Triton Unmanned Aircraft System. <https://www.airforce.gov.au/technology/aircraft/intelligence-surveillance-and-reconnaissance/mq-4c-triton-unmanned-aircraft>. Accessed 24 Oct 2021.
- Royal Australian Navy (2021). ScanEagle. <https://www.navy.gov.au/unmanned-systems/scaneagle>. Accessed 24 Oct 2021.
- Russell, S. and Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. Pearson Education, Inc., 221 River Street, Hoboken, NJ 07030.
- Savory, E. and Toy, N. (1986). Hemispheres and hemisphere-cylinders in turbulent boundary layers. *Journal of Wind Engineering and Industrial Aerodynamics*, 23:345–364.
- Schneiders, J. F. G., Probsting, S., Dwight, R. P., van Oudheusden, B. W., and Scarano, F. (2016). Pressure estimation from single-snapshot tomographic piv in a turbulent boundary layer. *Exp Fluids*, 57(53).
- Schroeder, A. and Willert, C. E. (2008). *Particle Image Velocimetry: New Developments and Recent Applications*. Springer, Gewerbestrasse 11, 6330 Cham, Switzerland.
- Schwabe, M. (1935). Über die druckermittlung in der nichtstationaren ebenen stromung. *Ing. Arch.*, 6:34–50.
- Shur, M. L., Spalart, P. R., Strelets, M. K., and Travin, A. K. (2008). A hybrid rans-les approach with delayed-des and wall-modelled les capabilities. *International Journal of Heat and Fluid Flow*, 29(6):1638–1649.
- Smagorinsky, J. (1963). General circulation experiments with the primitive equations: I. the basic experiment. *Monthly weather review*, 91(3):99–164.
- Taniguchi, S., Sakamoto, H., Kiya, M., and Arie, M. (1982). Time-averaged aerodynamic forces acting on a hemisphere immersed in a turbulent boundary. *Journal of Wind Engineering and Industrial Aerodynamics*, 9:257–273.

van Gent, P. L., Michaelis, D., van Oudheusden, B. W., Weiss, P. E., and de Kat, R. (2017). Comparative assessment of pressure field reconstructions from particle image velocimetry measurements and lagrangian particle tracking. *Experiments in Fluids*, 58(33).

van Gent, P. L., Schrijer, F. F. J., and van Oudheusden, B. W. (2018). Assessment of the pseudo-tracking approach for the calculation of material acceleration and pressure fields from time-resolved piv: part i. error propagation. *Meas. Sci. Technol.*, 29(4).

van Oudheusden, B. W. (2013). Piv-based pressure measurement. *Meas. Sci. Technol.*, 24(3).

Wang, C. P. (1988). Laser doppler velocimetry. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 40:309–319.

Westerweel, J. (1993). *Digital Particle Image Velocimetry: Theory and Application*. Delft University Press, 2628 CN Delft, the Netherlands.

Willert, C., Schulze, M., Waltenspul, S., Schanz, D., and Kompenhaus, J. (2019). Prandtl’s flow visualization film c1 revisited. In *13th International Symposium on Particle Image Velocimetry – ISPIV 2019*.

Wood, J. N., De Nayer, G., Schmidt, S., and Breuer, M. (2016). Experimental investigation and large-eddy simulation of the turbulent flow past a smooth and rigid hemisphere. *Flow, Turbulence and Combustion*, 97:79–119.