

# **Exploring and Envisioning Periodic Laminar Flow Around a Cylinder**

Miguel Darío Ortega López

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of

Master of Science  
in  
Engineering Mechanics

Clinton L. Dancey, Co-Chair  
Mechanical Engineering

Ronald D. Kriz, Co-Chair  
Engineering Science and Mechanics

Saad A. Ragab  
Engineering Science and Mechanics

April 24, 2009  
Blacksburg, Virginia

Keywords: Incompressible Flow, Viscous Flow, Fluid Simulation.

Copyright © 2009, Miguel Darío Ortega López

# Exploring and Envisioning Periodic Laminar Flow Around a Cylinder

Miguel Darío Ortega López

## (ABSTRACT)

It is well known that for small Reynolds numbers, flow around a cylinder is laminar and stable. For larger Reynolds numbers, although the flow regime remains laminar, the formation of complex periodic structures appear downstream. The cyclic nature of this periodic flow is well characterized by the vortex shedding frequency and Strouhal number. However, complexities of these periodic structures downstream continue to be a topic of research. Periodic laminar 2D incompressible viscous flow around a cylinder is simulated using OpenFoam, an open source computational fluid dynamics program. To better understand these complex structures downstream, a customized computer graphical tool, VerFlow-V.01, was created to analyze and study OpenFoam simulation results. This study includes an investigation of calculating the details of drag and lift coefficients for the cylinder using mathematical models that integrate properties in subdomains, an approach not previously explored to the knowledge of the author. Numerical integration is accomplished using a finite difference approach for solving surface and contour integrals in subdomains of interest. Special attention is given to pressure and to the second invariant of the velocity gradient, as they have a clear mathematical relationship, which is consistent with results previously published. A customized visual data analysis tool, called VerFlow-V.01, allowed investigators to compare simulation data variables in a variety of useful ways, revealing details not previously understood. Main subroutines and a user's manual are included as appendices to encourage reproducibility and future development of the numerical, analytical and graphical models developed here. Together these models resulted in a new understanding of periodic laminar flow around a cylinder. A unique approach was developed to *qualitatively* understand the origins of drag and lift coefficients associated with properties mapped as images in subdomains of interest downstream. These results explain the development of convergent, eddy, and stream zones embedded in flow fields downstream.

# Contents

(ABSTRACT) .....	ii
List of Figures .....	v
List of Tables .....	ix
List of Animations .....	x
Grant Information .....	xi
Dedication .....	xii
Acknowledgements.....	xiii
Introduction .....	1
Chapter 1 . Literature Review .....	3
1.1. Governing Equations for Incompressible Viscous Flows .....	3
1.2. The second invariant of the velocity gradient .....	4
1.3. Flow around a cylinder.....	9
1.4. Frequency analysis .....	14
1.5. Chapter nomenclature.....	16
Chapter 2 . Numerical Simulation: OpenFoam .....	19
2.1. The problem.....	19
2.2. Boundary conditions .....	22
2.3. The grid .....	22
2.4. Simulation time interval.....	27
2.5. Generating OpenFoam simulation “data” .....	29
2.6. Preliminary graphical results using ParaView .....	31
2.7. Frequency and time resolution optimization using VerFlow_V.01 .....	33
2.8. Chapter nomenclature.....	44
Chapter 3 . Mathematical and Numerical Models.....	47
3.1. Pressure independent of time .....	48
3.2. Q: the second invariant of the velocity gradient .....	52
3.3. Forces acting on the cylinder .....	56
3.4. Alternative equation for the pressure at a point $P$ .....	62
3.5. Chapter nomenclature.....	77

Chapter 4 . Results and Discussion .....	82
4.1. Forces on the cylinder.....	82
4.2. Prediction of the pressure at a point.....	94
4.3. Point effect on the cylinder boundary .....	102
4.4. Contributions from $Q$ to the drag and lift forces.....	105
4.5. Convergence and eddy zones .....	111
4.6. Verification of finite difference forms of $Q$ using VerFlow-V.01 .....	121
4.7. Verification of the pressure distribution around the cylinder.....	124
4.8. Chapter nomenclature.....	128
Chapter 5 . Conclusions.....	131
5.1. Programs .....	131
5.2. Forces on the cylinder.....	132
5.3. The second invariant of the velocity gradient, $Q$ .....	133
5.4. Additional <i>qualitative</i> remarks .....	135
References.....	137
Appendix A. General guide to modify the original OpenFoam Simulation.....	138
Appendix B. VerFlow-V.01: Main Subroutines.....	142
Appendix C. VerFlow-V.01: User's Manual .....	172
Appendix D. Solution to Poisson Equation in 2D .....	193

## List of Figures

Figure 1.1 Representation of terms in Equation 1.8.....	5
Figure 1.2 Representation of terms in Equation 1.9. Velocities in white circles are calculated from the nearest in gray circles. ....	6
Figure 1.3 General flow around a cylinder.....	9
Figure 1.4 Drag coefficient and Steady (S), Periodic Laminar (PL) and Periodic Turbulent (PT) regions for cylinders. ....	10
Figure 1.5 Regimen flows.....	11
Figure 1.6 Sequence of pictures showing fluid moving backward in the reverse region (M. Ortega 2009) .....	12
Figure 1.7 Vortex behind a cylinder .....	13
Figure 1.8 Vortices generated behind a cylinder which moves from right to the left (M. Ortega 2009) ...	13
Figure 1.9 Detail of vortices after a cylinder passed from right to the left (M. Ortega 2009).....	13
Figure 1.10 Vortices in the wake when the cylinder is moving to the left .....	15
Figure 1.11 Points(i,j)=(50,20) in each of four blocks selected for history graph in Figure 1.12 .....	15
Figure 1.12 History graph of the dimensionless horizontal velocities for selected points in Figure 1.1 ....	16
Figure 2.1 General dimensions .....	22
Figure 2.2 Blocks .....	23
Figure 2.3 Blocks 0, 1, 2 and 3 (80x80 cells each) and local curvilinear reference axis.....	23
Figure 2.4 Point ( $i = 10$ , $j = 55$ ) at each block and grid detail close to the cylinder (from $j = 51$ to $j = 80$ ).....	24
Figure 2.5 Basic geometry for blocks 0, 1, 2 and 3 .....	25
Figure 2.6 Block 4 (240x80 cells).....	27
Figure 2.7 OpenFoam result for the pressure per unit density field .....	31
Figure 2.8 OpenFoam result for the vorticity field .....	32
Figure 2.9 Selected point in the wake and negative $u_x$ .....	33
Figure 2.10 $u_x$ as a function of time for periodic behavior ( $u_\infty = 0.01 \text{ m/s}$ ).....	33
Figure 2.11 Primary selection for one cycle from time <b>24.2 s</b> to time <b>26.6 s</b> .....	34
Figure 2.12 $u_x$ data for one cell in nine cycles and one hundred nine consecutive times .....	35
Figure 2.13 Instantaneous frame showing $u_x$ for the time <b>24.2 s</b> , color legend Figure 2.14. ....	36
Figure 2.14 Legend for Figures 2.13, 2.15 – 2.19 .....	37
Figure 2.15 $u_x$ contour lines for <b>45.8 s</b> in yellow compared over the entire region for <b>24.2 s</b> , color legend Figure 2.14 .....	37
Figure 2.16 $u_x$ contour lines for <b>46.0 s</b> in yellow compared over the entire region for <b>24.2 s</b> , color legend Figure 2.14 .....	38
Figure 2.17 $u_x$ contour lines for <b>46.2 s</b> in yellow compared over the entire region for <b>24.2 s</b> , color legend Figure 2.14 .....	38
Figure 2.18 $u_x$ contour lines for <b>43.6 s</b> in yellow compared over the entire region for <b>24.2 s</b> , color legend Figure 2.14 .....	38

Figure 2.19 $ux$ contour lines for <b>48.4 s</b> in yellow compared over the entire region for <b>24.2 s</b> , color legend Figure 2.14 .....	39
Figure 2.20 Magnitude of the Fourier Coefficients vs. frequency .....	40
Figure 2.21 Increased resolution for one cycle.....	44
Figure 3.1 Forces originated by the pressure on the cylinder at a shaded cell .....	57
Figure 3.2 Individual contributions to the tangential velocity of arbitrary positive $u$ and $v$ .....	58
Figure 3.3 Viscous force and Velocity components for an arbitrary counter clockwise $veltg$ .....	59
Figure 3.4 Two dimensional domains defined (a) Arbitrary subdomain, (b) rectangular subdomain and (c) entire domain.....	63
Figure 3.5 Arbitrary point in a rectangular subdomain inside block 4. ....	64
Figure 3.6 Distance $r(i, j)$ between points $Q$ (where the second invariant is defined) and $P$ (where the pressure is calculated) .....	65
Figure 3.7 Left boundary detail and definitions for $CI1L$ .....	65
Figure 3.8 Right boundary detail and definitions for $CI1R$ .....	66
Figure 3.9 Top boundary detail and definitions for $CI1T$ .....	67
Figure 3.10 Bottom boundary detail and definitions for $CI1B$ .....	68
Figure 3.11 Left boundary detail and definitions for $CI2L$ .....	68
Figure 3.12 Right boundary detail and definitions for $CI2R$ .....	69
Figure 3.13 Top boundary detail and definitions for $CI2T$ .....	70
Figure 3.14 Bottom boundary detail and definitions for $CI2B$ .....	71
Figure 4.1 Drag coefficient of a cylinder for $Re=67$ .....	82
Figure 4.2 Pressure and viscous forces on the cylinder.....	83
Figure 4.3 Times of maximum forces <b>0.3s</b> top left, <b>0.5s</b> top right, <b>1.3s</b> bottom left and <b>1.5s</b> bottom right.....	85
Figure 4.4 Drag forces on the cylinder along one cycle .....	86
Figure 4.5 Amplification (100 times) of drag forces shown on Figure 4.4 for one complete cycle .....	86
Figure 4.6 Drag (top) and lift (bottom) oscillations for Fast Fourier Transform analysis (see Figure 4.7)..	87
Figure 4.7 Drag and lift oscillations frequency spectrums.....	87
Figure 4.8 Lift forces on the cylinder along one cycle .....	88
Figure 4.9 Contributions to drag (left) and lift (right) coefficients from arc 0 (in one cycle) .....	89
Figure 4.10 Contributions to drag (left) and lift (right) coefficients from arc 1 (in one cycle) .....	90
Figure 4.11 Contributions to drag (left) and lift (right) coefficients from arc 2 (in one cycle) .....	90
Figure 4.12 Contributions to drag (left) and lift (right) coefficients from arc 3 (in one cycle) .....	91
Figure 4.13 Drag (left) and lift (right) coefficient contributions from arcs 0, 1, 2 and 3 due to pressure (in one cycle) .....	91
Figure 4.14 Pressure drag coefficients in one cycle: Total (left), sum arcs 3 and 1 (center), and, sum arcs 0 and 2 (right) .....	92
Figure 4.15 Drag (left) and lift (right) coefficient contributions from arcs 0, 1, 2 and 3 due to viscous flow (in one cycle).....	92
Figure 4.16 Viscous drag coefficients in one cycle: Total (left), sum arcs 3 and 1 (center), and, sum arcs 0 and 2 (right) .....	93
Figure 4.17 Total drag coefficient in one cycle .....	93

Figure 4.18 Total drag (left) and pressure drag sum from arcs 1 and 3 (right) for $u_x = 0.01[cm/s]$ in one cycle .....	94
Figure 4.19 Pressure predicted at an arbitrary point in a rectangular domain at time $0.4(s)$ .....	96
Figure 4.20 Arbitrary point in the biggest rectangle at block 4, see color legend in Figure 4.19. ....	97
Figure 4.21 Pressure at an arbitrary point for the biggest rectangle in block 4. ....	97
Figure 4.22 Arbitrary point in the entire domain, see color legend in Figure 4.19. ....	98
Figure 4.23 Pressure at an arbitrary point for the entire domain .....	98
Figure 4.24 Pressure at a point for non-filtered data (rectangle domain) .....	100
Figure 4.25 Pressure at a point for filtered data (rectangle domain) .....	101
Figure 4.26 Pressure at a point for filtered data (entire domain) .....	101
Figure 4.27 Pressure at a point for filtered data (entire domain) - results .....	101
Figure 4.28 Contribution of red points $Q$ to the pressure at each cell along the cylinder boundary .....	103
Figure 4.29 Effect of points $Q$ close to the cylinder on pressure at its boundary .....	104
Figure 4.30 Contribution of each point in the domain to the cylinder drag (Min: Yellow, Max: Blue, zero: White) .....	106
Figure 4.31 Closer view of drag contributions to the cylinder from the second invariant of the velocity gradient field.....	106
Figure 4.32 Envisioning multiple instantaneous filtered information over the positive drag contributions from $Q$ .....	107
Figure 4.33 Evolution in time of positive contributions of $Q$ to the drag.....	109
Figure 4.34 Contribution of each point in the domain to the cylinder lift (Min: Yellow, Max: Blue, zero: White) .....	110
Figure 4.35 Closer view of lift contributions to the cylinder from the second invariant of the velocity gradient field.....	110
Figure 4.36 Fluid zones from Hunt paper with its original caption (Hunt, Wray, & Moin, 1988).....	112
Figure 4.37 Zones in the flow around a cylinder for a frame of reference moving at the mean velocity	113
Figure 4.38 Convergence zone C1 at the inlet (moving frame) .....	113
Figure 4.39 Counter clockwise eddy E1 (moving frame) .....	114
Figure 4.40 Convergence zone C2 (moving frame).....	114
Figure 4.41 eddy zone E2 (moving reference) .....	115
Figure 4.42 Streamlines (moving frame) over pressure (green: Min, blue: Max) .....	116
Figure 4.43 Streamlines (moving frame) over second invariant of velocity gradient (green: Min, blue: Max) .....	116
Figure 4.44 Streamlines (moving frame) over vorticity (green: Min, blue: Max) .....	117
Figure 4.45 Streamlines (static frame).....	117
Figure 4.46 Streamlines (static frame) near to the cylinder .....	118
Figure 4.47 Eddies (static frame) and stagnation points at the cylinder .....	119
Figure 4.48 Streamlines (static frame) over pressure (green: Min, blue: Max).....	119
Figure 4.49 Streamlines (static frame) over second invariant of velocity gradient (green: Min, blue: Max) .....	120
Figure 4.50 Streamlines (static frame) over vorticity (green: Min, blue: Max) .....	120
Figure 4.51 Legend bar for Figures 4.52, 4.53 and 4.54 .....	121

Figure 4.52 Second invariant of the velocity gradient from OpenFoam, $Q$ .....	122
Figure 4.53 $Q$ applying Equation 3.35 with a central difference approach in block 4.....	123
Figure 4.54 $Q$ applying Equation 3.36 with a central difference approach in block 4.....	123
Figure 4.55 $Q$ applying Equation 3.36 with a forward difference approach in block 4.....	123
Figure 4.56 $Q$ applying Equation 3.37 with a central difference approach in block 4.....	123
Figure 4.57 Instantaneous pressure distribution along the cylinder boundary envisioned by VerFlow-V.01 .....	124
Figure 4.58 Instantaneous pressure distribution as a function of the angle from the horizontal. ....	124
Figure 4.59 Pressure distribution for small Reynolds numbers (Churchill, 1988) .....	125
Figure 4.60 Comparison of instantaneous and mean pressure with experimental data (Churchill, 1988) .....	125
Figure 4.61 Predicted and simulated pressure along the cylinder boundary for $t = 0.35[s]$ .....	126
Figure 4.62 Integral components for the pressure along the cylinder boundary for $t = 0.35[s]$ .....	127
Figure 4.63 Components of (CI1) from each boundary for $t = 0.35[s]$ .....	128
Figure 4.64 Components of (CI2) from each boundary for $t = 0.35[s]$ .....	128
Figure C.1 VerFlow-V.01 window zones.....	172
Figure C.2 VerFlow-V.01 color legend bar.....	174
Figure C.3 Initial tab in Zone 2.....	175
Figure C.4 Streamwise velocity field (default) displayed in zone 4 after reading.....	176
Figure C.5 Exit alert.....	177
Figure C.6 Selection of variables.....	177
Figure C.7 Adjust tab in zone 2.....	179
Figure C.8 Envisioning negative pressure and positive $Q$ example.....	179
Figure C.9 Dimensional values for an instantaneous mouse pointer location in zone 4.....	180
Figure C.10 Dimensionless values for an instantaneous mouse pointer location in zone 4.....	181
Figure C.11 Labels in the Setting tab in zone 4.....	182
Figure C.12 Settings tab example. Note the effect of checking the box for stagnation points in the View label.....	183
Figure C.13 Representation of pressure and viscous forces over $Q$ .....	184
Figure C.14 Representation of pressure distribution from OpenFoam numerical “data” .....	186
Figure C.15 Representation of pressure distribution originated from an arbitrary cell (enlighten in red) .....	187
Figure C.16 Integration options for a rectangular subdomain .....	188
Figure C.17 Integration points for the entire domain .....	190
Figure C.18 $Q$ drag-lift & more tab options .....	191
Figure C.19 Representation of $Q$ drag after calculations realized on $Q$ drag-lift & more tab .....	191
Figure D.1 Region definitions.....	194
Figure D.2 Circular zone outside the region.....	194

## List of Tables

Table 2.1 Velocity and diameter required for water and $Re=70$ .....	21
Table 2.2 Velocities and diameter selected for simulation .....	21
Table 2.3 Conversion to a unique coordinate system .....	26
Table 2.4 Reference values for the maximum time interval allowed for flow in constricted region .....	28
Table 2.5 Relevant information in the <i>controlDict</i> dictionary .....	29
Table 2.6 Period and main frequencies obtained from three different methods .....	41
Table 2.7 Original nine cycles data .....	42
Table 2.8 Rotated data to reach a new order .....	43
Table 2.9 New names for the 109 time points in one cycle.....	43
Table 4.1 Drag contributions from integrals in Equation 3.84.....	127
Table C.1 Variables.....	178

## List of Animations

Animation 1.1 Fluid moving backward in the reverse flow region (M. Ortega 2009).....	12
Animation 4.1 Forces on the cylinder.....	84
Animation 4.2 Pressure at a point on a rectangular domain.....	96
Animation 4.3 Pressure at a point on the entire domain.....	98
Animation 4.4 Contributions of $Q$ to the drag.....	108
Animation 4.5 Contributions of $Q$ to the lift.....	110
Animation 4.6 Stagnation points on the cylinder, vorticity field background.....	119

## **Grant Information**

Several Institutions graciously opened their doors to my family and myself that changed our lives during the time I pursued this Master's Degree.

I acknowledge the Fulbright Commission – Ecuador for honoring me with a Fulbright Faculty Development Program Grant. The Fulbright scholarship is administered by LASPAU, which has been my sponsor. I am very grateful to the two Universities where I have studied while in the United States, the University at Buffalo in the English Training Program and Virginia Tech in the Graduate Program.

I especially recognize not only the financial but also the morale support I received from the Fulbright scholarship, the National Polytechnic School – Ecuador, the Graduate School, and the Engineering Science and Mechanics Department at Virginia Tech during these two years.

Thanks to all the people who participated in my behalf at these Institutions.

## **Dedication**

To Sofía and Pablo, for they are the daily energy in my life.

To Magy, for she is the lovely heart and soul of my dear family.

To my father Miguel and my mother Tere, for they gave me all the love and all what they are.

To Ms. Delita, Carli, María Margarita, Carlos Esteban, María Jose, Pol, Togue, Carito, Richi, Dany, Beto, Lore, Gaby, Xime, Maye, Fer and Hólger, for they have been always filling our hearts with love, peace and hope.

## Acknowledgements

A sincere appreciation to everyone who helped in different ways.

Professor Ronald Kriz, my advisor, teacher and friend. From him I enjoyed three excellent courses. He gave me the opportunity and directed me through the whole process at Virginia Tech including serving as co-chair on this thesis.

Professor Clinton Dancey, who, is a co-chair on this thesis, and who provided continuity and guidance throughout. Many discussions of the results and derivations of  $Q$  which was included in the creation of the VerFlow-V.01 computer program.

Professor Saad Ragab, an excellent professional who taught two courses and was a committee member in my thesis, who contributed to the verification of simulation results.

Together my committee, after countless meetings and discussions, contributed to the evolution of ideas used to create and customize VerFlow-V.01.

Patrick Shinpaugh who set up OpenFoam and ParaView on computers used in this thesis.

Brandon Dillon, who openly shared very important details and spent his time when I began using OpenFoam.

Ms. Susan Olivier, who kindly helped me with grammatical corrections.

Irene Furman, her family and Adrian Tola, who are our friends and our dear family at Blacksburg,

Our friends at LAIGSA, Virginia Tech, at the Holy Spirit Catholic church in the Spanish Masses and at our children's Harding Avenue Elementary School.

Our Ecuador, its people, the land, far from here, but always present.

Thanks to God. For you let us live, share and learn from this experience in the United States, which kindly opened its doors to my family and I.

## Introduction

Understanding the complexities of laminar and turbulent flow is a difficult problem that has been studied for many years. Researchers in this field have been creative and innovative by introducing several new techniques and definitions. One of those is the definition of the second invariant of the velocity gradient, which is particularly relevant to vortical flows. In this study, the well known problem of periodic laminar flow around a cylinder is reproduced using the OpenFoam computational fluid dynamic program. Simulation results are studied using VerFlow-V.01, a customized graphical post-processing tool, which reveals details of periodic behavior not previously understood in a *qualitative* sense.

The first chapter is a classical literature review. Papers and books are reviewed that have information related to either the second invariant of the velocity gradient, or the flow around a cylinder with periodic behavior. This information is presented chronologically.

The second chapter defines the problem and shows how to generate simulation data using OpenFoam (a fluid dynamics open source simulation computer program). This chapter also shows how to identify the principal frequencies associated with periodic flow and improve the time resolution for one period by combining frames from several cycles into one continuous cycle.

The third chapter is a mathematical description of equations used in the numerical simulation and VerFlow-V.01 computer programs. Some equations are generally used by the fluid dynamic simulation program and others are focused on specific situations, such as the two dimensional problem or the finite difference forms that were used to calculate pressures and other properties associated with velocity gradients in subdomains associated with convergent, eddy and stream zones.

The fourth chapter is a compilation and discussion of results obtained by using VerFlow-V.01, a graphical post-processing computer program developed in Visual Basic 2008. These results include forces on the cylinder due to pressure and viscous interactions. In this chapter a unique approach for calculating the pressure is developed at any arbitrary point, which is the sum of several factors evaluated by performing numerical integrations in a domain with a known

velocity field, pressure, and pressure gradients along the boundaries. Similarly an additional feature in VerFlow-V.01 was developed to map the effect on the cylinder boundary as lift and drag contributions from the second invariant at a point anywhere in the domain.

The fifth chapter concludes with a discussion of this approach and suggestions for future work.

Unique to this thesis are linked animations that provided an improved dynamic interpretation.

The first appendix contains some guidelines for modification and reproduction of OpenFoam files that facilitates reproduction of results shown here. The second appendix includes the main subroutines developed in VerFlow-V.01, which is copyrighted by the author. The third appendix provides a complete reference of VerFlow-V.01 as a User's Manual. The fourth appendix shows the derivation of the solution of the Poisson equation for the 2D problem.

## Chapter 1 . Literature Review

This chapter describes the general familiar expressions for viscous incompressible flows, such as the Navier-Stokes equations and the continuity equation. Following these basic expressions, a chronological review focuses on how the second invariant  $Q$  of the velocity gradient first appeared and how it became both a variable for a local balance in the Poisson's equation and also a way to identify vortices in periodic fluid flow. Next the review introduces the two-dimensional (2D) expression to calculate the pressure at a point when  $Q$  is known in the domain. This expression is further developed in Chapter 3. Finally a review of empirical parameters, e.g. Reynolds and Strouhal number, and experimental observations of periodic flow contribute to interpreting results predicted by analytic and numerical models.

### 1.1. Governing Equations for Incompressible Viscous Flows

The Navier-Stokes equations for a viscous incompressible flow in a 3D space can be written in indicial form as follows (Frederick & Chang, 1972):

$$\frac{D}{Dt}(u_i) = f_i - \frac{\partial P}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} \quad (1.1)$$

Where three equations come from the free indices  $i = 1, 2$  and  $3$ ,  $x_i$  and  $u_i$  are the spatial and velocity components respectively,  $f_i$  are the acceleration components or body force components due to external fields (e.g. gravitational), the summation convention is used,  $P$  is the pressure  $p$  divided by the density  $\rho$ , and, the kinematic viscosity  $\nu$  is the ratio between the dynamic viscosity  $\mu$  and the density  $\rho$  as follows:

$$P = \frac{p}{\rho} \quad (1.2)$$

$$\nu = \frac{\mu}{\rho} \quad (1.3)$$

Applying the comoving derivative to the left hand side of equation (1.1),

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = f_i - \frac{\partial P}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} \quad (1.4)$$

The continuity equation for an incompressible flow is given by:

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (1.5)$$

## 1.2. The second invariant of the velocity gradient

The following sentence was written almost fifty years ago and is still applicable: “With the advent of the high-speed computer, it has become possible to develop methods for studying theoretically many of the non-steady incompressible fluid flow problems which previously had been hopelessly complicated for analysis” (Fromm, 1963). Fromm was a pioneer in computational fluid dynamics, who presented a complete and detailed technical report in which he developed a finite difference approach for the solution of the previous equations. Surely the complex nature of incompressible flows challenges our understanding and the continuous technological development of computers aids in meeting this challenge. However, fluid dynamic researchers can be overwhelmed by multiple 3D properties embedded in massive datasets generated by these simulations. The complexity of the governing differential equations and the massive simulation data sets contributes little to our intuitive understanding of complex flow structures e.g. vortices, eddies, etc. that are observed experimentally.

Embedded in this vast sea of quantitative simulation data is a *qualitative* understanding. Richard Feynman commented on the *qualitative* forms and features of equations associated with laminar periodic flow; “When we have similar equations in a less familiar situation, and one for which we cannot yet experiment, we try to solve the equations in a primitive, halting, and confused way to try to determine what new *qualitative* features may come out, or what new *qualitative* forms are a consequence of the equations” (Feynman, Leighton, & Sands, 1964). This idea of *qualitative* forms is used in this thesis when describing both mathematical and graphical models.

In this thesis the second invariant,  $Q$ , of the velocity gradient tensor,  $\frac{\partial u_i}{\partial x_j}$ , is used for two reasons. First, because the second invariant of the velocity gradient, calculated by OpenFoam, was ambiguous where a literature review confirmed equation 1.15 was indeed used by OpenFoam. The literature review that follows revealed other equivalent mathematical forms of  $Q$  which are further developed in Chapter 3 and used by VerFlow-V.01 to calculate  $Q$  from the velocity field (see equations 3.35, 3.36, 3.37, 3.49, 3.52 and 3.54). Second,  $Q$  is defined as shown in equations 1.15, to predict the pressure at a point according to equation 3.84, which is used to predict lift and drag contributions discussed in section 4.2.

During the sixties, Fromm and other authors were developing finite difference numerical methods to solve incompressible flow problems. By using the Navier Stokes and the continuity equations as the basic equations, and pressure as an auxiliary variable for 2D problems, Fromm introduced another variable,  $Q_F$ . This variable was used to calculate the pressure at each point for a fixed time through a Poisson equation.

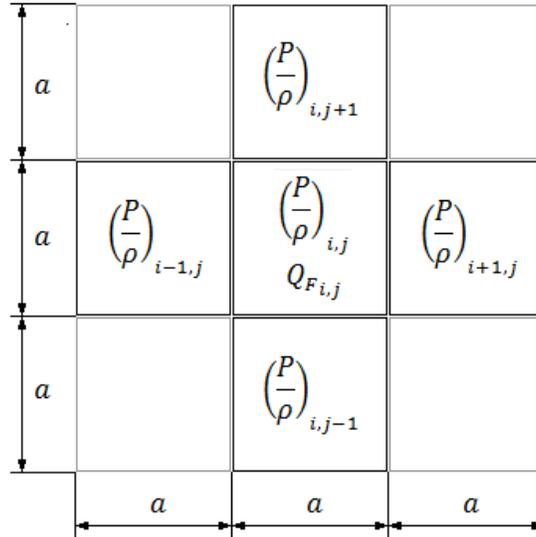


Figure 1.1 Representation of terms in Equation 1.8

The variable  $Q_F$  introduced by Fromm was defined for a 2D problem as:

$$Q_F \equiv \frac{\partial^2 u^2}{\partial x^2} + 2 \frac{\partial^2 uv}{\partial x \partial y} + \frac{\partial^2 v^2}{\partial y^2} \quad (1.6)$$

Fromm also shows the relationship of  $Q_F$  with the pressure as follows:

$$\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} = -\rho Q_F \quad (1.7)$$

Finite difference forms of these equations are written as equations 1.8 and 1.9 below. Figures 1.1 and 1.2 represent each term in these equations. Pressure at cell  $i, j$  is calculated by averaging the neighboring pressures and adding the contribution from the velocity field in its vicinity, given by  $Q_{F,i,j}$ .

$$\left(\frac{P}{\rho}\right)_{i,j} = \frac{1}{4} \left[ \left(\frac{P}{\rho}\right)_{i+1,j} + \left(\frac{P}{\rho}\right)_{i-1,j} + \left(\frac{P}{\rho}\right)_{i,j+1} + \left(\frac{P}{\rho}\right)_{i,j-1} + a^2 Q_{F,i,j} \right] \quad (1.8)$$

$$a^2 Q_{F,i,j} = (u_{i+1,j}^2 + u_{i-1,j}^2 - 2u_{i,j}^2 + v_{i,j+1}^2 + v_{i,j-1}^2 - 2v_{i,j}^2) + 2 \left[ (uv)_{i+\frac{1}{2},j+\frac{1}{2}} - (uv)_{i-\frac{1}{2},j+\frac{1}{2}} + (uv)_{i-\frac{1}{2},j-\frac{1}{2}} - (uv)_{i+\frac{1}{2},j-\frac{1}{2}} \right] \quad (1.9)$$

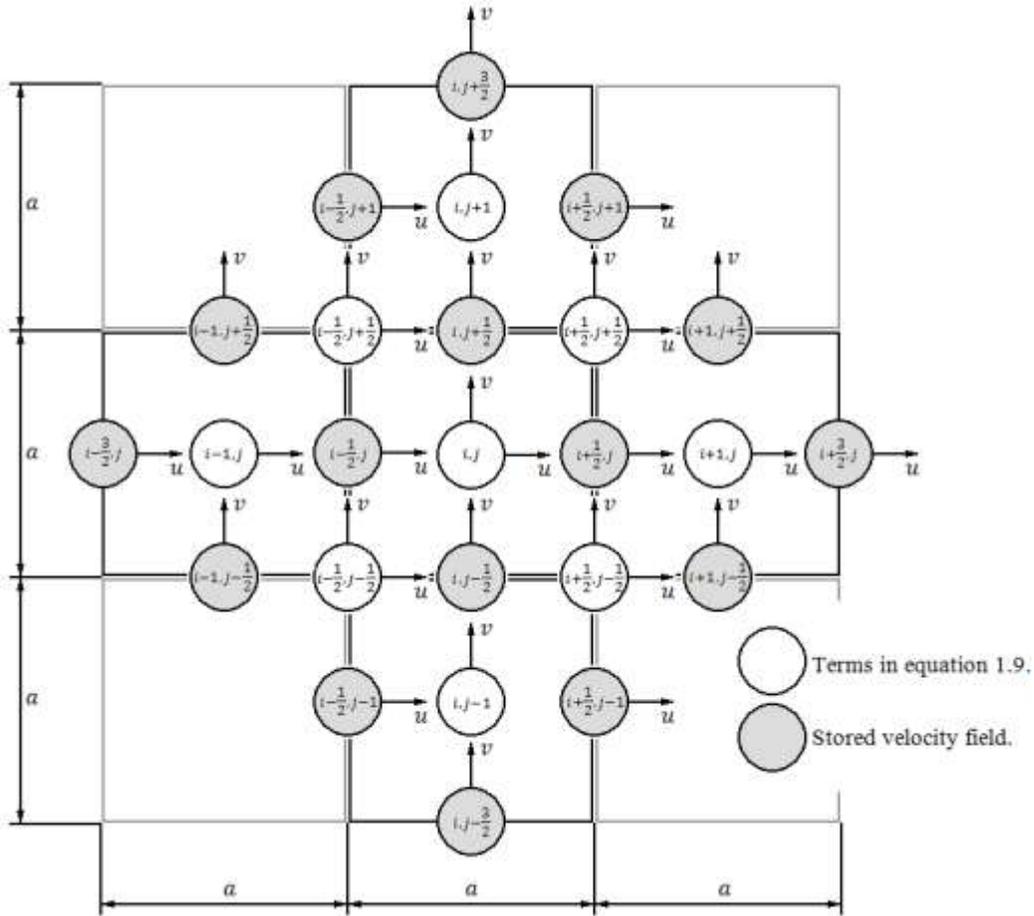


Figure 1.2 Representation of terms in Equation 1.9. Velocities in white circles are calculated from the nearest in gray circles.

The velocities at each point were obtained by averaging the nearest available velocities.

Besides its relationship to pressure, interestingly,  $Q_F$  has also been associated with flow structures in the flow or “eddies” (Sections 3.2.1 and 3.2.2 show how eddies are associated with zones of minimum pressure and minimum  $Q_F$ ), which are defined as regions in which the irrotational straining  $E_{i,j}$  is small when compared with the vorticity, and at the same time the pressure tends to a minimum value (Hunt, Wray, & Moin, 1988). This condition is realized when the second invariant  $II$  of the deformation tensor  $\frac{\partial u_i}{\partial x_j}$  is less than a specific reference value. Using the indicial notation and in a three-dimensional (3D) space, the authors, Hunt et al, defined  $II$  for an incompressible flow as:

$$II = \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} = E_{i,j} E_{i,j} - \frac{1}{2} \omega_i \omega_i \quad (1.10)$$

$$E_{i,j} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (1.11)$$

$$\omega_i = \epsilon_{ijk} \frac{\partial u_k}{\partial x_j} \quad (1.12)$$

Where  $E_{i,j}$  in equation 1.11 is called either rate of strain tensor or the symmetric part of the velocity gradient tensor  $\frac{\partial u_i}{\partial x_j}$ .

Hunt and the other authors also recognized and defined the existence of convergence and stream regions in general flow domains. These concepts will be used later in the *qualitative* investigation of the laminar flow around a circular cylinder.

The characteristic equation of the velocity gradient tensor,  $\frac{\partial u_i}{\partial x_j}$ , from a general form (Mase, 1970), is given by equation 1.13, where  $P$ ,  $Q$  and  $R$  are, respectively, the first, second and third invariants of the velocity gradient tensor,  $\frac{\partial u_i}{\partial x_j}$ .

$$\sigma^3 - P\sigma^2 + Q\sigma - R = 0 \quad (1.13)$$

$P$  is zero for incompressible flow,  $Q$  simplifies (again for incompressible), and  $R$  is the determinant of the velocity gradient:

$$P = \frac{\partial u_i}{\partial x_i} = 0 \quad (1.14)$$

$$Q = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_i} \frac{\partial u_j}{\partial x_j} - \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} \right) = -\frac{1}{2} \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} \quad (1.15)$$

$$R = \text{Det} \left( \frac{\partial u_i}{\partial x_j} \right) \quad (1.16)$$

Equation 1.13 considering  $P = 0$  reduces to:

$$\sigma^3 + Q\sigma - R = 0 \quad (1.17)$$

In 1990, Jeong proposed using the complex eigenvalues  $\sigma$  of the velocity gradient tensor  $\frac{\partial u_i}{\partial x_j}$  to define a vortex core (Jeong & Hussain, 1994). According to Jeong, the eigenvalues  $\sigma$  of equation 1.17 are complex when the discriminant,  $\Delta$ , is positive.

$$\Delta = \left( \frac{1}{3} Q \right)^3 + \left( \frac{1}{2} R \right)^2 > 0 \quad (1.18)$$

Note that  $Q$  in Equation 1.18 has to be positive in regions where a vortex core exists.

Comparing Equations 1.6 and 3.37, which is based on the definition given in equation 1.15 and a mathematical derivation in Chapter 3, gives the relation between  $Q_F$  and  $Q$  as:

$$Q_F = -2Q \quad (1.19)$$

Similarly, comparison of the definitions for the second invariant of the velocity gradient tensor given in equations 1.10 and 1.15 gives:

$$II = -2Q = Q_F \quad (1.20)$$

The definition given in equation 1.15 is used in this thesis for the second invariant of the velocity gradient tensor. Note that the variable used by Fromm,  $Q_F$ , and that used by Hunt,  $II$ , is just  $Q$  but different by a constant.

In 1994, an interesting description of the second invariant of the velocity gradient tensor was stated: “ $Q$  represents the local balance between shear strain rate and vorticity magnitude” (Jeong & Hussain, 1994), which makes sense when we look at the Poisson equation described in equation 1.7 and some of the different ways this variable can be written for incompressible flow.

$$Q \equiv \frac{1}{2} \left[ \left( \frac{du_i}{dx_i} \right)^2 - \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} \right] = -\frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} \frac{\partial u_j}{\partial x_i} \right) = \frac{1}{2} (\|\Omega\|^2 - \|S\|^2) \quad (1.21)$$

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (1.22)$$

$$\Omega_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \quad (1.23)$$

where  $S$  is the symmetric part and  $\Omega$  is the antisymmetric part of the velocity gradient tensor. They also proposed the second largest eigenvalue  $\lambda_2$  of the symmetric tensor  $S^2 + \Omega^2$  to identify the vortex core.

### 1.3. Flow around a cylinder

It is known that the flow around a cylinder is characterized by its Reynolds number, given by the Equation 1.24. Figure 1.3 shows a confined channel in which fluid passes around a cylinder.

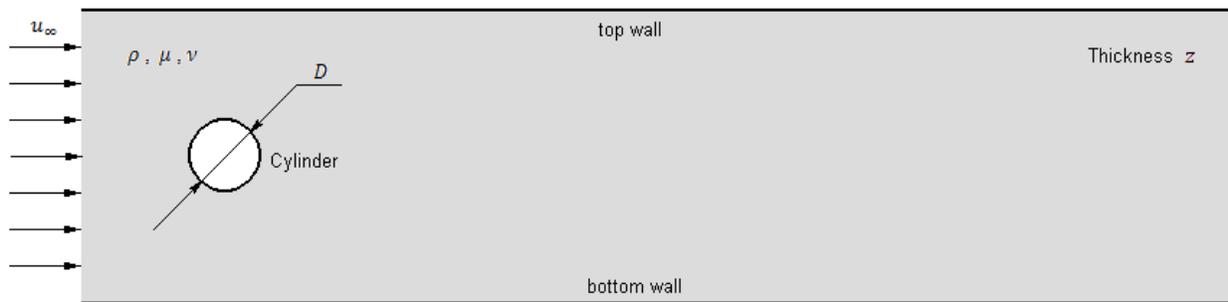


Figure 1.3 General flow around a cylinder

$$\text{Re} = \frac{\rho u_\infty D}{\mu} = \frac{u_\infty D}{\nu} \quad (1.24)$$

Here  $u_\infty$  is the velocity of the fluid at the inlet (left side in Figure 1.3),  $D$  is the characteristic length (in this case the diameter of the cylinder),  $\rho$  is the density,  $\mu$  is the dynamic viscosity and  $\nu$  is the kinematic viscosity.

The drag coefficient  $C_D$  is a number obtained from the expression (Feynman, Leighton, & Sands, 1964):

$$C_D = \frac{F}{\frac{1}{2} \rho u_\infty^2 D z} \quad (1.25)$$

This is a dimensionless coefficient that characterizes the drag force acting on a submerged body where  $z$  is the length of the cylinder. The authors show a graph as a function of the Reynolds number. In Figure 1.4, [adapted from (Feynman, Leighton, & Sands, 1964)] we have extracted from this graph a region that includes the specific Reynolds numbers associated with the simulation in this study.

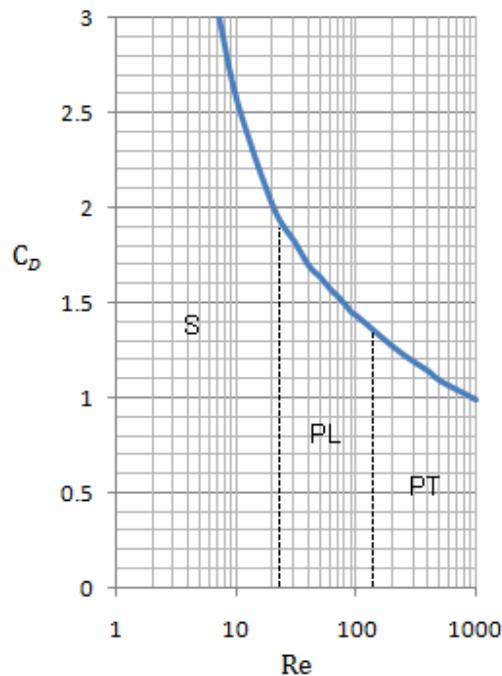


Figure 1.4 Drag coefficient and Steady (S), Periodic Laminar (PL) and Periodic Turbulent (PT) regions for cylinders.

Figure 1.4 also shows three regions for the flow. For relative small Reynolds numbers a steady flow (S) is observed. When the Reynolds number is increased, the regime changes to a periodic laminar (PL) flow. For relatively high Reynolds numbers the flow becomes periodic turbulent (PT). The transition from laminar to periodic laminar flow is highlighted as a dashed vertical line on the left in Figure 1.4.

The coefficient of drag in Figure 1.4 is shown as a function of Reynolds,  $C_D(Re)$ , which differs from one author to another [compare (Feynman, Leighton, & Sands, 1964), (Warsi, 1993), (Kundu & Cohen, 2004), (Churchill, 1988)]. Different flow regimes are classified using Reynolds numbers (Kundu & Cohen, 2004).

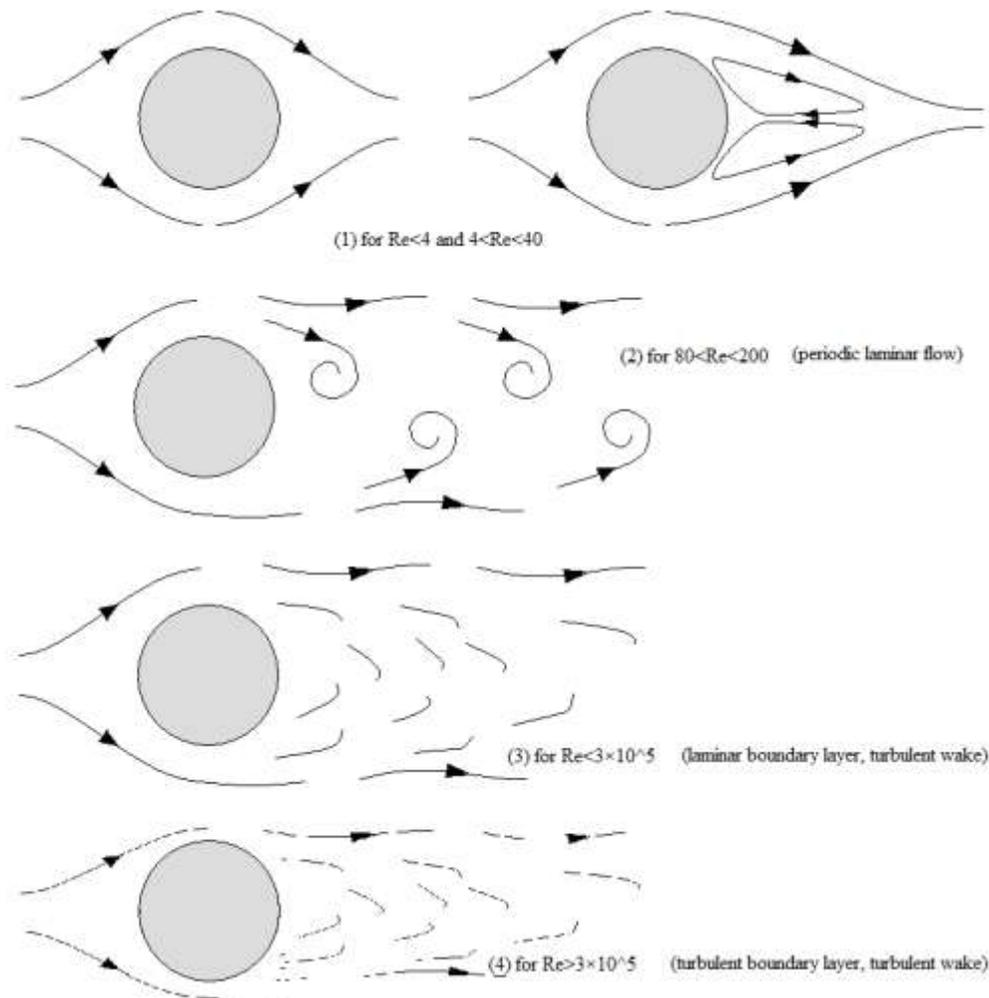


Figure 1.5 Regimen flows

For discussion here, Figure 10.17 from Kundu et al (Kundu & Cohen, 2004) is adapted in Figure 1.5. Four different flow regimes are represented in Figure 1.5: (1) for  $Re < 4$  and  $4 < Re < 40$  which

corresponds to the steady laminar flow in Figure 1.4, (2) for  $80 < Re < 200$  which corresponds to periodic laminar and also to periodic turbulent flow in Figure 1.4, and for turbulent flow there are two additional turbulent regimes of flow, (3) for  $Re < 3 \times 10^5$  and (4) for  $Re > 3 \times 10^5$ . Pier establishes the Reynolds number transition from laminar to periodic at a critical value  $49 < Re < 49.5$  (Pier, 2002).

A reverse flow appears in a region where the fluid flows backward behind the cylinder. In this 2D case, the flow has a negative horizontal (streamwise) velocity, which always occurs for Reynolds numbers below 200.

Figure 1.6 and Animation 1.1 show fluid moving backward behind a cylinder in a reverse region as a sequence of pictures. Flow goes from left to right and the cylinder is located at the left. In this case the Reynolds number is approximately 1000 but the flow is considerably affected by the no-slip condition not only on the cylinder, top and bottom walls, but also on the base wall perpendicular to the cylinder. The visualization was created with colored ink dropped upstream. Some ink fell down to the base of the channel where it remained static behind the cylinder.

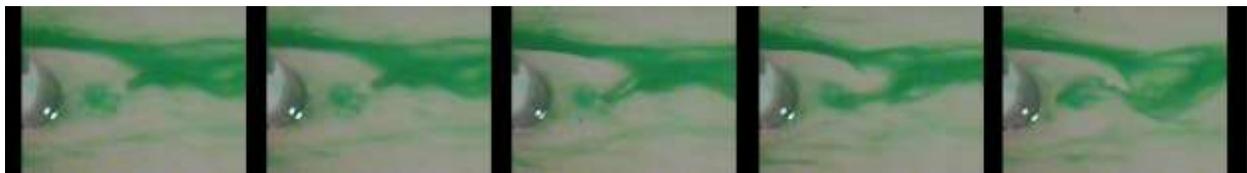
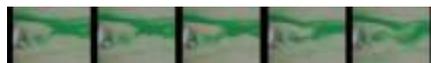


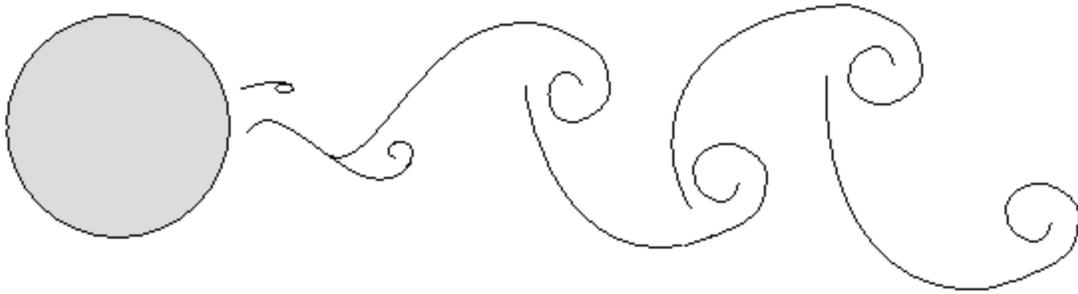
Figure 1.6 Sequence of pictures showing fluid moving backward in the reverse region (M. Ortega 2009)



Animation 1.1 [Fluid moving backward in the reverse flow region \(M. Ortega 2009\)](#)

The reverse region is also displayed as a green region in Figure 2.9.

Figure 1.7 shows vortices behind a cylinder for periodic laminar flow. The representation is based on observations and photographs, see Van Dyke's book for reference (Van Dyke, 1982). Clear clockwise vortices are located from the horizontal mid line to the top and the counterclockwise vortices are located from the mid line to the bottom.



**Figure 1.7 Vortex behind a cylinder**



**Figure 1.8 Vortices generated behind a cylinder which moves from right to the left (M. Ortega 2009)**



**Figure 1.9 Detail of vortices after a cylinder passed from right to the left (M. Ortega 2009)**

Flow moving around a fixed cylinder and a cylinder moving through initially static fluid in the opposite direction are similar problems. Figure 1.8 shows an instantaneous picture of the fluid behind a cylinder moving from right to the left. The fluid in this case is a mixture of water and

soap, original idea of Sofia Ortega (who is 10) to capture flow phenomena. An approximate makeup of the mixture is water 90% and soap 10%. The diameter of the cylinder is 5 mm (a small part of the cylinder is still in the picture) and the velocity is approximately 60 mm/s. The water viscosity is altered by viscosity of the soap. Viscosity is not measured. Result shown here are not quantitative and used here only for illustrative purposes.

Both Figures, 1.8 and 1.9, demonstrates how vortices form, which are clockwise at the top of the picture and counter clockwise at the bottom. This is valid when fluid moves from left to right around a cylinder or when the cylinder moves from right to left through the fluid. In either case, results are the same independent of the frame of reference, e.g. movement of the cylinder with respect to the fluid vs. movement of the fluid with respect to the cylinder.

#### 1.4. Frequency analysis

OpenFoam numerical results are referred to here as “data”. The objective of the frequency analysis in this section is to validate the “data” generated by the OpenFoam numerical simulation results.

The flow around a cylinder for the Reynolds numbers of interest in this study, exhibits periodic behavior. Periodic behavior is analyzed here in two ways: first using the dimensionless shedding frequency Strouhal number, and second using the Fast Fourier Transform (FFT) for a particular fixed point in the flow.

The “shedding frequency”  $f_S$  can be calculated using the dimensionless Strouhal number according to the following definition (Norberg, 1994):

$$St \equiv \frac{f_S D}{u_\infty} \quad (1.26)$$

The original formula was based on observations and published by Strouhal in 1878. Von Kármán studied the movement of the fluid behind the cylinder and its geometrical pattern in 1911. This pattern is still recognized as the Von Kármán’s street (Den Hartog, 1953) and it can be seen in Figure 1.10.



Figure 1.10 Vortices in the wake when the cylinder is moving to the left

The fast Fourier Transform is used to calculate the principal frequencies in time based data. A different frequency spectrum is obtained for each fixed point (see for example Figures 1.11 and 1.12). The first frequency after zero should match with the shedding frequency and the Strouhal number can be calculated applying equation 1.26. The shedding frequency is calculated in Section 2.7.1 and the frequency spectrum is calculated in Section 2.7.2.

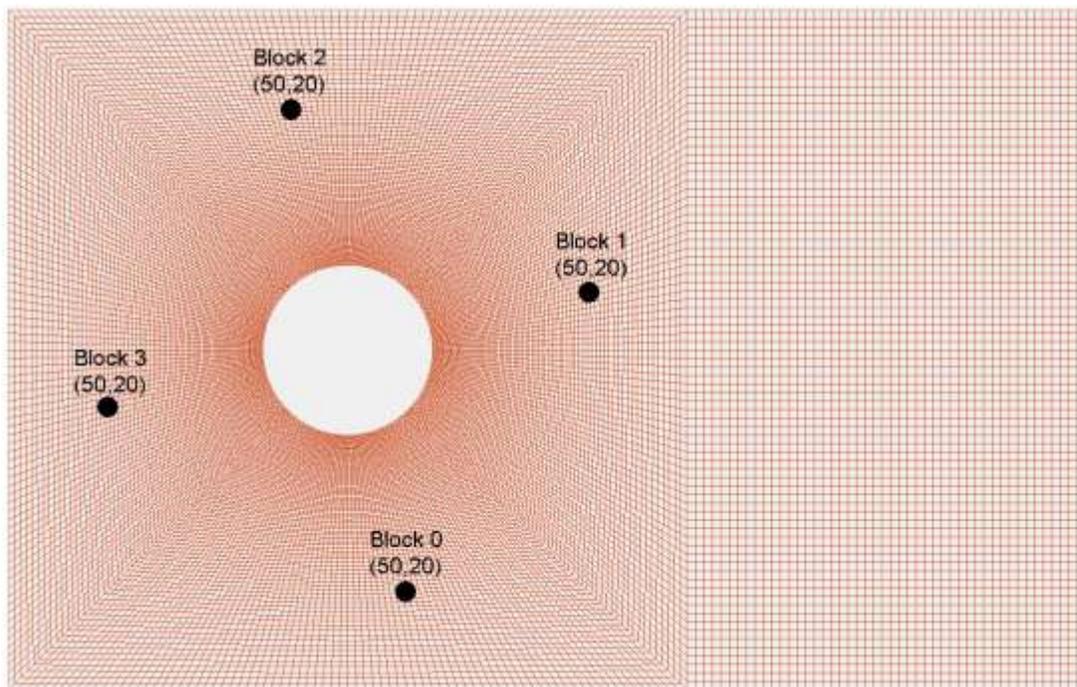


Figure 1.11 Points(i,j)=(50,20) in each of four blocks selected for history graph in Figure 1.12

The frequency spectrum is presented in amplitude (logarithmic scale) versus frequency graphs. A different graph can be drawn for different regions since the results for small subdomains in the numerical simulation are also different.

An example of how graphs vary from one point to another is shown in Figures 1.11 and 1.12 where the horizontal velocity is mapped in time at four points, each in a different block around the cylinder and using  $(i,j) = (50,20)$  for each block (see also Figures 2.2 and 2.3). Important contributions of the higher harmonics can be seen for locations behind the cylinder, where the periodic behavior is much more complicated than in other regions.

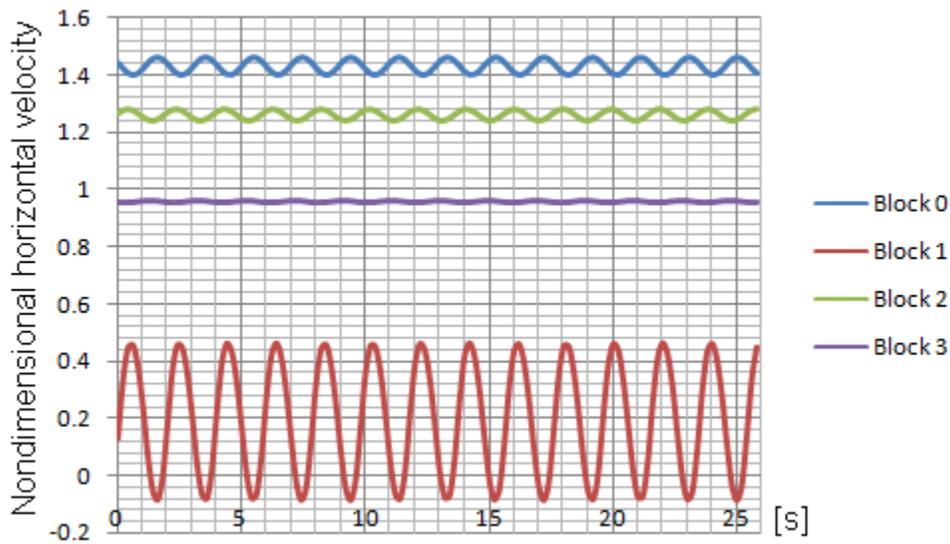


Figure 1.12 History graph of the dimensionless horizontal velocities for selected points in Figure 1.1

There are specific tools developed to do the analysis using the Fast Fourier Transforms (FFT). A FFT is used in this study to find the main frequencies that characterize the flow. A tool available in Excel (see section 3.7.2) was used here.

## 1.5. Chapter nomenclature

$\partial$  partial derivative

$\Delta$  discriminant

$\epsilon_{ijk}$  permutation symbol

$\lambda_2$  second largest eigenvalue of  $S^2 + \Omega^2$

$\mu$	dynamic viscosity
$\nu$	kinematic viscosity
$\rho$	density
$\omega_i$	vorticity
$\Omega$	antisymmetric part of the velocity gradient tensor, $\frac{du_i}{dx_j}$
$\sigma$	eigenvalues of the characteristic equation of the velocity gradient tensor, $\frac{du_i}{dx_j}$
$a$	side of the finite difference square cells
$C_D$	drag coefficient
$D$	total derivative
$E_{i,j}$	symmetric part of the velocity gradient tensor, $\frac{du_i}{dx_j}$
$F$	drag force
$f_i$	body force $i = 1,2,3$
$i$	cell in the $x$ direction or its left wall
$i$	index 1, 2 or 3
$II$	second invariant of the velocity gradient tensor, $\frac{du_i}{dx_j}$
$j$	cell in the $y$ direction or its bottom wall
$j$	index 1, 2 or 3
$N_k$	kinematic vorticity number
$P$	pressure
$P$	first invariant of the velocity gradient $P = 0$ (incompressible)

$Q$	second invariant of the velocity gradient, $\frac{du_i}{dx_j}$
$Q_F$	auxiliary variable used by Fromm
$R$	third invariant of the velocity gradient, $R = Det\left(\frac{du_i}{dx_j}\right)$
Re	Reynolds number
$S$	symmetric part of the velocity gradient tensor, $\frac{du_i}{dx_j}$
$u$	velocity in the $x$ direction at a point
$u_i$	velocities $i = 1,2,3$
$v$	velocity in the $y$ direction at a point
$x$	axis 1
$x_i$	axes $i = 1,2,3$
$y$	axis 2
$z$	thickness

## Chapter 2 . Numerical Simulation: OpenFoam

The study of the flow around a cylinder begins with the generation of simulation “data”. This is accomplished by defining the problem and then using OpenFoam to do the simulation using details such as a well defined grid, the initial velocity and pressure fields, slip and no slip at walls and fluid properties.

This chapter includes a preliminary evaluation of the simulation results to validate the “data”. Initially ParaView is used and after that a customized computer program VerFlow-V.01 was developed to analyze and interpret OpenFoam simulation results in section 2.6.

### 2.1. The problem

In order to solve the flow around a cylinder some assumptions are necessary, since mathematical expressions become simpler and the solutions are still close to real cases.

A 2D flow is considered in this thesis because it simplifies the problem, both mathematically and computationally. Although simulating 3D flow would be ideal, 3D Openfoam simulations would require multi-processor supercomputers. In this thesis 2D OpenFoam simulations were accomplished using high speed single processor desktop computers and simulation “data” was envisioned (visually analyzed) using laptop computers. Consequently the simulation results shown here are more likely to be reproduced by a larger community of researchers.

The temperature is assumed to be constant, which is appropriate since there are no thermal sources or sinks. The consequences of this assumption are that thermal dilatation or contraction can be neglected and the viscosity (kinematic and dynamic) can be considered as constants.

While gases are highly compressible, liquids behave drastically differently. Although density can change, when pressure and/or temperature changes, this change is very small and can be ignored. We expect the values of the pressure field to be comparable with the mean pressure in the whole region, so the changes in pressure are also very small and since the compressibility factor for liquids is very close to zero, density can be considered as constant. Although the variation in

pressure is small, it cannot be neglected. Pressure variations are balanced mechanically with the velocity field instantaneously.

The viscous stress equation for Newtonian fluids is a linear equation, which does well to represent the behavior of many fluids including water. We choose a laminar flow not steady, but periodic, that requires a range of Reynolds numbers defined in Figure 1.4. Since an infinite domain is not possible, the domain is confined with walls.

A summary of the main assumptions:

- i. Flow is 2D and laminar
- ii. The fluid is water which is considered incompressible and Newtonian
- iii. Flow is not temperature dependent
- iv. The velocity and dimensions yield a Reynolds number such that the flow is in the periodic laminar region (see figure 1.4)
- v. The slip condition is given for the side walls and the no-slip condition applies for the cylinder surface
- vi. Flow is not affected by the gravity field

Although some of the values in equation 1.24 depend on temperature, they can be considered constants when the variation in temperature is small. For water at ambient temperatures acceptable approximate values are:

$$\begin{aligned}\rho &= 1000 \left[ \frac{kg}{m^3} \right] \\ \mu &= 0.00089 \left[ \frac{kg}{m \cdot s} \right] \\ \nu &= 8.9 \times 10^{-7} \left[ \frac{m^2}{s} \right]\end{aligned} \tag{2.1}$$

According to figure 1.4, the periodic laminar region exists approximately for Reynolds numbers from 21 to 120. Selecting  $Re=70$  is a reasonable approximation for periodic laminar flow. When using equation 1.24 and properties given in equation 2.1, the following property relationship can be approximated as follows:

$$\begin{aligned}
u_{\infty} D &\approx \nu \text{ Re} \\
u_{\infty} D &\approx 8.9 \times 10^{-7} \left[ \frac{\text{m}^2}{\text{s}} \right] 70 \\
u_{\infty} D &\approx 6.23 \times 10^{-5} \left[ \frac{\text{m}^2}{\text{s}} \right]
\end{aligned}
\tag{2.2}$$

The data in Table 2.1 shows possible values for the velocity and the diameter using a Reynolds number of 70.

$\nu \text{ Re}$	$u_{\infty} [\text{m/s}]$	$D [\text{m}]$
6.23E-05	1	0.0000623
6.23E-05	0.1	0.000623
6.23E-05	0.01	0.00623
6.23E-05	0.001	0.0623
6.23E-05	0.0001	0.623

**Table 2.1 Velocity and diameter required for water and Re=70**

The combination gives either a very low velocity (fourth and fifth cases) or a very small diameter (first and second cases). Note, that for the third choice, the velocity and the diameter maintain a reasonable relationship that could be easily examined experimentally. Three velocities and their corresponding diameters are shown in Table 2.2. The table also shows the Reynolds number for each case.

$u_{\infty} [\text{m/s}]$	$D [\text{m}]$	$\nu \text{ Re}$	Re
0.01	0.005	5.00E-05	56.2
0.012	0.005	6.00E-05	67.4
0.014	0.005	7.00E-05	78.7

**Table 2.2 Velocities and diameter selected for simulation**

The dimensions of the channel are functions of the selected diameter in Figure 2.1. There is a special interest in the region behind the cylinder, so the dimensions are shown to scale.

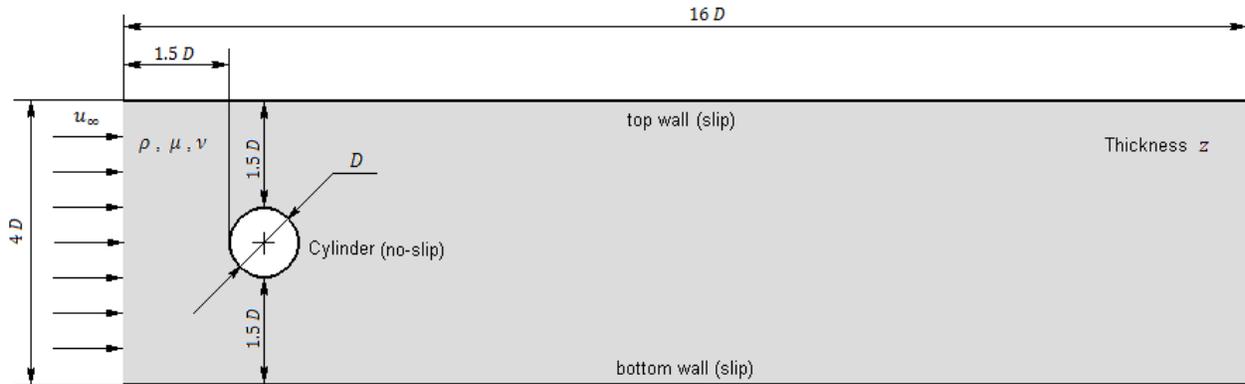


Figure 2.1 General dimensions

## 2.2. Boundary conditions

A uniform velocity is given at the inlet (left side) and the velocity field can change in the interior of the domain as the flow field develops in time.

The pressure at the outlet (right side) is set to zero as the boundary condition assuming that the fluid is going out to the atmospheric pressure (zero relative). Here the pressure is calculated per unit density. When the fluid is incompressible the density must be constant. Gravity is neglected.

The fluid is idealized to pass along the top and bottom walls; however, the mechanical friction is eliminated between the fluid and these walls by allowing the slip condition. So the top and bottom walls are not “realistic” in the sense that they only act as a frictionless channel for the flow.

The effect of the friction between the cylinder and the flow is established by assuming the no-slip condition.

## 2.3. The grid

To generate the grid, the domain is divided into five blocks as indicated in figure 2.2.

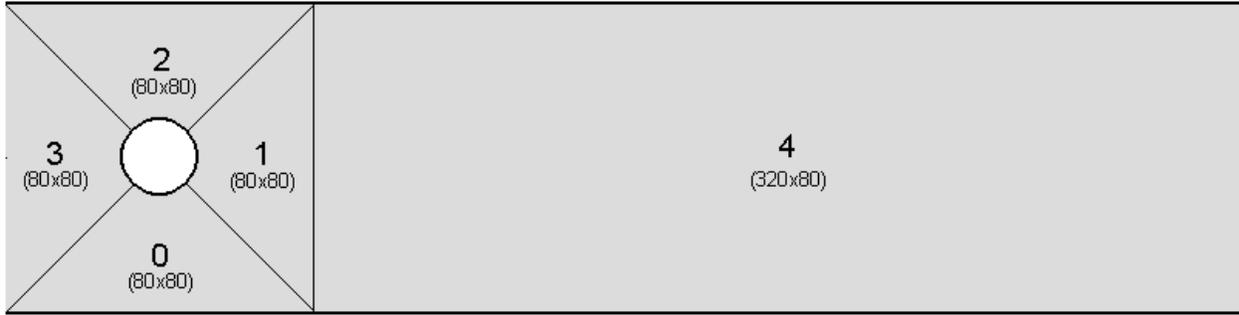


Figure 2.2 Blocks

This grid has some advantages in comparison to other choices. Although this is a simple grid, it has a very good resolution near the cylinder. For our purposes, the simplicity of the grid is an advantage, but the most relevant advantage of this grid is precisely its good resolution to solve features down to  $0.01D$  close to the cylinder,  $0.05D$  in the far region at the right in block 4. This is because the width of a cell in block 4 is  $0.05D$  while at the cylinder boundary it is even smaller than  $0.01D$ .

The blocks 0, 1, 2 and 3, are designed so that the resolution increases toward the center, not only in the  $x$  direction but also in the radial  $y$  direction ( $x$  and  $y$  are in this case the local non-orthogonal and curvilinear reference axis shown in Figure 2.3). The  $y$  axis represents just radial straight lines. The  $x$  axis transitions from being straight at the wall boundary at  $y = 0$  to curvilinear at the cylinder boundary.

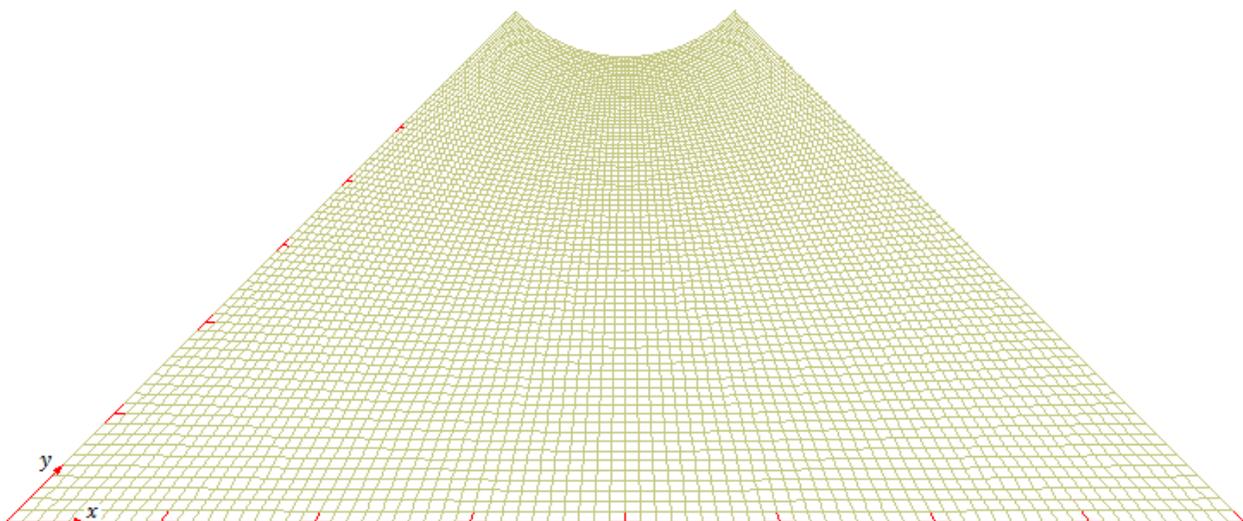


Figure 2.3 Blocks 0, 1, 2 and 3 (80x80 cells each) and local curvilinear reference axis

Figure 2.4 shows the detail of the first 30 rows (from a total of 80) closest to the center after blocks 0, 1, 2 and 3 are assembled.

The gradual change in grid size toward the center obeys a geometric relation. The ratio of the radial dimensions of the last cell at the cylinder boundary,  $j = 80$ , to the first cell, at  $j = 1$  (both over the same line), is set as  $f = 0.25$ . The problem is completely defined in the  $y$  direction. The block is divided along the  $x$  direction uniformly.

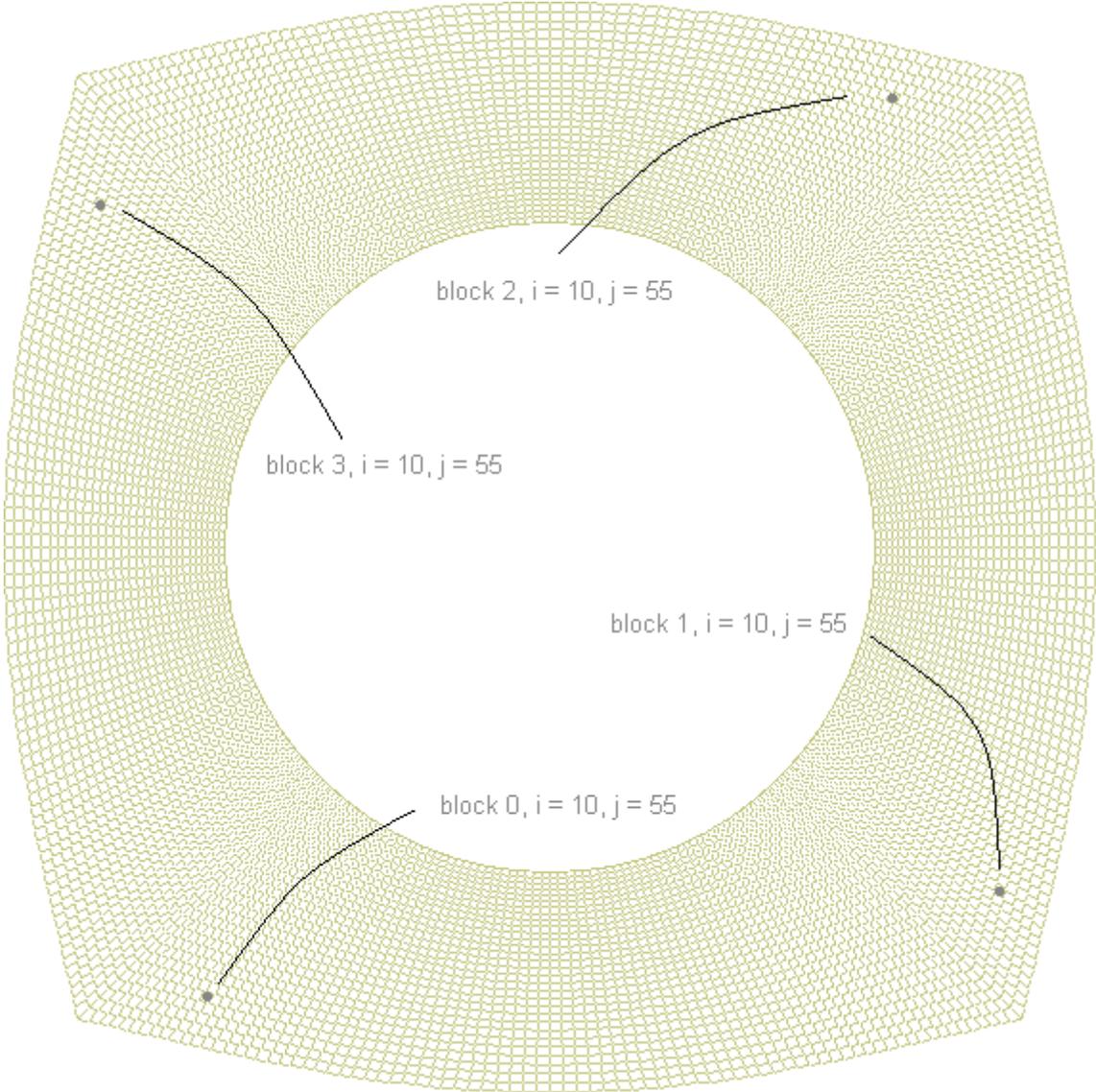


Figure 2.4 Point ( $i = 10, j = 55$ ) at each block and grid detail close to the cylinder (from  $j = 51$  to  $j = 80$  )

Now for convenience and clarity a new local rectangular  $(x, y)$  coordinate system is used for finite difference calculations, see Figure 2.5. For an arbitrary cell  $(i, j)$ , where  $1 \leq i \leq 80$  and  $1 \leq j \leq 80$ , the vertices  $(i - 1, j - 1)$ ,  $(i, j - 1)$ ,  $(i - 1, j)$  and  $(i, j)$  are considered. The local coordinates of these vertices are  $(x_{i-1}, y_{j-1})$ ,  $(x_i, y_{j-1})$ ,  $(x_{i-1}, y_j)$  and  $(x_i, y_j)$ . Let us label  $(x_e, 0)$  as the coordinates for the vertices  $(i - 1, 0)$  at the boundary; and  $(x_c, y_c)$  for the vertices  $(i - 1, 80)$  along the cylinder boundary.

$$c_1 = \sqrt{(2D - x_e)^2 + 4D^2} \quad (2.3)$$

$$c_2 = c_1 - D/2 \quad (2.4)$$

$$x_c = 2D - \left(\frac{D}{2}\right) \frac{2D - x_e}{c_1} \quad (2.5)$$

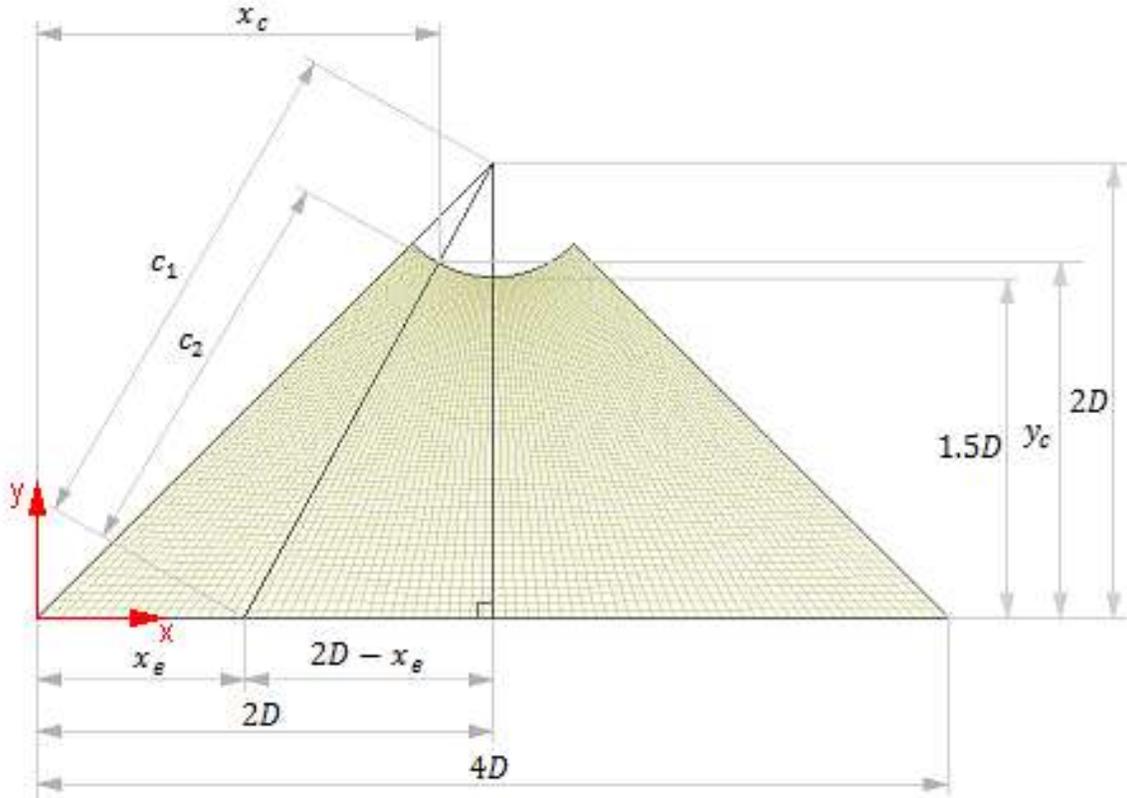


Figure 2.5 Basic geometry for blocks 0, 1, 2 and 3

$$y_c = 2D - \left(\frac{D}{2}\right) \frac{2D}{c_1} \quad (2.6)$$

The first length  $a_1$  can be calculated from the  $(i - 1, 0)$  vertice to the  $(i - 1, 1)$  vertice where the geometric progression is given by:

$$a_1 = \frac{c_2(1 - f^{1/79})}{1 - f^{80/79}} \quad (2.7)$$

The length  $a_j$  from the  $(i - 1, j - 1)$  vertice to the  $(i - 1, j)$  vertice can be evaluated as:

$$a_j = a_1 * f^{(j-1)/79} \quad (2.8)$$

As noted before the coordinates at the boundary are written:

$$\begin{aligned} x_0 &= x_e \\ y_0 &= 0 \end{aligned} \quad (2.9)$$

And for the other vertices:

$$x_j = x_{j-1} + a_j \frac{2D - x_e}{c_1} \quad (2.10)$$

$$y_j = y_{j-1} - a_j \frac{2D}{c_1} \quad (2.11)$$

These pairs  $(x_j, y_j)$  are converted to a unique coordinate system  $(x_u, y_u)$  which in this case coincides with the one used for block 0. Then block 0 needs no conversion but the others do according to table 2.3.

block	$x_u$	$y_u$
0	$x_j$	$y_j$
1	$4D - y_j$	$x_j$
2	$4D - x_j$	$4D - y_j$
3	$y_j$	$4D - x_j$
4	$x_u = 4D + x_j$	$y_j$

**Table 2.3 Conversion to a unique coordinate system**

The grid for block 4 (the fifth block) looks like that shown in Figure 2.6.



Figure 2.6 Block 4 (240x80 cells)

If a local reference axis for block 4 is located at the bottom-left corner (although the values are stored from the right to the left in OpenFoam), a horizontal translation is needed to reach the reference axis (valid for block 4 in Table 2.3).

## 2.4. Simulation time interval

The objective in this section is to define a reasonable value for the time interval used by the OpenFoam numerical simulation. Development of ideas presented in this section ends in the Courant number definition (see Equation 2.16) when estimating the simulation time interval.

The grid cells' size and shape have been defined, and using the inlet velocity  $u_{\infty}$ , as a reference value, the computational time interval for the OpenFoam simulation can be established.

Since velocity is dimensionally the distance divided by the time, it follows, the smallest cell size,  $L_{small}$ , divided by the inlet velocity,  $u_{\infty}$ , can be used to approximate the simulation time interval,  $t_i$ . The smaller cells exist at the cylinder boundary where their length was set at  $0.01D$ . Specifying a diameter of  $D = 0.5[cm]$  for example yields:

$$L_{small} \cong 0.005[cm] \tag{2.12}$$

To establish a reasonable estimate for the simulation time interval, the maximum velocity in the flow region must be estimated first. The flow has a uniform velocity at the inlet but when it passes around the cylinder, the flow is constricted and the mean velocity is increased by a factor. This factor is the ratio of the cross sectional lengths, so that the mean velocity in the constricted area is estimated by:

$$\bar{u}_{const.} \cong \frac{4}{3} u_{\infty} \quad (2.13)$$

For now a reference value for the time interval,  $t_{ic}$ , is estimated by considering flow across this constricted region:

$$t_{ic} = \frac{L_{small}}{\bar{u}_{const.}} \quad (2.14)$$

$$t_{ic} \cong \frac{0.005[cm]}{\frac{4}{3} u_{\infty}} \quad (2.15)$$

$u_{\infty}$ [cm/s]	$t_{ic}$ [s]
1	0.00375
1.2	0.003125
1.4	0.002679

**Table 2.4 Reference values for the maximum time interval allowed for flow in constricted region**

The Courant number, Co, establishes a limit on the simulation time interval,  $t_i$ , which must be smaller than  $t_{ic}$ :

$$Co = \frac{\bar{u}_{const.} t_{ic}}{L_{small}} \leq 1 \quad (2.16)$$

To satisfy all cases listed in Table 2.4, the simulation time interval,  $t_i$  (*deltaT* in OpenFoam), must be smaller than  $t_{ic}$ :

$$t_i = 0.001[s] \quad (2.17)$$

Finally, simulation results are stored as defined by the *writeInterval* property in OpenFoam that was set at  $200 \times t_i = 0.2$  [s]. Since storing results requires more than 10Mb after each time step, results are not saved after each computational time interval. The calculations are done precisely, every 0.001[s], but storing results is realized only at certain intervals, e.g. every 10 or 12 frames per computation cycle, so as not to exceed disk storage on typical desktop computers. Again it is noted that this requirement is established so that the results shown here can be reproduced using desktop computers.

## 2.5. Generating OpenFoam simulation “data”

Here the more important details are described when using OpenFoam running on the Linux Operating System (OS). For the grid defined in section 2.3, a high speed single processor desktop computer was used to generate the “data” and these results were graphically processed running Paraview or VerFlow-V.01 on a laptop computer using Windows OS.

As a first step, pre-processing tools must be used to modify or create the required OpenFoam files, because everything is stored separately (geometry, velocity, pressure, boundary conditions) in what are called *dictionaries*.

The tools referenced here are either FoamX or some editor program like WinVi. FoamX is a user friendly OpenFoam tool, used to manage the data from all of those dictionaries.

The *controlDict* dictionary includes information about the solver *application* (to be used in the simulation), the duration of the simulation *endTime*, the time interval *deltaT*, the interval of time for storing results *writeInterval*, and the data storage format *writeFormat*. The next table shows an example of those values:

Property	Value
<i>application</i>	icoFoam
<i>deltaT</i>	0.001
<i>writeInterval</i>	200
<i>writeFormat</i>	ascii

Table 2.5 Relevant information in the *controlDict* dictionary

The *transportProperties* dictionary stores the information of the kinematic viscosity  $\nu$ , which is set as: “ $\nu$  [0 2 -1 0 0 0 0] 8.9e-07;”. The numbers in the brackets are units of mass, length, time, etc. The viscosity is then  $8.9(10)^{-7} [m^2/s]$ .

The *blockMeshDict* dictionary is the file containing the geometry detail. A scale is used at the very beginning of this file as a *convertToMeters* property and in our case this value is 0.02. Following this information are the coordinates of the 20 *vertices* in three dimensions. Here only one cell in the  $z$  direction is used to solve the two dimensional problem. It follows the definition of the *blocks*, which includes the *vertices*, the number of divisions in each direction, and if the division is uniform or if it follows a geometric progression relation. To realize this requirement the size of the cells are decreased toward the center in the radial direction, here a factor of 0.25 is used. In order to get the curvature at the cylinder, the *edges* are defined by introducing the *vertices* numbers and one additional point along the *arc*. Finally the *patches* define the walls, the cylinder and the inlet and outlet surfaces.

The initial velocity and pressure fields are set in the  $U$  and  $p$  files.

In the *boundary* file put the *physicalType* for the *patches*. These are *wall* (no-slip condition), *inlet*, *outlet* and *slip* (slip condition).

The pre-processing part is done when the *blockmesh* tool is used in FoamX.

After the pre-processing part is completed, the simulation is executed by selecting the *Start Calculation* button in FoamX.

Finally, it is necessary to complete the calculations in a post-processing stage by running the  $Q$  and *Vorticity foamUtilities*. It is important to note that OpenFoam uses the variable  $p$  as the pressure per unit density.

Additional details are given in Appendix A: General Guide to Modify the Original OpenFoam Simulation.

## 2.6. Preliminary graphical results using ParaView

ParaView is a computer program that was designed to envision data resulting from an OpenFoam simulation. In order to see the results in a Microsoft Windows OS, it is necessary to install the appropriate version of ParaView and then proceed as follows. Copy the file *controlDict* in the same location (inside the *system* folder) but with the name *controlDict.foam*. Here only a portion of results are shown using ParaView.

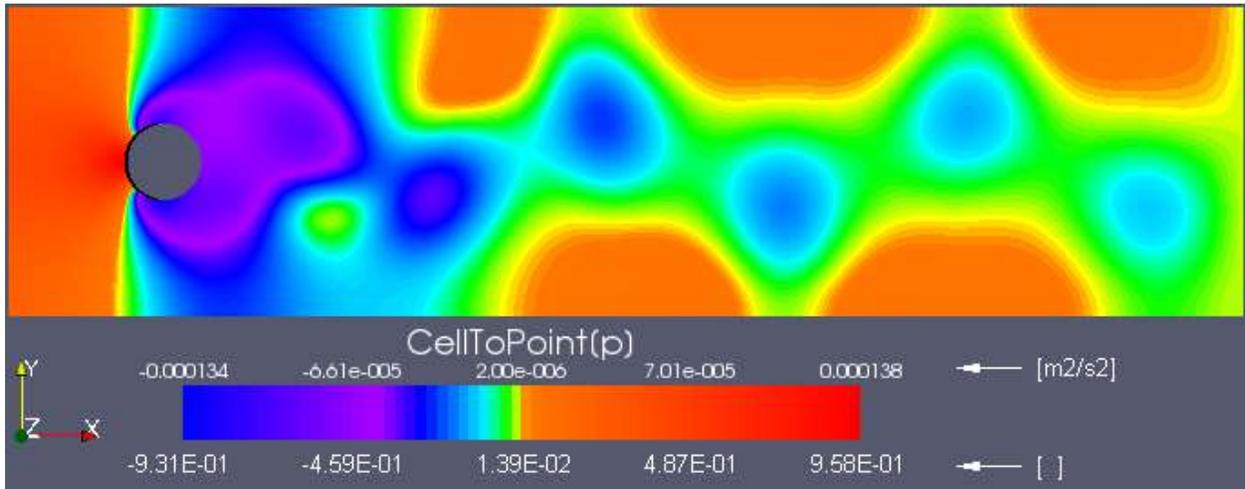


Figure 2.7 OpenFoam result for the pressure per unit density field

Figure 2.7 shows the pressure per unit density given in units of  $[m^2/s^2]$ . Dimensionless values  $[\ ]$  are added to the color legend bar considering the velocity at the inlet  $u_\infty = 0.012[m/s]$ . Dimensionless pressure is obtained by dividing the pressure per unit density by the square of the velocity at the inlet,  $p/u_\infty^2$ . Note that the pressure varies over a small range of values in the downstream region, where an alternate pattern of relative low pressure is localized at each side around the center line. This low pressure pattern is associated with vortices traveling downstream.

In Figure 2.8 the vorticity field  $w$  given in units of  $[1/s]$  for the same instant of time shown in Figure 2.7. Note that the vorticity values are close to zero at the center of the color legend. In general, when the vorticity is positive the flow is moving clockwise and when negative, counter-clockwise. In contrast to the general case, the vorticity does not necessarily directly identify vortices in the flow. An example of this is given at the cylinder boundary, where the vorticity

reaches higher values (in magnitude) although the flow is moving on each side of the cylinder in the same direction.

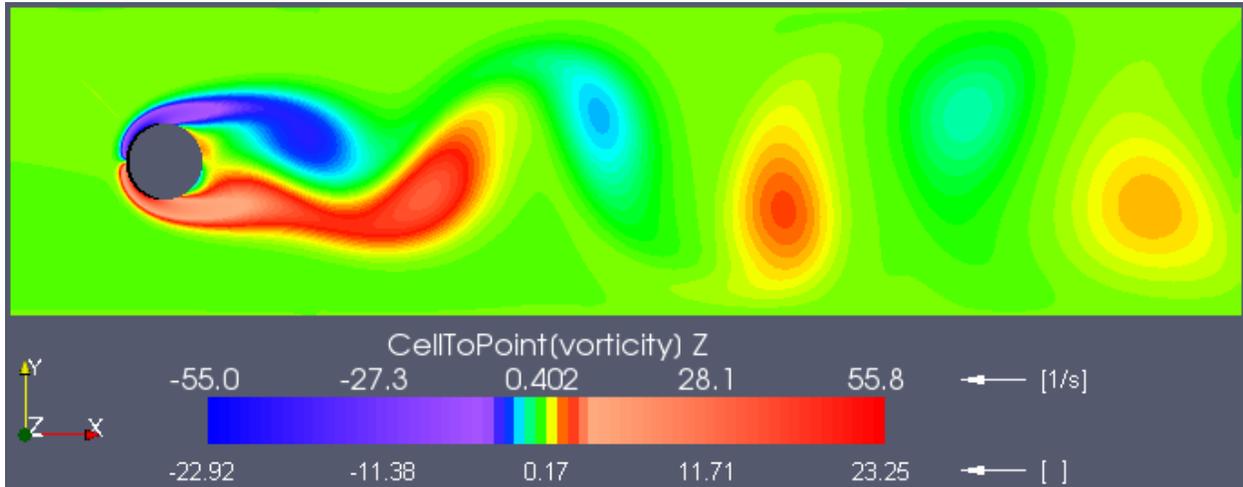


Figure 2.8 OpenFoam result for the vorticity field

A dimensionless vorticity has been added again to the graph in Figure 2.8, which is obtained when dividing the vorticity by the velocity at the inlet ( $u_\infty = 0.012[m/s]$ ) and multiplying this result by the diameter ( $D = 0.005[m]$ ),  $(w/u_\infty)D$ . The Reynolds number in this case is 67.4.

Both, Figures 2.7 and 2.8, capture important details of how the flow generates the alternating vortices in the downstream region and how their intensity decreases as the flow moves to the outlet.

Although ParaView is a powerful program that allows us to choose and change colors in the legend bar, zoom, camera angle, etc., our understanding is limited to features uniquely designed in this program. ParaView works for most general applications. However, the objective here is different, which is to calculate and envision the *qualitative* content of information for more than one variable simultaneously in space and time, and envision results using different mathematical models developed in Chapter 3. Imagination should not be limited to the use of conventional tools; here tools are created that solve our specific problem. This is the motivation for creating a customized graphical and analytical computer tool called VerFlow-V.01 which was used for exploring and envisioning flow around a cylinder in this thesis.

## 2.7. Frequency and time resolution optimization using VerFlow\_V.01

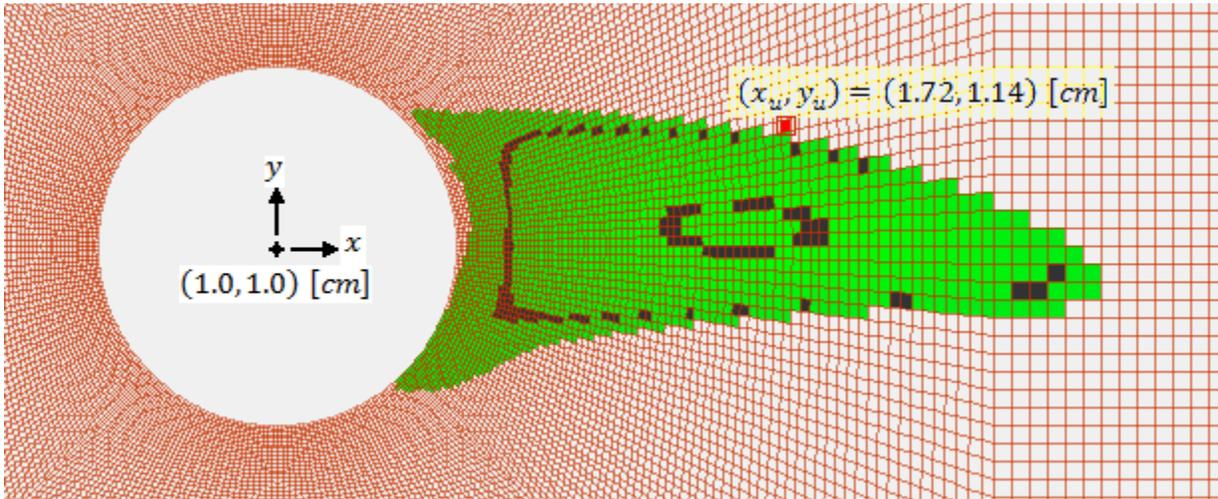


Figure 2.9 Selected point in the wake and negative  $u_x$

Here data generated by OpenFoam is used by VerFlow-V.01 to envision periodic behavior at a specific point, see Appendix C for details on how to use VerFlow-V.01. The velocity at the inlet for this case is  $u_\infty = 0.01$  [m/s] ( $Re=56.2$ ), assuming water is the fluid.

Figure 2.9 shows a reverse flow region in green, in which the velocities in the horizontal direction are negative (pointing to the left). In this region, and close to it, the velocity has relatively large variations in direction and magnitude. The point in block 1 is selected at  $(i, j) = (50, 20)$ . These coordinates are converted to  $(x_u, y_u) = (1.72, 1.14)$  [cm]. In other words, the point is located  $0.72$  [cm] to the right and  $0.14$  [cm] to the top from the center of the cylinder. The dimensionless values from the center are  $1.44 D$  to the right and  $0.28 D$  to the top, again see Appendix C for definitions.

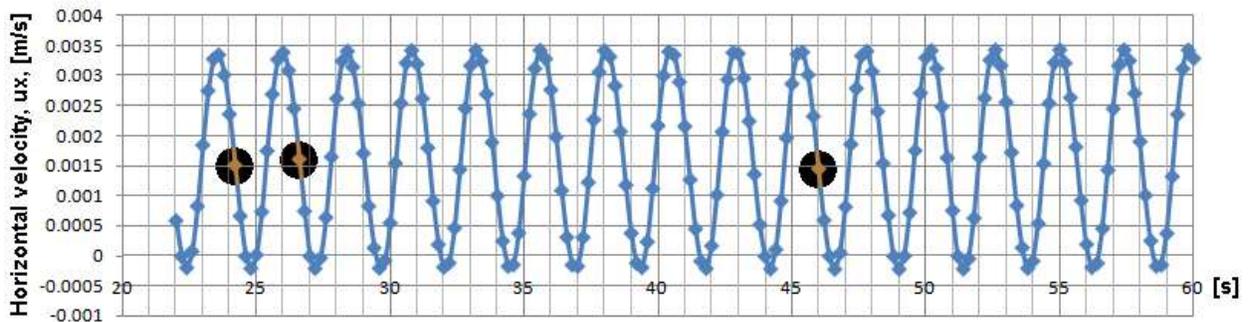


Figure 2.10  $u_x$  as a function of time for periodic behavior ( $u_\infty = 0.01$  [m/s])

At point  $(x_u, y_u)$  a periodic behavior for the velocity component,  $u_x$ , is observed in Figure 2.10, where the information is shown starting at time 22.0 [s] after which we observe the flow is periodic. This starting time is relative where 40 [s] for example can be replaced by 0[s] in a new simulation using OpenFoam.

### 2.7.1 Determining the shedding frequency

In this section, several models are developed for calculating the shedding frequency and period. There are two reasons for studying these different models. First, to establish a more precise value for the period and for the frequency and second, to improve time resolution, which in the next section, is used to understand complexities in the flow downstream of the cylinder. For example, how, when and exactly where complex convergence zones appear and what is their role in the development of periodic flow.

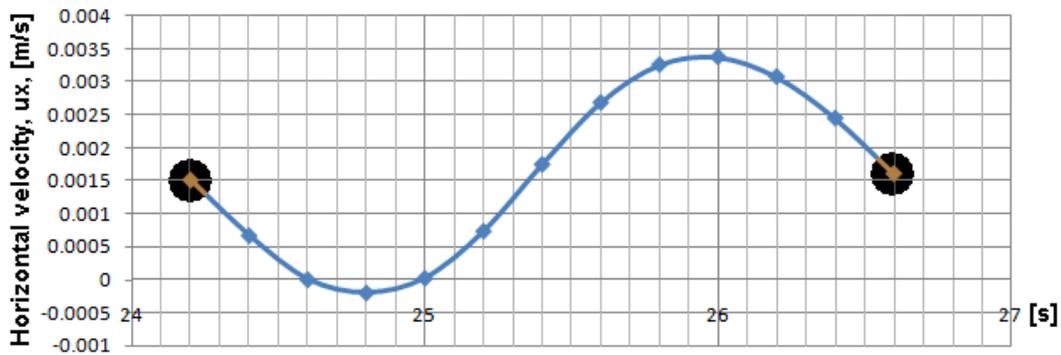


Figure 2.11 Primary selection for one cycle from time 24.2 [s] to time 26.6 [s]

The first approximation for one cycle is done by simple observation. The highlighted points at 24.2 [s] and 26.6 [s] in Figure 2.10 defines one complete cycle as shown in Figure 2.11.

Each point in Figure 2.11 is associated with a different “frame” in the representation of the flow.

According to this graph, twelve frames are enough to represent one cycle, because the thirteenth is the closest to the first one. The time for one cycle is the period, which in this example is labeled  $T_1$ :

$$T_1 = 26.6 [s] - 24.2 [s] = 2.4 [s] \quad (2.18)$$

where the frequency  $f_1$  is the reciprocal of the period  $T_1$ :

$$f_1 = \frac{1}{T_1} = \frac{1}{2.4 [s]} = 0.4167 [Hz] \quad (2.19)$$

The maximum error in determining  $T_1$  is the time interval between frames divided by two:  $\pm 0.1[s]$ . So the following expression is based in data from only one cycle:

$$\begin{aligned} 2.3[s] &\leq T \leq 2.5[s] \\ 0.4[Hz] &\leq f \leq 0.4348[Hz] \end{aligned} \quad (2.20)$$

Now taking advantage of the information over several cycles in Figure 2.10, the period and shedding frequency can be calculated more precisely. For one cycle the initial and final points on the vertical axis are not identical but similar, giving a small error. Taking more cycles, this error grows. A new point is considered which again matches well with the first one, but only after several cycles. That point is found after nine cycles. Figure 2.12 shows the selected nine cycles. The time step is 0.2 [s]. The first time shown in the graph is 24.2 [s] and the last time is 46.0 [s] which is also highlighted in Figure 2.10. This gives a new period labeled  $T_9$ :

$$T_9 = \frac{46.0 [s] - 24.2 [s]}{9} = 2.4222 [s] \quad (2.21)$$

$$f_9 = \frac{1}{T_9} = \frac{1}{2.42 [s]} = 0.4132 [Hz] \quad (2.22)$$

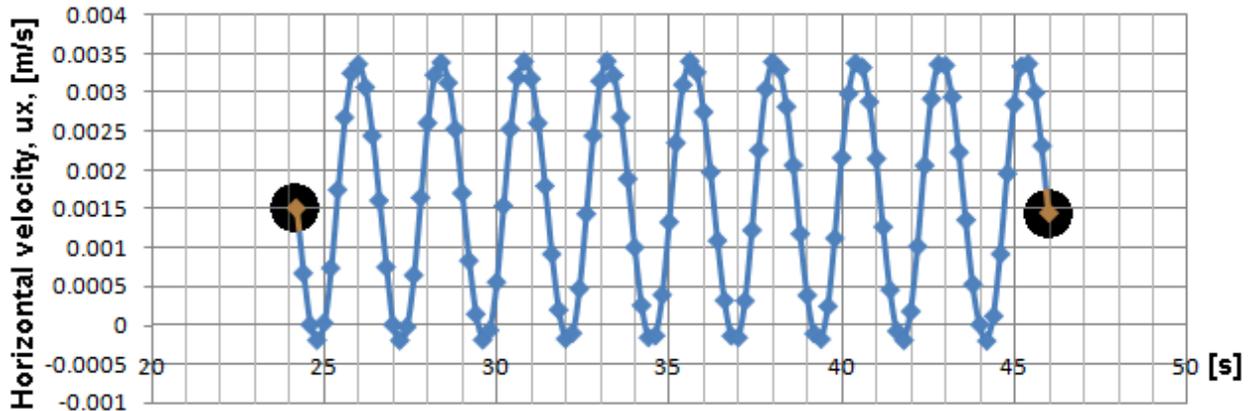


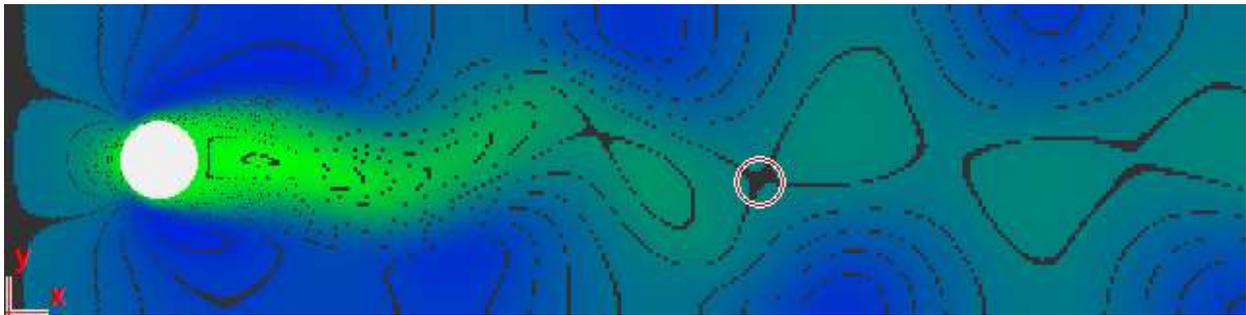
Figure 2.12  $u_x$  data for one cell in nine cycles and one hundred nine consecutive times

Now the error is the time interval divided by nine cycles and must be divided by two because the positive and negative variations are included in the error:  $\pm 0.0111[s]$ . The period and frequency are limited, using the nine cycles, as follows:

$$\begin{aligned} 2.4111[s] &\leq T \leq 2.4333[s] \\ 0.4110[Hz] &\leq f \leq 0.4147[Hz] \end{aligned} \tag{2.23}$$

It is clear that the frequency  $f_9$  and the period  $T_9$  are better results than  $f_1$  and  $T_1$ . Note that this information is obtained from only one cell among 44800 cells over the entire region, which includes blocks 0 through 4.

A more accurate evaluation of the nine cycles is required for all cells over the entire region. The horizontal velocity,  $u_x$ , at time 24.2 [s] is represented in the Figure 2.13.



**Figure 2.13** Instantaneous frame showing  $u_x$  for the time 24.2 [s], color legend Figure 2.14.

The color legend in Figure 2.14, applies to Figures 2.13, 2.15, 2.16, 2.17, 2.18 and 2.19. In these figures and according to the color legend, the higher velocities are represented in blue and the lower in green. Equally spaced black lines in the color legend, Figure 2.14, appear as contour lines whose line width appears to expand over a region in Figure 2.13. This expansion of the line width is highlighted as a red circle in Figure 2.13, which can also be used to envision gradients. This idea of envisioning gradients is included as a graphical model in VerFlow-V.01.

Note that the contour lines were moved to get one of the contour lines to coincide with the inlet velocity.

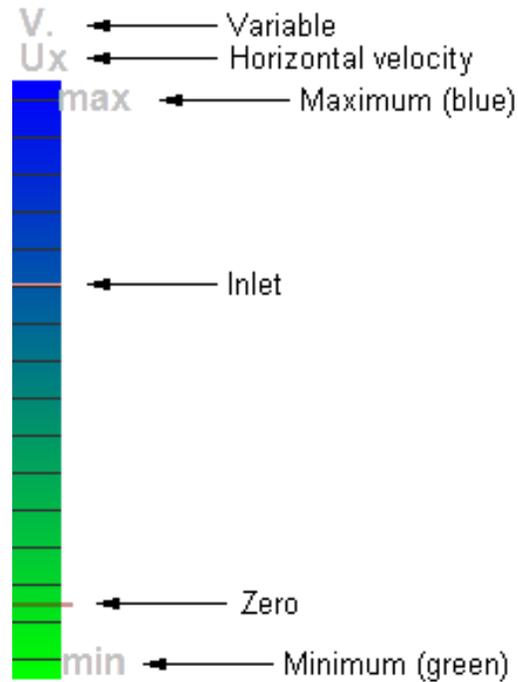


Figure 2.14 Legend for Figures 2.13, 2.15 – 2.19

Figures 2.15, 2.16 and 2.17 use contour lines in orange for times 45.8 [s], 46 [s] and 46.2 [s] which are compared with contour lines in black starting at 24.2 [s] in Figure 2.13.

This comparison allows us to see the accumulated error after nine cycles at time 45.8 [s] over the entire region in Figure 2.15.

The times 46.0 [s] (orange contours) and 24.2 [s] (black contours) match very well as seen in Figure 2.16. This confirms the validity of the value  $T_9$  for the period.

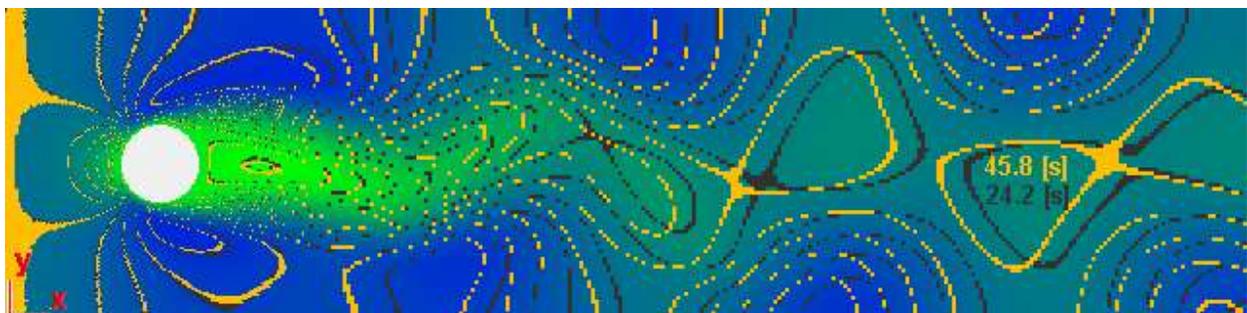


Figure 2.15  $u_x$  contour lines for 45.8 [s] in yellow compared over the entire region for 24.2 [s], color legend Figure 2.14

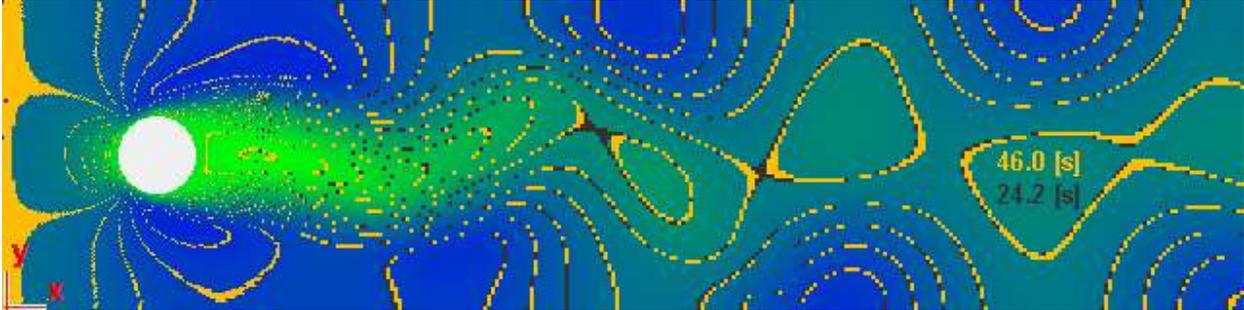


Figure 2.16  $u_x$  contour lines for 46.0 [s] in yellow compared over the entire region for 24.2 [s], color legend Figure 2.14

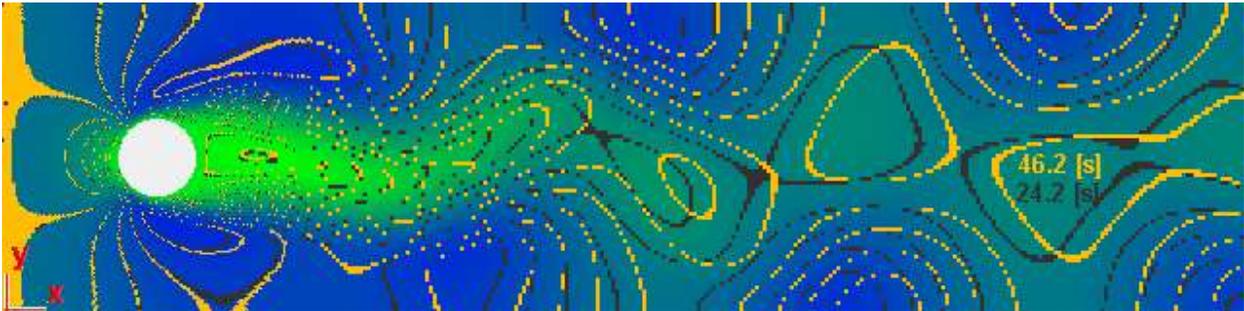


Figure 2.17  $u_x$  contour lines for 46.2 [s] in yellow compared over the entire region for 24.2 [s], color legend Figure 2.14

In Figure 2.17, the frame for time 46.2 [s] (orange contours) is again too different compared with that for time 24.2 [s] (black contours).

But why do nine cycles yield better result than eight or ten? To answer this question, it is necessary to compare the frames for the times 43.6 [s] and 48.4 [s] with 24.2 [s].

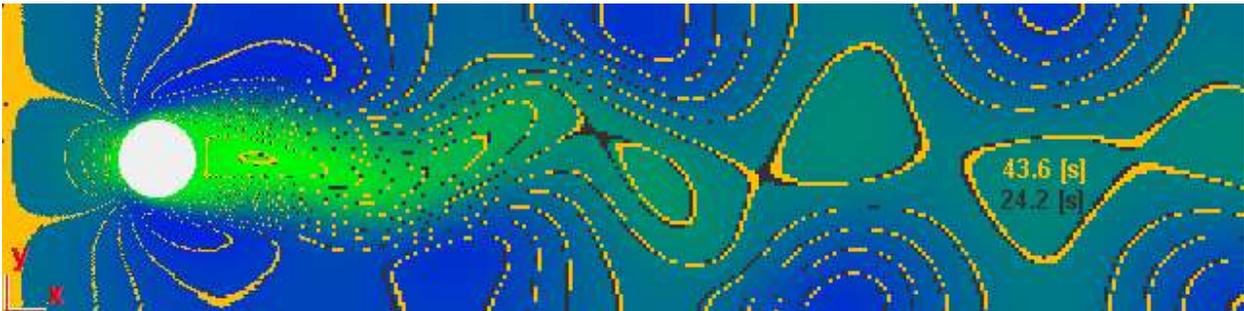


Figure 2.18  $u_x$  contour lines for 43.6 [s] in yellow compared over the entire region for 24.2 [s], color legend Figure 2.14

Although results are close for both time 43.6 [s] in Figure 2.18 and time 48.4 [s] in Figure 2.19, neither match as well as those shown in Figure 2.16. The orange contours, which are overlaid on

top of the black contours, appear to move slightly to the right in Figure 2.18 and to the left in Figure 2.19.

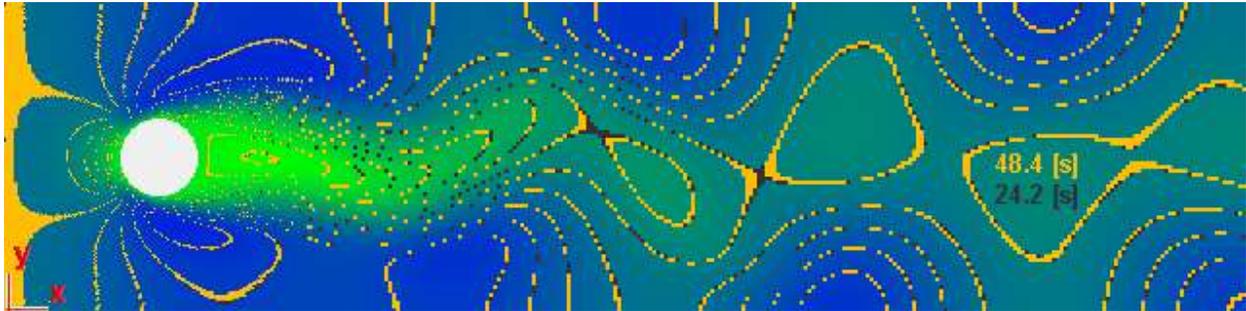


Figure 2.19  $u_x$  contour lines for 48.4 [s] in yellow compared over the entire region for 24.2 [s], color legend Figure 2.14

So, the selection of nine cycles is better than those of eight or ten cycles, and this result is realized not for one point but for all points over the entire region. For other cases, the pattern is repeated in a different number of cycles but the procedure is exactly the same. This discussion demonstrates why a graphical computer program needs to be customized within the scientific context as understood by the fluid dynamic researcher, e.g. “think outside the computer programmer’s box”.

### 2.7.2 Fourier Transform

Investigating periodicity for this problem continues where  $u_\infty = 1 [cm/s]$  and the data is evaluated from the same point that was highlighted in Figure 2.9. In order to identify the main frequencies involved, the Fourier Discrete Transform from Excel is used following the procedure detailed in this link:

<http://online.sfsu.edu/~larryk/Common%20Files/Excel.FFT.pdf>

The data is sampled every 0.2 [s] which means the sampling frequency is 5 [Hz]. The method is restricted to a number of samples  $N$ , such that  $N$  is a power of 2, e.g.  $2^7 = 128$ , so  $N$  is selected as 128. The *Fourier Analysis* tool in Excel is applied where the output data is 128 complex coefficients *FFTcomplex*.

The magnitude of these complex coefficients is divided by one half the number  $N$  and this is labeled *FFTmag*.

Divide 5 [Hz] by  $N$  to get the frequency interval of 0.0391 [Hz]. The frequency that corresponds to each sample varies linearly from 0 to 5 [Hz] minus the frequency interval labeled  $FFTfreq$ .

Finally the plot of the  $FFTmag$  as a function of the frequency  $FFTfreq$  is shown in Figure 2.20.

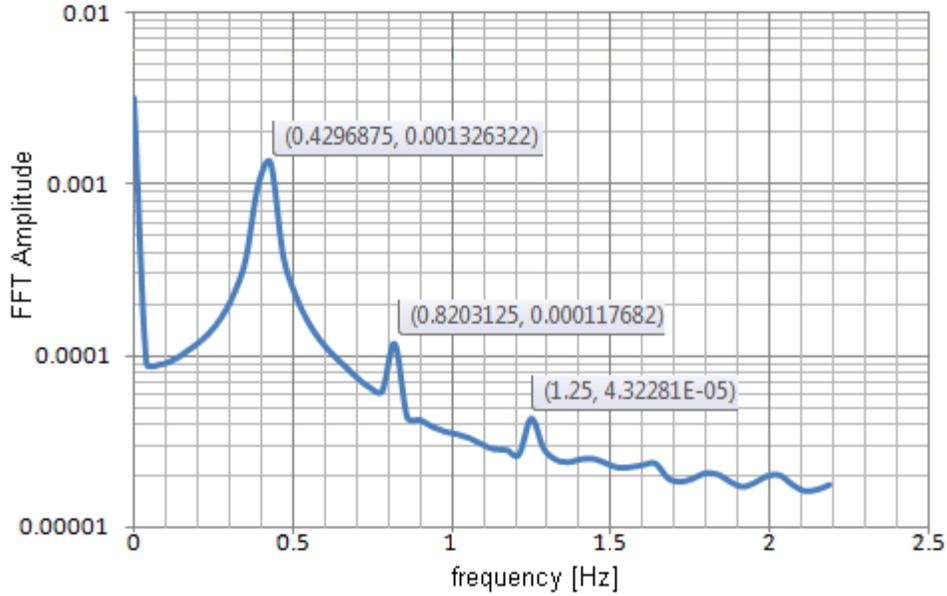


Figure 2.20 Magnitude of the Fourier Coefficients vs. frequency

The dominant frequency is:

$$f_{FFT} = 0.4297 [Hz] \quad (2.24)$$

$$T_{FFT} = \frac{1}{f_{FFT}} = 2.3273 [s] \quad (2.25)$$

A reasonable error in the prediction of the dominant frequencies using this method is equal to  $\pm 0.0195$  [Hz]. The main frequency and period based on the Fourier Transform are limited as follows:

$$\begin{aligned} 0.4102 [Hz] &\leq f \leq 0.4492 [Hz] \\ 2.2261 [s] &\leq T \leq 2.4381 [s] \end{aligned} \quad (2.26)$$

There are at least two clear secondary frequencies:

$$f_{FFT2} = 0.8203 \pm 0.0195 [Hz] \quad (2.27)$$

$$f_{FFT3} = 1.25 \pm 0.0195[Hz] \quad (2.28)$$

### 2.7.3 Comparative results of sections 2.7.1 and 2.7.2

Results in the previous sections are summarized in Table 2.6:

	Period [s]			Frequency [Hz]		
	Number of cycles		FFT	Number of cycles		FFT
	1	9		1	9	
Calc.	2.4	2.4222	2.3273	0.4167	0.4132	0.4297
min	2.3	2.4111	2.2261	0.4	0.4110	0.4102
max	2.5	2.4333	2.4381	0.4348	0.4147	0.4492
err	$\pm 0.1$	$\pm 0.0111$	(0.2120)	(0.0348)	(0.0037)	$\pm 0.0195$

Table 2.6 Period and main frequencies obtained from three different methods

These results, demonstrate that a better approximation of the main frequency from nine cycles is realized when the smallest of the maximum values and the biggest of the minimum are observed, both for period and frequency. Results indicate a small error. However, neither 1 cycle nor 9 cycles solutions reveal the secondary frequencies as the Fourier Transform does.

### 2.7.4 Strouhal number

The Strouhal number for this flow ( $Re = 56.2$ ) is calculated using Equation 1.21:

$$St \equiv \frac{f_s D}{u_\infty} = \frac{0.4132[Hz]0.005[m]}{0.01[\frac{m}{s}]} = 0.2066 \quad (2.29)$$

The result of the Strouhal number for the simulated flow with Reynolds number  $Re = 67.4$  (period 1.95[s] and velocity 0.012[m/s]) is  $St = 0.2137$ . These results are close to the constant value 0.22 (Den Hartog, 1953), where it is possible that the top and bottom walls affect the results when compared to unbounded flow over the cylinder. For these Reynolds numbers, experimental data (Churchill, 1988) gives smaller Strouhal numbers, around 0.15. Kundu et al points that 3D simulations show an asymptote of 0.21 which includes 3D instabilities while 2D experiments with soap film gives an asymptote of 0.2417 (Kundu & Cohen, 2004). The 2D

simulation instead of 3D and the constriction of the flow must generate this change as it also affects the drag (see Figure 4.1).

### 2.7.5 Improving the time resolution

Returning again to the same problem where  $u_{\infty} = 1[cm/s]$ , the data is sampled every  $0.2[s]$  in Figure 2.12. The idea is to build a single cycle but using the information over nine cycles. Combining data points from nine cycles into one is justified because the 109 points shown in Figure 2.12 were selected from 21,800 points that were calculated by the OpenFoam simulation every  $0.001[s]$  over the same nine cycles.

The number 109 is not divisible by 9 and, as a consequence, all those time points represent a unique (not repeated) location in the cycle. Because these points are unique, information can be reorganized in a coherent order.

Instead of having 109 frames in nine cycles with a uniform time interval of  $0.2[s]$  we will have 109 frames in one cycle with a uniform time interval given by:

$$0.2/9 [s] = 0.022 [s] \quad (2.30)$$

This is definitively an improvement in time resolution.

To specify the final correct order, the data is reordered as follows. The original data can be written as shown in Table 2.7. Each of the nine cycles are represented in each of the horizontal rows where the initial and end frames, 24.2 and 46.0 are included at the corners.

24.2	24.4	24.6	24.8	25.0	25.2	25.4	25.6	25.8	26.0	26.2	26.4	26.6	
	26.8	27.0	27.2	27.4	27.6	27.8	28.0	28.2	28.4	28.6	28.8	29.0	
	29.2	29.4	29.6	29.8	30.0	30.2	30.4	30.6	30.8	31.0	31.2	31.4	
	31.6	31.8	32.0	32.2	32.4	32.6	32.8	33.0	33.2	33.4	33.6	33.8	
	34.0	34.2	34.4	34.6	34.8	35.0	35.2	35.4	35.6	35.8	36.0	36.2	
	36.4	36.6	36.8	37.0	37.2	37.4	37.6	37.8	38.0	38.2	38.4	38.6	
	38.8	39.0	39.2	39.4	39.6	39.8	40.0	40.2	40.4	40.6	40.8	41.0	
	41.2	41.4	41.6	41.8	42.0	42.2	42.4	42.6	42.8	43.0	43.2	43.4	
	43.6	43.8	44.0	44.2	44.4	44.6	44.8	45.0	45.2	45.4	45.6	45.8	46.0

Table 2.7 Original nine cycles data

Close examination of data in Figure 2.12 suggests the proper order to follow. The time point following time point 24.2 must be 43.6 and after that 41.2, etc. as shown in Table 2.8.

								24.2
43.6	41.2	38.8	36.4	34.0	31.6	29.2	26.8	24.4
43.8	41.4	39.0	36.6	34.2	31.8	29.4	27.0	24.6
44.0	41.6	39.2	36.8	34.4	32.0	29.6	27.2	24.8
44.2	41.8	39.4	37.0	34.6	32.2	29.8	27.4	25.0
44.4	42.0	39.6	37.2	34.8	32.4	30.0	27.6	25.2
44.6	42.2	39.8	37.4	35.0	32.6	30.2	27.8	25.4
44.8	42.4	40.0	37.6	35.2	32.8	30.4	28.0	25.6
45.0	42.6	40.2	37.8	35.4	33.0	30.6	28.2	25.8
45.2	42.8	40.4	38.0	35.6	33.2	30.8	28.4	26.0
45.4	43.0	40.6	38.2	35.8	33.4	31.0	28.6	26.2
45.6	43.2	40.8	38.4	36.0	33.6	31.2	28.8	26.4
45.8	43.4	41.0	38.6	36.2	33.8	31.4	29.0	26.6
46.0								

**Table 2.8 Rotated data to reach a new order**

The desired order is accomplished by rotating the Table 2.7 clockwise ninety degrees as shown in Table 2.8. The desired reordering, although unexpected, is indeed accomplished by this geometric transformation.

								24.20
24.22	24.24	24.27	24.29	24.31	24.33	24.36	24.38	24.40
24.42	24.44	24.47	24.49	24.51	24.53	24.56	24.58	24.60
24.62	24.64	24.67	24.69	24.71	24.73	24.76	24.78	24.80
24.82	24.84	24.87	24.89	24.91	24.93	24.96	24.98	25.00
25.02	25.04	25.07	25.09	25.11	25.13	25.16	25.18	25.20
25.22	25.24	25.27	25.29	25.31	25.33	25.36	25.38	25.40
25.42	25.44	25.47	25.49	25.51	25.53	25.56	25.58	25.60
25.62	25.64	25.67	25.69	25.71	25.73	25.76	25.78	25.80
25.82	25.84	25.87	25.89	25.91	25.93	25.96	25.98	26.00
26.02	26.04	26.07	26.09	26.11	26.13	26.16	26.18	26.20
26.22	26.24	26.27	26.29	26.31	26.33	26.36	26.38	26.40
26.42	26.44	26.47	26.49	26.51	26.53	26.56	26.58	26.60
26.62								

**Table 2.9 New names for the 109 time points in one cycle.**

The proper order is now realized, however these points need to be assigned to new times. The first point is fixed and the others are reassigned as shown in Table 2.9.

The result of the new order is shown in Figure 2.21 as a smooth single cycle, which can be compared with the previous cycle in Figure 2.11, over the same time period. This higher resolution cycle is necessary when comparing different properties graphically over the same nine simulation cycles.

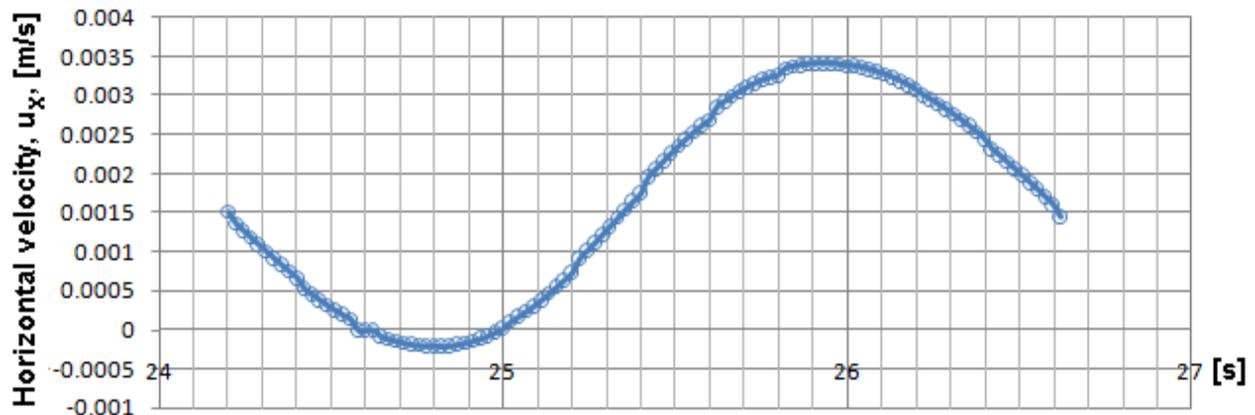


Figure 2.21 Increased resolution for one cycle

As expected, this method is much better than using linear interpolation on the original data. This approach is applied not only at one point, but over the entire region, so that the final result is smooth in time where the movement of a particle at each point can be envisioned more accurately both in a quantitative and *qualitative* sense as discussed in the introduction, Section 1.2.

## 2.8. Chapter nomenclature

- $\nu$       kinematic viscosity
- $a_1$      first term (length) in the geometric progression
- $a_j$      j-term (length) in the geometric progression
- $c_1$      geometric radius defined in Figure 2.5
- $c_2$       $c_1$  minus the cylinder radius

Calc.	calculated value
$D$	diameter of the cylinder
$D$	Courant number
err	error
$f$	factor for the geometric progression OR frequency
$f_1$	frequency based in one cycle information
$f_9$	frequency based in nine cycles information
$f_{FFT}$	main frequency based in the Fast Fourier Transform
$f_{FFT}$	second frequency based in the Fast Fourier Transform
$f_{FFT}$	third frequency based in the Fast Fourier Transform
$FFT$	Fast Fourier Transform
$f_s$	shedding frequency
$i$	relative position of a cell in the local curvilinear $x$ direction
$j$	relative position of a cell in the local curvilinear $y$ direction
$L_{small}$	reference characteristic size for smaller cells in the whole mesh
min	minimum value
max	maximum value
p	pressure per unit density field file (OpenFoam)
Q	second invariant of the velocity gradient field file and utility (OpenFoam)
Re	Reynolds number
St	Strouhal number

$T_1$	period based in one cycle information
$T_9$	period based in nine cycles information
$t_i$	time interval
$t_{ic}$	reference value for time interval associated with constricted region
$T_{FFT}$	period based in the Fast Fourier Transform
$U$	velocity field file (OpenFoam)
$u_\infty$	velocity at the inlet
$\bar{u}_{const}$	mean velocity at constricted region when the flow passes around the cylinder
$x_c$	local rectangular $x$ coordinate at internal boundary defined in Figure 2.5
$x_e$	local rectangular $x$ coordinate at external boundary defined in Figure 2.5
$x_j$	local rectangular $x$ coordinate of internal points for cell $j$
$x_{j-1}$	local rectangular $x$ coordinate of internal points for cell $j - 1$
$x_0$	local rectangular $x$ coordinate equal to $x_e$
$x_u$	horizontal unique coordinate
$y_c$	local rectangular $y$ coordinate defined in Figure 2.5
$y_{j-1}$	local rectangular $y$ coordinate of internal points for cell $j - 1$
$y_0$	local rectangular $y$ coordinate equal to 0
$y_u$	vertical unique coordinate

## Chapter 3 . Mathematical and Numerical Models

In this chapter mathematical expressions are derived that are used in Chapters two and four. One of the goals is to explore the second invariant of the velocity gradient and its relationship with the pressure field using numerical results from simulation “data” calculated by OpenFoam. The second invariant of the velocity gradient appears in the literature with different mathematical expressions, nomenclature and names. Three of these expressions are derived for the second invariant of the velocity gradient ensuring they are equivalent for an incompressible flow.

With 2D numerical simulation results, finite difference expressions are introduced that calculate pressure, the second invariant of the velocity gradient, drag and lift forces from simulation “data”. For comparison and use in the entire region, OpenFoam also calculates the second invariant. In Section 4.5 comparison of three different finite difference expressions ensure again that the second invariant of the velocity gradient from OpenFoam matches with these three expressions validating that they are indeed identical except by a factor of negative two.

The Navier Stokes equations establish the relation of the velocity field in space and time, while pressure depends on space. The dependence of pressure on time is not reflected directly in these equations. Hence, the pressure field is related to the velocity field over the entire region including boundary conditions at each instant of time. A common finite difference approach first, predicts how the velocity field changes in time, and second, determines the pressure field from the velocity field. For an incompressible flow, the pressure field is a function of the second invariant of the velocity gradient defined in this chapter. In this thesis, the second invariant of the velocity gradient is used to predict, not only the pressure at a point over the entire region, but also contributes to the drag and lift forces on the cylinder at any arbitrary point in this region. These contributions can now be mapped over the entire region. The development of the equations discussed above is the objective of this chapter.

### 3.1. Pressure independent of time

In a general 3D space, the Navier Stokes and the continuity equations can be written as (1.1) and (1.5). For a 2D problem with no external forces velocities, principal axes and body forces are:

$$u = u_1 \quad , \quad v = u_2 \quad (3.1)$$

$$x = x_1 \quad , \quad y = x_2 \quad (3.2)$$

$$f_1 = f_2 = 0 \quad (3.3)$$

The Navier-Stokes equations become:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial P}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (3.4)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial P}{\partial y} + \nu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (3.5)$$

And the continuity equation becomes:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (3.6)$$

In order to eliminate time, in the classic method Equation 3.4 is differentiated with respect to  $x$ , Equation 3.5 is differentiated with respect to  $y$  and the two resultant equations are added. This procedure is shown in Equations 3.7 to 3.15.

$$\frac{\partial}{\partial x} \left( \frac{\partial u}{\partial t} \right) + \frac{\partial}{\partial x} \left( u \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial x} \left( v \frac{\partial u}{\partial y} \right) = -\frac{\partial}{\partial x} \left( \frac{\partial P}{\partial x} \right) + \nu \frac{\partial}{\partial x} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (3.7)$$

$$\frac{\partial}{\partial y} \left( \frac{\partial v}{\partial t} \right) + \frac{\partial}{\partial y} \left( u \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left( v \frac{\partial v}{\partial y} \right) = -\frac{\partial}{\partial y} \left( \frac{\partial P}{\partial y} \right) + \nu \frac{\partial}{\partial y} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (3.8)$$

$$\begin{aligned}
& \frac{\partial}{\partial t} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial x} \left( u \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial x} \left( v \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \left( u \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left( v \frac{\partial v}{\partial y} \right) \\
&= -\frac{\partial}{\partial x} \left( \frac{\partial P}{\partial x} \right) + v \frac{\partial}{\partial x} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\partial}{\partial y} \left( \frac{\partial P}{\partial y} \right) \\
&+ v \frac{\partial}{\partial y} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)
\end{aligned} \tag{3.9}$$

The first term cancels by applying the continuity equation. Working with the remaining terms:

$$\begin{aligned}
& \frac{\partial}{\partial x} \left( u \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial x} \left( v \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \left( u \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left( v \frac{\partial v}{\partial y} \right) \\
&= -\frac{\partial}{\partial x} \left( \frac{\partial P}{\partial x} \right) + v \frac{\partial}{\partial x} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\partial}{\partial y} \left( \frac{\partial P}{\partial y} \right) + v \frac{\partial}{\partial y} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)
\end{aligned} \tag{3.10}$$

$$\begin{aligned}
& \frac{\partial u}{\partial x} \frac{\partial u}{\partial x} + u \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) + \frac{\partial v}{\partial x} \frac{\partial u}{\partial y} + v \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial y} \right) \\
&+ \frac{\partial u}{\partial y} \frac{\partial v}{\partial x} + u \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) + \frac{\partial v}{\partial y} \frac{\partial v}{\partial y} + v \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial y} \right) \\
&= -\left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right) + v \left[ \frac{\partial^3 u}{\partial x^3} + \frac{\partial}{\partial x} \left( \frac{\partial^2 u}{\partial y^2} \right) + \frac{\partial^3 v}{\partial y^3} + \frac{\partial}{\partial y} \left( \frac{\partial^2 v}{\partial x^2} \right) \right]
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
& \left( \frac{\partial u}{\partial x} \right)^2 + u \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) + 2 \frac{\partial v}{\partial x} \frac{\partial u}{\partial y} + v \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial y} \right) + u \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) + \left( \frac{\partial v}{\partial y} \right)^2 + v \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial y} \right) \\
&= -\left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right) + v \left[ \frac{\partial^3 u}{\partial x^3} + \frac{\partial}{\partial x} \left( \frac{\partial^2 u}{\partial y^2} \right) + \frac{\partial^3 v}{\partial y^3} + \frac{\partial}{\partial y} \left( \frac{\partial^2 v}{\partial x^2} \right) \right]
\end{aligned} \tag{3.12}$$

$$\begin{aligned}
& \left( \frac{\partial u}{\partial x} \right)^2 + u \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) + 2 \frac{\partial v}{\partial x} \frac{\partial u}{\partial y} + v \frac{\partial}{\partial y} \left( \frac{\partial u}{\partial x} \right) + u \frac{\partial}{\partial x} \left( \frac{\partial v}{\partial y} \right) + \left( \frac{\partial v}{\partial y} \right)^2 + v \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial y} \right) \\
&= -\left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right) + v \left[ \frac{\partial^2}{\partial x^2} \left( \frac{\partial u}{\partial x} \right) + \frac{\partial^2}{\partial y^2} \left( \frac{\partial u}{\partial x} \right) + \frac{\partial^2}{\partial y^2} \left( \frac{\partial v}{\partial y} \right) + \frac{\partial^2}{\partial x^2} \left( \frac{\partial v}{\partial y} \right) \right]
\end{aligned} \tag{3.13}$$

$$\begin{aligned}
& \left( \frac{\partial u}{\partial x} \right)^2 + u \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + 2 \frac{\partial v}{\partial x} \frac{\partial u}{\partial y} + v \frac{\partial}{\partial y} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \left( \frac{\partial v}{\partial y} \right)^2 \\
&= -\left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \right) + v \left[ \frac{\partial^2}{\partial x^2} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{\partial^2}{\partial y^2} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \right]
\end{aligned} \tag{3.14}$$

Using again the continuity equation and dropping the corresponding terms results in Equation 3.15, a first form of Poisson equation.

$$\left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + \left(\frac{\partial v}{\partial y}\right)^2 = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \quad (3.15)$$

Modification of Equation 3.15 by adding and subtracting an additional term in the left hand side gives:

$$\left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + \left(\frac{\partial v}{\partial y}\right)^2 + 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} - 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \quad (3.16)$$

By changing the order of the terms and factoring:

$$\left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \left(\frac{\partial v}{\partial y}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} - 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \quad (3.17)$$

$$\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} - 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \quad (3.18)$$

Once more using the continuity equation simplifies Equation 3.18 to Equation 3.19, a second form of Poisson equation.

$$2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} - 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \quad (3.19)$$

A third equivalent result can be obtained from the first result (3.15) by adding the square of the continuity equation to the left hand side and two additional zeros (functions of the continuity equation) to the right hand side:

$$\begin{aligned} \left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)^2 + 2u\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) + 2v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y} + \frac{\partial u}{\partial x}\right) \\ = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) + 0^2 + 2u\frac{\partial}{\partial x}(0) + 2v\frac{\partial}{\partial y}(0) \end{aligned} \quad (3.20)$$

$$\begin{aligned} \left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + \left(\frac{\partial v}{\partial y}\right)^2 + 2u\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right) + 2u\frac{\partial}{\partial x}\left(\frac{\partial v}{\partial y}\right) \\ + 2v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y}\right) + 2v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y}\right) = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) + 0 + 0 + 0 \end{aligned} \quad (3.21)$$

$$\begin{aligned}
2\left(\frac{\partial u}{\partial x}\right)^2 + 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + 2\left(\frac{\partial v}{\partial y}\right)^2 + 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + 2u\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right) + 2u\frac{\partial}{\partial x}\left(\frac{\partial v}{\partial y}\right) \\
+ 2v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y}\right) + 2v\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x}\right) = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right)
\end{aligned} \tag{3.22}$$

Reorganizing terms gives:

$$\begin{aligned}
\left[2\left(\frac{\partial u}{\partial x}\right)^2 + 2u\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right)\right] + \left[2\left(\frac{\partial v}{\partial y}\right)^2 + 2v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y}\right)\right] + \left[2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} + 2u\frac{\partial}{\partial x}\left(\frac{\partial v}{\partial y}\right)\right] \\
+ \left[2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + 2v\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x}\right)\right] = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right)
\end{aligned} \tag{3.23}$$

Solving inside each bracket reduces to the following expressions:

$$\begin{aligned}
\left[2\frac{\partial u}{\partial x}\frac{\partial u}{\partial x} + 2u\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right)\right] + \left[2\frac{\partial v}{\partial y}\frac{\partial v}{\partial y} + 2v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial y}\right)\right] + \left\{2\frac{\partial}{\partial x}\left[u\left(\frac{\partial v}{\partial y}\right)\right]\right\} \\
+ \left\{2\frac{\partial}{\partial y}\left[v\left(\frac{\partial u}{\partial x}\right)\right]\right\} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right)
\end{aligned} \tag{3.24}$$

$$\left[2\frac{\partial}{\partial x}\left(u\frac{\partial u}{\partial x}\right)\right] + \left[2\frac{\partial}{\partial y}\left(v\frac{\partial v}{\partial y}\right)\right] + 2\frac{\partial}{\partial x}\left(\frac{\partial uv}{\partial y}\right) = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \tag{3.25}$$

$$\left[\frac{\partial}{\partial x}\left(\frac{\partial u^2}{\partial x}\right)\right] + \left[\frac{\partial}{\partial y}\left(\frac{\partial v^2}{\partial y}\right)\right] + 2\frac{\partial^2 uv}{\partial x\partial y} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \tag{3.26}$$

The final result is given in 3.27, a third expression for the Poisson equation.

$$\frac{\partial^2 u^2}{\partial x^2} + \frac{\partial^2 v^2}{\partial y^2} + 2\frac{\partial^2 uv}{\partial x\partial y} = -\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}\right) \tag{3.27}$$

Combining Equations 1.6 and 1.7, used by Fromm (Fromm, 1963) yields an equivalent result to Equation 3.27. Note that Fromm used  $P$  as the pressure and  $P$  in Equation 3.27 is the pressure per unit density.

Equations 3.15, 3.19 and 3.27 are three equivalent expressions rewritten here as Equations 3.28, 3.29 and 3.30 switching the left and right hand sides. These three equations are used in the following section to demonstrate three equivalent forms of the second invariant of the velocity gradient.

$$\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} = -\left(\frac{\partial u}{\partial x}\right)^2 - 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} - \left(\frac{\partial v}{\partial y}\right)^2 \quad (3.28)$$

$$\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} = -2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + 2\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} \quad (3.29)$$

$$\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} = -\frac{\partial^2 u^2}{\partial x^2} - \frac{\partial^2 v^2}{\partial y^2} - 2\frac{\partial^2 uv}{\partial x\partial y} \quad (3.30)$$

### 3.2. Q: the second invariant of the velocity gradient

In this section, Equation 3.31 defines  $Q$  as the second invariant of the velocity gradient, which is the same as presented in Equation 1.15. Expansion of this equation and comparison to equation 3.28 are also presented to demonstrate that the right hand side of Equations 3.28, 3.29 and 3.30 are equal to the second invariant of the velocity gradient. Finite difference forms of the three expressions of  $Q$  are derived here and they are used in VerFlow-V.01 to compare with the second invariant of the velocity gradient obtained from OpenFoam. It was subsequently confirmed that these three expressions are in fact equivalent to the expression used by OpenFoam.

The definition of  $Q$  is given by:

$$Q = -\frac{1}{2}\frac{\partial u_k}{\partial x_i}\frac{\partial u_i}{\partial x_k} \quad (3.31)$$

#### 3.2.1 2D expressions for $Q$

Expanding equation (3.31) for the 2D problem and using equations (3.1), (3.2) and (3.3) we have:

$$Q = -\frac{1}{2}\left(\frac{\partial u}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial v}{\partial x}\frac{\partial u}{\partial y} + \frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial v}{\partial y}\frac{\partial v}{\partial y}\right) \quad (3.32)$$

$$2Q = -\left(\frac{\partial u}{\partial x}\right)^2 - 2\frac{\partial v}{\partial x}\frac{\partial u}{\partial y} - \left(\frac{\partial v}{\partial y}\right)^2 \quad (3.33)$$

The right hand side of this expression matches the right hand side of (3.28), therefore:

$$\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} = 2Q \quad (3.34)$$

Where  $Q$  can be evaluated from one of the following choices:

$$Q = -\frac{1}{2} \left( \frac{\partial u}{\partial x} \right)^2 - \frac{\partial v}{\partial x} \frac{\partial u}{\partial y} - \frac{1}{2} \left( \frac{\partial v}{\partial y} \right)^2 \quad (3.35)$$

$$Q = -\frac{\partial v}{\partial x} \frac{\partial u}{\partial y} + \frac{\partial u}{\partial x} \frac{\partial v}{\partial y} \quad (3.36)$$

$$Q = -\frac{1}{2} \frac{\partial^2 u^2}{\partial x^2} - \frac{1}{2} \frac{\partial^2 v^2}{\partial y^2} - \frac{\partial^2 uv}{\partial x \partial y} \quad (3.37)$$

From equations 3.34 and 3.35, a relationship between pressure and the velocity field is given by  $Q$ . The first and third terms of  $Q$  in equation 3.35,  $-\frac{1}{2} \left( \frac{\partial u}{\partial x} \right)^2$  and  $-\frac{1}{2} \left( \frac{\partial v}{\partial y} \right)^2$ , are either zero or negative. The factors in the second term in Equation 3.35,  $-\frac{\partial v}{\partial x} \frac{\partial u}{\partial y}$ , can be interpreted separately as follows. To get counter clockwise rotation, the factor  $\frac{\partial v}{\partial x}$  must be positive and the factor  $\frac{\partial u}{\partial y}$  must be negative at the same time. Similarly, to get clockwise rotation,  $\frac{\partial v}{\partial x}$  must be negative and  $\frac{\partial u}{\partial y}$  positive simultaneously. In both cases, the product  $-\frac{\partial v}{\partial x} \frac{\partial u}{\partial y}$  results positive. In other words, positive values of the second term of  $Q$  in Equation 3.35,  $-\frac{\partial v}{\partial x} \frac{\partial u}{\partial y}$ , imply rotation and *rotation implies positive values of  $-\frac{\partial v}{\partial x} \frac{\partial u}{\partial y}$* . Since the first and third terms in Equation 3.35 are also positive, *rotation implies positive values of  $Q$* . However positive values of  $Q$  not always imply rotation since there is not enough information about the second term in Equation 3.35. But what are the negative values of  $Q$ ? The negative values of  $Q$  can only come from the second term in Equation 3.35,  $-\frac{\partial v}{\partial x} \frac{\partial u}{\partial y}$ , which signifies that  $\frac{\partial v}{\partial x}$  and  $\frac{\partial u}{\partial y}$  are both either positive or negative and the fluid does not experience rotation but straining.

### 3.2.2 2D Finite difference form for $P$ as a function of $Q$

Equation 3.34 can be written in a central finite difference form around the cell  $(i, j)$  as Equation 3.38 considering a uniform grid with square cells of side  $a$ . Here,  $i$  and  $j$  are the integer numbers

associated with  $x$  and  $y$  to define the relative location of the cells (Note: In the upcoming equations, unless stated otherwise,  $i$  and  $j$  do not correspond to the indices used in indicial tensor notation).

$$\frac{P_{i-1,j} - 2P_{i,j} + P_{i+1,j}}{a^2} + \frac{P_{i,j-1} - 2P_{i,j} + P_{i,j+1}}{a^2} = 2Q_{i,j} \quad (3.38)$$

Equation 3.39 defines an expression for the pressure at a cell as  $P_{i,j}$ .

$$P_{i,j} = \frac{1}{4}(P_{i-1,j} + P_{i+1,j} + P_{i,j-1} + P_{i,j+1} - 2Q_{i,j}a^2) \quad (3.39)$$

The finite difference form shown in Equation 3.39 establishes the pressure per unit density at a cell,  $P_{i,j}$ , as the average of the neighboring pressures minus  $\frac{1}{4}(2Q_{i,j}a^2)$ . In Section 3.2.1, rotation was associated with positive  $Q$ . A clear consequence of this is that pressure decays from its average value where rotation is present. As the rotation becomes concentrated at a point, the velocities tend to zero and  $Q$  tends to its maximum value at that location. This explains how eddies are associated with zones of minimum pressure and positive  $Q$ .

Considering the relation between  $Q$  and  $Q_F$  given in Equation 1.19, Equation 3.39 matches with Equation 1.8 used by Fromm (Fromm, 1963). Here, eddies are associated with zones of minimum pressure and minimum  $Q_F$ .

### 3.2.3 2D Finite difference forms for $Q$

The expressions for the components of  $Q$  are generated in order to get the finite difference forms for  $Q$ . Once again a 2D uniform grid is used. The grid is formed by square cells, whose sides have a length equal to  $a$ . The finite difference form for Equations 3.40 to 3.43 is forward difference and for equations 3.44 to 3.46 is central difference:

$$\frac{\partial u}{\partial x} \rightarrow \frac{\Delta u}{\Delta x} = \frac{u_{i+1,j} - u_{i,j}}{a} \quad (3.40)$$

$$\frac{\partial u}{\partial y} \rightarrow \frac{\Delta u}{\Delta y} = \frac{u_{i,j+1} - u_{i,j}}{a} \quad (3.41)$$

$$\frac{\partial v}{\partial x} \rightarrow \frac{\Delta v}{\Delta x} = \frac{v_{i+1,j} - v_{i,j}}{a} \quad (3.42)$$

$$\frac{\partial v}{\partial y} \rightarrow \frac{\Delta v}{\Delta y} = \frac{v_{i,j+1} - v_{i,j}}{a} \quad (3.43)$$

$$\frac{\partial^2 u^2}{\partial x^2} \rightarrow \frac{\Delta(\Delta u^2)}{(\Delta x)^2} = \frac{u_{i-1,j}^2 - 2u_{i,j}^2 + u_{i+1,j}^2}{a^2} \quad (3.44)$$

$$\frac{\partial^2 v^2}{\partial y^2} \rightarrow \frac{\Delta(\Delta v^2)}{(\Delta y)^2} = \frac{v_{i,j-1}^2 - 2v_{i,j}^2 + v_{i,j+1}^2}{a^2} \quad (3.45)$$

$$\frac{\partial^2 uv}{\partial x \partial y} \rightarrow \frac{\Delta(\Delta uv)}{(\Delta x)(\Delta y)} = \frac{(uv)_{i+\frac{1}{2},j+\frac{1}{2}} - (uv)_{i-\frac{1}{2},j+\frac{1}{2}} - (uv)_{i+\frac{1}{2},j-\frac{1}{2}} + (uv)_{i-\frac{1}{2},j-\frac{1}{2}}}{a^2} \quad (3.46)$$

$Q$ , given by Equations 3.35, 3.36 and 3.37, can be written as  $Q_{i,j}$ , where  $i, j$  are the relative cell locations in the grid. Working on equation 3.35:

$$Q_{i,j} = -\frac{1}{2} \left( \frac{u_{i+1,j} - u_{i,j}}{a} \right)^2 - \left( \frac{v_{i+1,j} - v_{i,j}}{a} \right) \left( \frac{u_{i,j+1} - u_{i,j}}{a} \right) - \frac{1}{2} \left( \frac{v_{i,j+1} - v_{i,j}}{a} \right)^2 \quad (3.47)$$

$$Q_{i,j} = \frac{-1}{a^2} \left[ \frac{(u_{i+1,j} - u_{i,j})^2}{2} + (v_{i+1,j} - v_{i,j})(u_{i,j+1} - u_{i,j}) + \frac{(v_{i,j+1} - v_{i,j})^2}{2} \right] \quad (3.48)$$

Which yields Equation 3.49 which can be compared with Equation 3.35.

$$Q_{i,j} = \frac{-1}{a^2} \left( \frac{u_{i+1,j}^2}{2} - u_{i+1,j}u_{i,j} + \frac{u_{i,j}^2}{2} + u_{i,j+1}v_{i+1,j} - u_{i,j}v_{i+1,j} \right. \\ \left. - u_{i,j+1}v_{i,j} + u_{i,j}v_{i,j} + \frac{v_{i,j+1}^2}{2} - v_{i,j+1}v_{i,j} + \frac{v_{i,j}^2}{2} \right) \quad (3.49)$$

Similarly for Equation 3.36:

$$Q_{i,j} = - \left( \frac{v_{i+1,j} - v_{i,j}}{a} \right) \left( \frac{u_{i,j+1} - u_{i,j}}{a} \right) + \left( \frac{u_{i+1,j} - u_{i,j}}{a} \right) \left( \frac{v_{i,j+1} - v_{i,j}}{a} \right) \quad (3.50)$$

$$Q_{i,j} = \frac{-1}{a^2} (u_{i,j+1}v_{i+1,j} - u_{i,j+1}v_{i,j} - u_{i,j}v_{i+1,j} + u_{i,j}v_{i,j} - u_{i+1,j}v_{i,j+1} \\ + u_{i,j}v_{i,j+1} + u_{i+1,j}v_{i,j} - u_{i,j}v_{i,j}) \quad (3.51)$$

Further simplification yields Equation 3.52 which corresponds to Equation 3.36.

$$Q_{i,j} = \frac{-1}{a^2} (u_{i,j+1}v_{i+1,j} - u_{i,j+1}v_{i,j} - u_{i,j}v_{i+1,j} - u_{i+1,j}v_{i,j+1} + u_{i,j}v_{i,j+1} + u_{i+1,j}v_{i,j}) \quad (3.52)$$

Similarly for equation 3.37:

$$Q_{i,j} = -\frac{1}{2} \left( \frac{u_{i-1,j}^2 - 2u_{i,j}^2 + u_{i+1,j}^2}{a^2} \right) - \frac{1}{2} \left( \frac{v_{i,j-1}^2 - 2v_{i,j}^2 + v_{i,j+1}^2}{a^2} \right) - \left[ \frac{(uv)_{i+\frac{1}{2},j+\frac{1}{2}} - (uv)_{i-\frac{1}{2},j+\frac{1}{2}} - (uv)_{i+\frac{1}{2},j-\frac{1}{2}} + (uv)_{i-\frac{1}{2},j-\frac{1}{2}}}{a^2} \right] \quad (3.53)$$

Finally, the result that corresponds to Equation 3.37 is given by Equation 3.54:

$$Q_{i,j} = \frac{-1}{a^2} \left[ \frac{u_{i-1,j}^2}{2} - u_{i,j}^2 + \frac{u_{i+1,j}^2}{2} + \frac{v_{i,j-1}^2}{2} - v_{i,j}^2 + \frac{v_{i,j+1}^2}{2} + (uv)_{i+\frac{1}{2},j+\frac{1}{2}} - (uv)_{i-\frac{1}{2},j+\frac{1}{2}} - (uv)_{i+\frac{1}{2},j-\frac{1}{2}} + (uv)_{i-\frac{1}{2},j-\frac{1}{2}} \right] \quad (3.54)$$

Section 4.6 shows the results using Equations 3.49, 3.52 and 3.54 in block 4 (the fifth block in the mesh, see Figure 2.2.) for the second invariant of the velocity gradient in Figures 4.53, 4.54 and 4.55 respectively. Figure 4.52 is the second invariant from OpenFoam as a reference.

Note again that Equation 1.9, used by Fromm, and equation 3.54 are equivalent considering the relation given by Equation 1.19.

### 3.3. Forces acting on the cylinder

The purpose of this section is to derive mathematical expressions used to calculate drag and lift forces acting on the cylinder caused by pressure and viscous friction forces acting at the fluid cylinder interface. These equations are implemented in VerFlow-V.01. Results are presented in Section 4.1.

The location around the cylinder is defined by an angle,  $+\alpha$  counter clock wise positive, from the horizontal axis, see Figure 3.1.

### 3.3.1 Forces originating from pressure

Forces originated by the pressure at a cell are shown in Fig. 3.1. The shaded area represents a finite but small cell. The normal vector  $\bar{n}$  points outward from the cylinder. Considering the thickness  $z$  and the width,  $a$ , of the cell, the following equations describe the force and its orthogonal components:

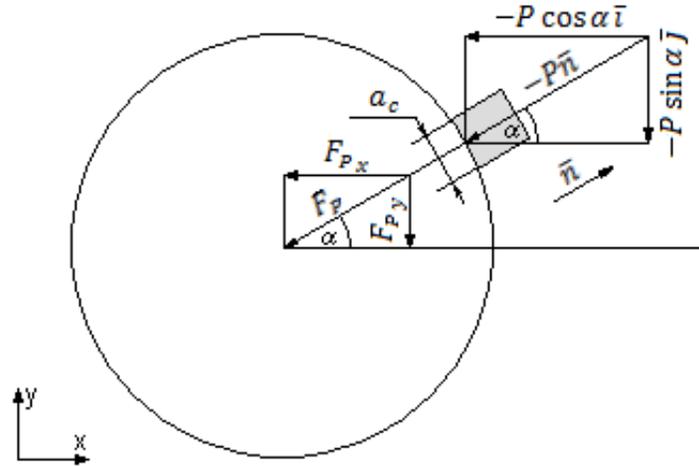


Figure 3.1 Forces originated by the pressure on the cylinder at a shaded cell

$$\bar{F}_{P_x} = -a_c z P \cos \alpha \bar{i} \quad (3.55)$$

$$\bar{F}_{P_y} = -a_c z P \sin \alpha \bar{j} \quad (3.56)$$

$$\bar{F}_P = -a_c z P \bar{n} \quad (3.57)$$

### 3.3.2 Forces originating from viscous effect

For arbitrary positive values of  $u$  and  $v$ , their individual contributions to the tangential counter clockwise velocity  $vel_{tg}$  are  $-u_{tg}$  and  $v_{tg}$ . This is represented in the following figure and corresponding formulas:

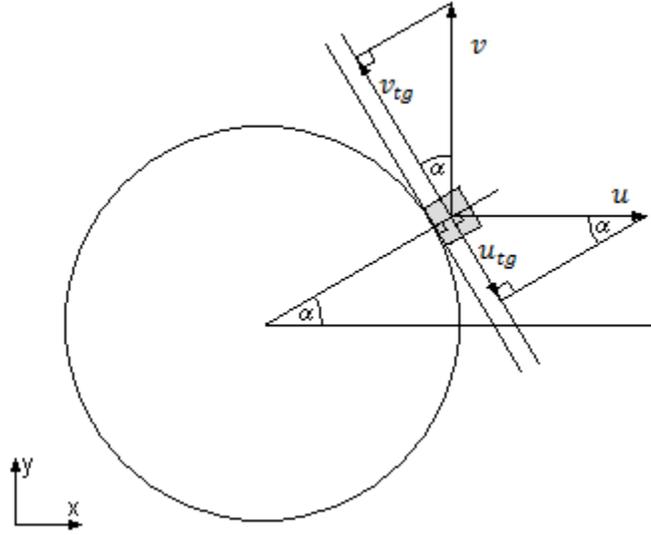


Figure 3.2 Individual contributions to the tangential velocity of arbitrary positive  $u$  and  $v$ .

$$u_{tg} = u \sin \alpha \quad (3.58)$$

$$v_{tg} = v \cos \alpha \quad (3.59)$$

$$vel_{tg} = -u_{tg} + v_{tg} \quad (3.60)$$

$$vel_{tg} = -u \sin \alpha + v \cos \alpha \quad (3.61)$$

Note that for the Equation 3.61,  $vel_{tg}$  is positive if it is counter clockwise.

$$\overline{vel}_{tg} = vel_{tg} \bar{\theta}_1 \quad (3.62)$$

Where  $\bar{\theta}_1$  is the tangential unit vector.

Now going back to the Cartesian reference, for an arbitrary counter clockwise tangential velocity, the individual components are obtained as vectors.

$$\bar{u} = -vel_{tg} \sin \alpha \bar{i} \quad (3.63)$$

$$\bar{v} = vel_{tg} \cos \alpha \bar{j} \quad (3.64)$$

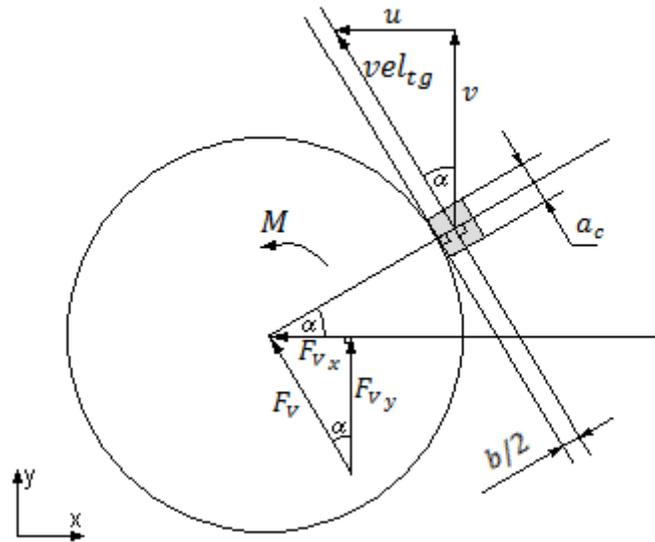


Figure 3.3 Viscous force and Velocity components for an arbitrary counter clockwise  $vel_{tg}$

Close to the cylinder, the fluid moves parallel to the surface. That is, the fluid moves parallel to a tangent line, when flow exists near the cylinder. The tangential velocity has to change from zero (no-slip condition at the cylinder) to a known value at the center of the cell.

The basic linear equation for the viscous force  $F_{fr}$  generated by the relative movement in the  $x$  axis of two parallel surfaces is given by Equation 3.65.

$$F_{fr} = \frac{\nu z \Delta x}{\Delta y} \Delta vel \quad (3.65)$$

where  $\Delta x$  is the distance parallel to the surfaces,  $\Delta y$  is the distance between them,  $\Delta vel$  is the relative velocity,  $z$  the thickness and  $\nu$  is the kinematic viscosity.

Equation 3.65 is applied to determine the viscous frictional force as a contribution for a given angle  $\alpha$ . That force acts as a tangent force just on the boundary. Moving the location of the force to the center of the cylinder requires adding a moment. The viscous force is proportional to the change in velocities ( $vel_{tg} - 0$ ), the length  $a_c$  parallel to the force, the kinematic viscosity  $\nu$  and the inverse of the perpendicular length  $b/2$ . This force has the same direction as the velocity  $\bar{vel}_{tg}$ .

The individual viscous force and the individual tangential velocity for a cell are parallel. So, the viscous force vector and its components can be evaluated as follows:

$$\bar{F}_V = \frac{vza_c}{b/2} \overline{vel}_{tg} \quad (3.66)$$

$$\bar{F}_{V_x} = -F_V \sin \alpha \bar{i} \quad (3.67)$$

$$\bar{F}_{V_y} = F_V \cos \alpha \bar{j} \quad (3.68)$$

Or a direct calculation can be done from  $\bar{u}$  and  $\bar{v}$ :

$$\bar{F}_{V_x} = \frac{vza_c}{b/2} \bar{u} \quad (3.69)$$

$$\bar{F}_{V_y} = \frac{vza_c}{b/2} \bar{v} \quad (3.70)$$

### 3.3.3 Total force

The total effect of pressure forces can be obtained by summing all the contributions around the cylinder.

$$\bar{F}_{P_{total}} = \sum_k \bar{F}_{P_k} \quad (3.71)$$

$$\bar{F}_{P_{total\ drag}} = \sum_k \bar{F}_{P_{x_k}} \quad (3.72)$$

$$\bar{F}_{P_{total\ lift}} = \sum_k \bar{F}_{P_{y_k}} \quad (3.73)$$

Again, the total drag and lift viscous forces can be obtained by summing all the contributions.

$$\bar{F}_{V_{total}} = \sum_k \bar{F}_{V_k} \quad (3.74)$$

$$\bar{F}_{V_{total\ drag}} = \sum_k \bar{F}_{V_{x_k}} \quad (3.75)$$

$$\bar{F}_{V_{total\ lift}} = \sum_k \bar{F}_{V_{y_k}} \quad (3.76)$$

The sum of both results gives the total forces.

$$\bar{F}_{total} = \bar{F}_{P_{total}} + \bar{F}_{V_{total}} \quad (3.77)$$

$$\bar{F}_{total\ drag} = \bar{F}_{P_{total\ drag}} + \bar{F}_{V_{total\ drag}} \quad (3.78)$$

$$\bar{F}_{total\ lift} = \bar{F}_{P_{total\ lift}} + \bar{F}_{V_{total\ lift}} \quad (3.79)$$

Note that although the drag and lift forces are written in vector form, they are just pure horizontal and vertical forces.

### **3.3.4 Moment on the cylinder**

The pressure forces are always pointing to the center, so these forces will not require an external balancing moment, but that is not what occurs with the viscous forces. Although the final result shows a relative small requirement, the moment acting on the cylinder due to viscous forces can be evaluated as follows:

$$\bar{M} = \bar{r} \times \bar{F}_V \quad (3.80)$$

$$\bar{M}_{total} = \sum_k (\bar{r}_k \times \bar{F}_{V_k}) \quad (3.81)$$

Alternate equations can be realized when considering that the radius is constant and always perpendicular to the tangent force (easily understood in the cylindrical coordinate system).

$$\bar{M} = r \bar{F}_V \quad (3.82)$$

$$\bar{M}_{total} = r \sum_k \bar{F}_{V_k} \quad (3.83)$$

The deviation from the center for the forces was found to be very small and was ignored.

### 3.4. Alternative equation for the pressure at a point $P$

This section introduces Equation 3.84, which is the solution to Poisson equation (Equation 1.7) in 2D. The derivation of this equation, given in Appendix D, was contributed by Clinton Dancey. Equation 3.84 is used in this thesis to predict the drag and lift coefficients and the pressure at any point in the flow field domain. Components of these predicted results are calculated using VerFlow-V.01. This Section also includes the finite difference forms for two particular cases: (1) within a rectangular subdomain, or (2) anywhere in the entire domain, which is implemented in VerFlow-V.01 where the user can interactively select and explore different points and subdomains. Results are discussed in Sections 4.2, 4.3 and 4.4.

Warsi presents an expression for the pressure at a point in volumetric space as a function of the velocity field, and the pressure and pressure gradient acting along surface boundaries enclosing that volumetric space, see equations 6.183 and 6.184 in his book (Warsi, 1993).

The two dimensional version of Warsi's three dimensional expression can be written as equation 3.84 (see Appendix D for the derivation).

$$P_P = -\frac{1}{\pi} \iint_A Q_Q \ln\left(\frac{1}{r}\right) dA + \frac{1}{2\pi} \oint_L \ln\left(\frac{1}{r}\right) \frac{\partial P}{\partial n} dL - \frac{1}{2\pi} \oint_L P \frac{\partial \ln\left(\frac{1}{r}\right)}{\partial n} dL \quad (3.84)$$

Where  $P_P$  is the pressure per unit density at point  $P$ , the first term is a surface integral of all  $Q$  points on area  $A$  and the other terms are contour integrals enclosing  $A$  on the boundaries, where  $\bar{n}$  is the outward normal along the boundary  $L$ . Figure 3.4 shows these definitions for (a) an arbitrary domain, (b) a rectangular domain, and (c) the entire domain.

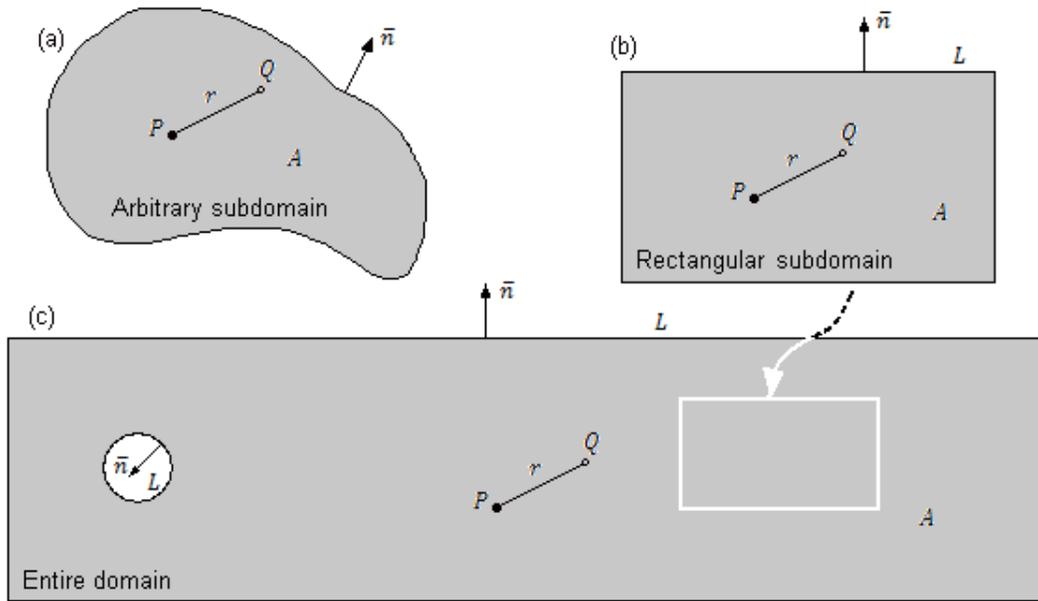


Figure 3.4 Two dimensional domains defined (a) Arbitrary subdomain, (b) rectangular subdomain and (c) entire domain.

Each term in Equation 3.84 is labeled, as shown below, and used in the discussion that follows in sections 3.4.1 and 3.4.2.

$$(Q) \equiv -\frac{1}{\pi} \iint_A Q_Q \ln\left(\frac{1}{r}\right) dA \quad (3.84a)$$

$$(CI_1) \equiv \frac{1}{2\pi} \oint_L \ln\left(\frac{1}{r}\right) \frac{\partial P}{\partial n} dL \quad (3.84b)$$

$$(CI_2) \equiv -\frac{1}{2\pi} \oint_L P \frac{\partial \ln\left(\frac{1}{r}\right)}{\partial n} dL \quad (3.84c)$$

These same terms are calculated and displayed in VerFlow-V.01 for comparison both quantitatively and *qualitatively* as an image such that the researcher can interpret the contribution of each term. This interpretation extends to the dynamic case where animations are included in the discussion of results in section 4.0.

### 3.4.1 Finite difference forms for a rectangular domain

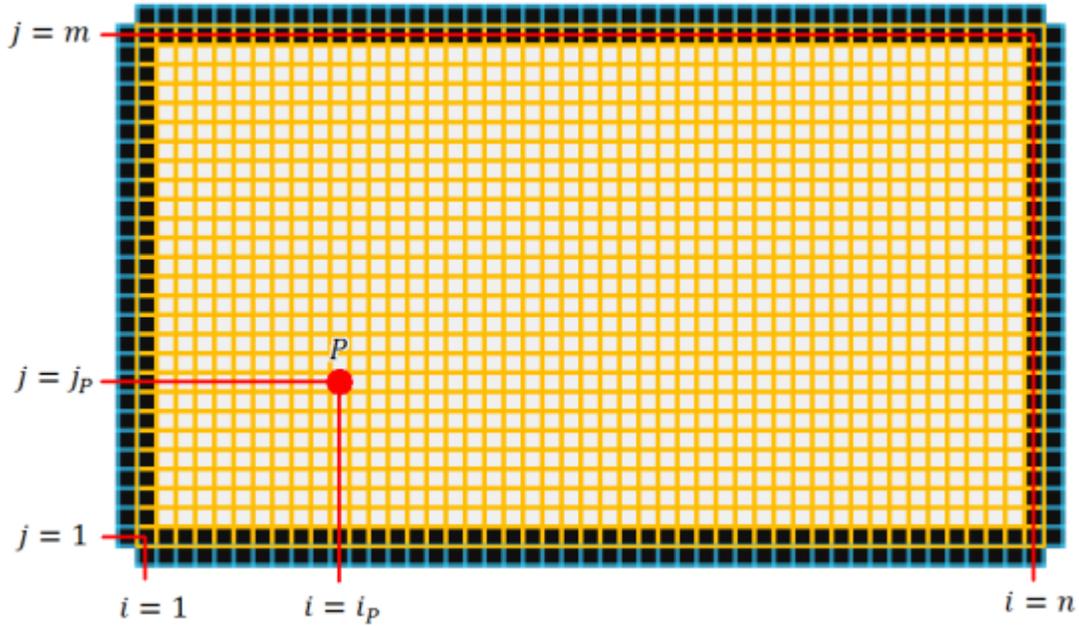


Figure 3.5 Arbitrary point in a rectangular subdomain inside block 4.

Figure 3.5 shows a rectangular subdomain, whose cells have orange boundaries. This subdomain is inside block 4, see Figure 2.2. All cells in the grid are squares with sides of length  $a$ . A local reference for the cells in the rectangular domain is defined from the bottom left corner where the first cell is  $(i, j) = (1, 1)$ . The last cell,  $(n, m)$ , is located at the top right corner. The second invariant of the velocity gradient  $Q_{(i,j)}$  is known at each cell  $(i, j)$  with orange boundaries and the pressure  $P$  is known at every cell with black background. Point  $P$  (red) is selected inside the domain where the pressure  $P_P$  is calculated.

The finite difference form of the surface integral, labeled as  $(Q)$ , in Equation 3.84a is given in Equation 3.85.

$$(Q) = -\frac{1}{\pi} \sum_{i=1}^n \sum_{j=1}^m \left[ Q_{(i,j)} a^2 \ln \left( \frac{1}{r_{(i,j)}} \right) \right]_{(i,j) \neq (i_p, j_p)} \quad (3.85)$$

Note that Equation 3.85 requires  $(i, j) \neq (i_p, j_p)$ .

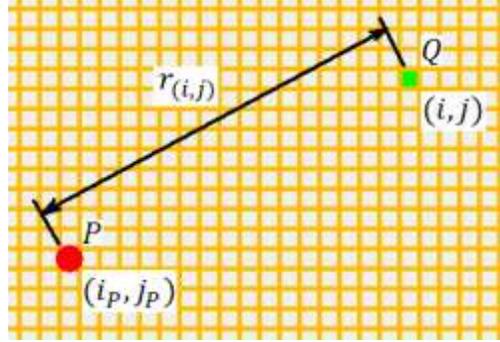


Figure 3.6 Distance  $r_{(i,j)}$  between points  $Q$  (where the second invariant is defined) and  $P$  (where the pressure is calculated)

The distance  $r_{(i,j)}$  between cell  $(i, j)$  and cell  $(i_P, j_P)$  is calculated using Equation 3.86 .

$$r_{(i,j)} = \sqrt{(x_i - x_P)^2 + (y_j - y_P)^2} \quad (3.86)$$

where  $x_i, y_i, x_P = x_{i_P}$  and  $y_P = y_{j_P}$  are the coordinates at the center of the cell using the reference location defined in Table 2.3.

Finite difference forms of the first Contour Integral ( $CI$ ) in Equation 3.84b are further decomposed and labeled as  $(CI_1)_L, (CI_1)_R, (CI_1)_T$  and  $(CI_1)_B$ . Here  $L, R, T$  and  $B$  refer to Left, Right, Top and Bottom respectively. This Contour Integral is defined as before and used in VerFlow-V.01 and in Section 4.2.

The definitions given here for  $x_i$  and  $y_j$  are used in this section:  $x_i$  is the  $x$  coordinate of the  $i$ -th column and  $y_j$ , the  $y$  coordinate of the  $j$ -th row;  $i$  is either 0, 1,  $i_P, n$  or  $n + 1$  while  $j$  is 0, 1,  $j_P, m$  or  $m + 1$  .

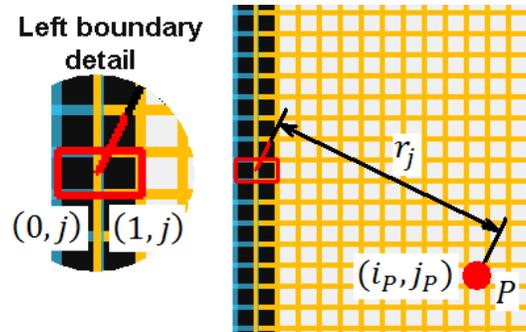


Figure 3.7 Left boundary detail and definitions for  $(CI_1)_L$

The contour integral  $(CI_1)_L$  is calculated by Equation 3.87.

$$(CI_1)_L = \frac{1}{2\pi} \sum_{j=1}^m [P_{(0,j)} - P_{(1,j)}] \ln\left(\frac{1}{r_j}\right) \quad (3.87)$$

$$r_j = \sqrt{\left(\frac{x_0 + x_1}{2} - x_p\right)^2 + (y_j - y_p)^2} \quad (3.88)$$

where  $P_{(0,j)}$  and  $P_{(1,j)}$  are the pressures at cells  $(0,j)$  and  $(1,j)$ ;  $r_j$  is the distance from cell  $(i_p, j_p)$  to the midpoint of cells  $(0,j)$  and  $(1,j)$ .

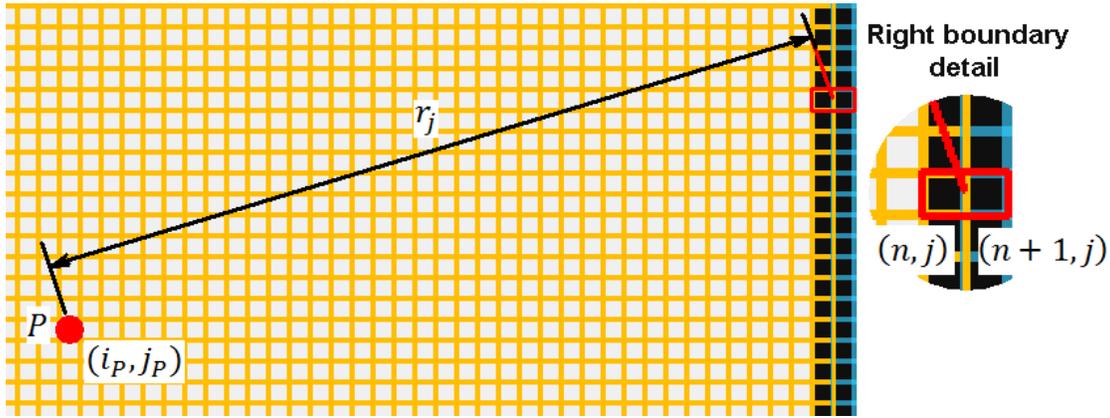


Figure 3.8 Right boundary detail and definitions for  $(CI_1)_R$

Note that the finite difference distance for the normal direction is  $a$ , which cancels with the finite difference distance for the contour integral which is also  $a$ . This occurs at the four boundaries.

Similarly, equations for the right boundary (see Figure 3.8) are written as:

$$(CI_1)_R = \frac{1}{2\pi} \sum_{j=1}^m [P_{(n+1,j)} - P_{(n,j)}] \ln\left(\frac{1}{r_j}\right) \quad (3.89)$$

$$r_j = \sqrt{\left(\frac{x_n + x_{n+1}}{2} - x_p\right)^2 + (y_j - y_p)^2} \quad (3.90)$$

Note that  $r_j$ , in Equation 3.88 applies to Equation 3.87 while  $r_j$  in Equation 3.90 applies to Equation 3.89. In Equations 3.89,  $P_{(n+1,j)}$  and  $P_{(n,j)}$  are the pressures outside and inside the subdomain along the right boundary at a vertical location defined by  $j$ .

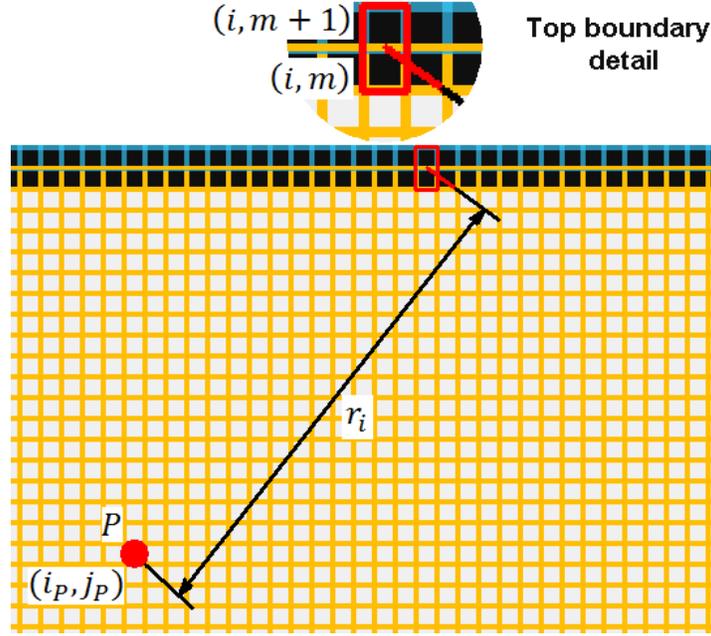


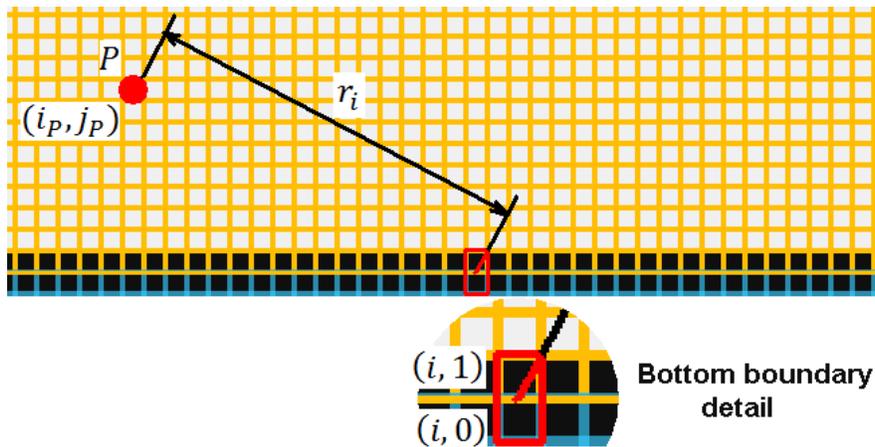
Figure 3.9 Top boundary detail and definitions for  $(CI_1)_T$

For the top boundary (see Figure 3.9), Equations 3.91 and 3.92 define  $(CI_1)_T$  and  $r_i$ .

$$(CI_1)_T = \frac{1}{2\pi} \sum_{i=1}^n [P_{(i,m+1)} - P_{(i,m)}] \ln \left( \frac{1}{r_i} \right) \quad (3.91)$$

$$r_i = \sqrt{(x_i - x_P)^2 + \left( \frac{y_m + y_{m+1}}{2} - y_P \right)^2} \quad (3.92)$$

In equation 3.91,  $P_{(i,m+1)}$  and  $P_{(i,m)}$  are the pressures outside and inside the top boundary respectively at a horizontal location defined by  $i$ .



**Figure 3.10 Bottom boundary detail and definitions for  $(CI_1)_B$**

For the bottom boundary (see Figure 3.10), the contour integrals  $(CI_1)_B$  and  $r_i$  are defined in Equations 3.91 and 3.92.

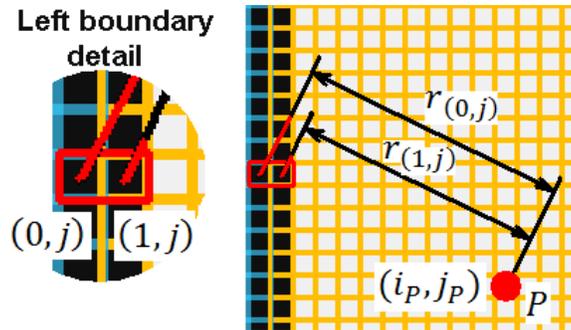
$$(CI_1)_B = \frac{1}{2\pi} \sum_{i=1}^n [P_{(i,0)} - P_{(i,1)}] \ln\left(\frac{1}{r_i}\right) \quad (3.93)$$

$$r_i = \sqrt{(x_i - x_P)^2 + \left(\frac{y_0 + y_1}{2} - y_P\right)^2} \quad (3.94)$$

For equation 3.93,  $P_{(i,0)}$  and  $P_{(i,1)}$  are the pressures outside and inside the bottom boundary respectively of the rectangular subdomain at a horizontal location established by  $i$ .

Note that the distance  $r_i$  in Equation 3.92 applies to Equation 3.91 while  $r_i$  in Equation 3.94 applies to Equation 3.93.

Finite difference forms for the second contour integral in Equation 3.84 are labeled as  $(CI_2)_L$ ,  $(CI_2)_R$ ,  $(CI_2)_T$  and  $(CI_2)_B$  for each boundary. These contour integrals are also used in VerFlow-V.01, where contributions of each component is envisioned in a *qualitative* sense and discussed in Section 4.2.



**Figure 3.11 Left boundary detail and definitions for  $(CI_2)_L$**

The contour integral  $(CI_2)_L$  is evaluated by using Equations 3.95, 3.96 and 3.97 (see Figure 3.11).

$$(CI_2)_L = -\frac{1}{2\pi} \sum_{j=1}^m P_{(0,j)} \left[ \ln \left( \frac{1}{r_{(0,j)}} \right) - \ln \left( \frac{1}{r_{(1,j)}} \right) \right]_{(1,j) \neq (i_P, j_P)} \quad (3.95)$$

$$r_{(0,j)} = \sqrt{(x_0 - x_P)^2 + (y_j - y_P)^2} \quad (3.96)$$

$$r_{(1,j)} = \sqrt{(x_1 - x_P)^2 + (y_j - y_P)^2} \quad (3.97)$$

Note again that the finite difference distance for the contour integral cancels with the finite difference distance for the normal, since both have the same magnitude  $a$ , which is the side of the square cells.

There is an additional restriction  $(1, j) \neq (i_P, j_P)$  given in Equation 3.95. Other restrictions are  $(n, j) \neq (i_P, j_P)$ ,  $(i, m) \neq (i_P, j_P)$  and  $(i, 1) \neq (i_P, j_P)$  in Equations 3.98, 3.101 and 3.104 respectively. All these restrictions limit either the location of point  $P$  to be inside the white rectangle in Figure 3.5 or the calculations to avoid that specific location in the summation.

Note also that  $r_{(0,j)}$  and  $r_{(1,j)}$  are the lengths from point  $P$  to the center of the cells  $(0, j)$  and  $(1, j)$  respectively. It is not difficult to imagine that this contour integral will be more important when point  $P$  is close to the wall. This criteria applies to the other walls as well.

Similarly,  $(CI_2)_R$  is given by using Equations 3.98, 3.99 and 3.100 (see Figure 3.12).

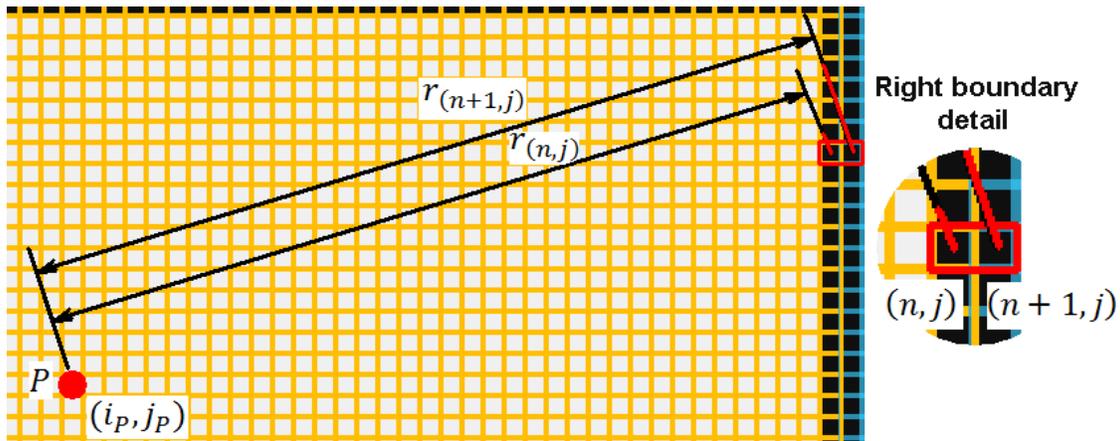


Figure 3.12 Right boundary detail and definitions for  $(CI_2)_R$

$$(CI_2)_R = -\frac{1}{2\pi} \sum_{j=1}^m P_{(n+1,j)} \left[ \ln \left( \frac{1}{r_{(n+1,j)}} \right) - \ln \left( \frac{1}{r_{(n,j)}} \right) \right]_{(n,j) \neq (i_P, j_P)} \quad (3.98)$$

$$r_{(n+1,j)} = \sqrt{(x_{n+1} - x_P)^2 + (y_j - y_P)^2} \quad (3.99)$$

$$r_{(n,j)} = \sqrt{(x_n - x_P)^2 + (y_j - y_P)^2} \quad (3.100)$$

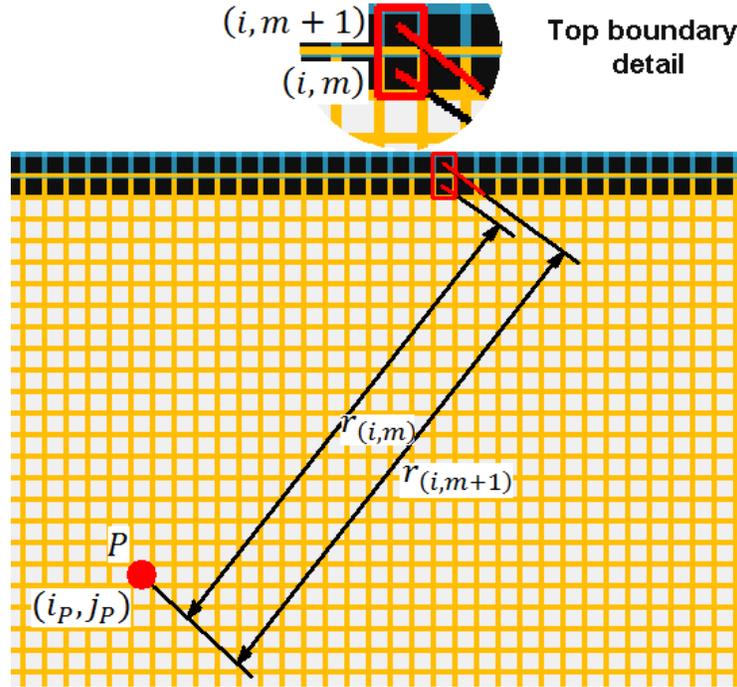


Figure 3.13 Top boundary detail and definitions for  $(CI_2)_T$

The integral on the top boundary,  $(CI_2)_T$  is evaluated by Equations 3.101, 3.102 and 3.103 (see Figure 3.13).

$$(CI_2)_T = -\frac{1}{2\pi} \sum_{i=1}^n P_{(i,m+1)} \left[ \ln \left( \frac{1}{r_{(i,m+1)}} \right) - \ln \left( \frac{1}{r_{(i,m)}} \right) \right]_{(i,m) \neq (i_P, j_P)} \quad (3.101)$$

$$r_{(i,m+1)} = \sqrt{(x_i - x_P)^2 + (y_{m+1} - y_P)^2} \quad (3.102)$$

$$r_{(i,m)} = \sqrt{(x_i - x_P)^2 + (y_m - y_P)^2} \quad (3.103)$$

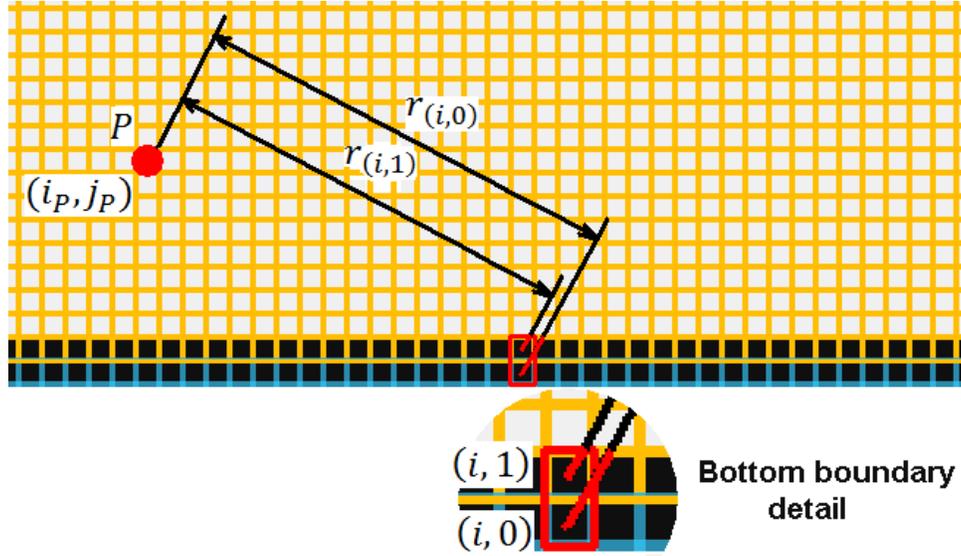


Figure 3.14 Bottom boundary detail and definitions for  $(CI_2)_B$

Finally, the contour integral  $(CI_2)_B$  is calculated by Equations 3.104, 3.105 and 3.106 (see Figure 3.14).

$$(CI_2)_B = -\frac{1}{2\pi} \sum_{i=1}^n P_{(i,0)} \left[ \ln \left( \frac{1}{r_{(i,0)}} \right) - \ln \left( \frac{1}{r_{(i,1)}} \right) \right]_{(i,1) \neq (i_P, j_P)} \quad (3.104)$$

$$r_{(i,0)} = \sqrt{(x_i - x_P)^2 + (y_0 - y_P)^2} \quad (3.105)$$

$$r_{(i,1)} = \sqrt{(x_i - x_P)^2 + (y_1 - y_P)^2} \quad (3.106)$$

### 3.4.2 Finite difference forms for the entire domain

The integration on the entire flow domain is the same as the integration on the rectangular subdomain studied in Section 3.4.1 where the only difference is that the cells around the cylinder are not squares and consequently the cell geometry must be more carefully considered.

A portion of the calculations is already reviewed in Section 3.4.1 while some terms must be added to complete the integrations. The integrals from Equation 3.84 are again classified in as a surface integral ( $Q$ ), and two contour integrals  $(CI_1)$  and  $(CI_2)$ .

The surface integral  $(Q)$ , is divided into five blocks:  $(Q)_0$ ,  $(Q)_1$ ,  $(Q)_2$ ,  $(Q)_3$ , and  $(Q)_4$  corresponding to each of the computational simulation grid blocks. The surface integral over the rectangular domain,  $(Q)_4$ , is given by equation 3.107, similar to Equation 3.85 considering  $n = 238$  and  $m = 78$  (since block 4 has  $240 \times 80$  cells), which defines the biggest rectangular domain in block 4 according to the definitions given in Figure 3.5.

$$(Q)_4 = -\frac{1}{\pi} \sum_{i=0}^{238} \sum_{j=1}^{78} \left\{ \left[ Q_{(i,j)} a^2 \ln \left( \frac{1}{r_{(i,j)}} \right) \right]_{(x_i, y_j) \neq (x_P, y_P)} \right\}_4 \quad (3.107)$$

Note the restriction of  $(x_i, y_j) \neq (x_{i_P}, y_{j_P})$  is required on the summation, since point  $P$ , located in any block, requires a unique reference, see Table 2.3. The cell where point  $P$  is defined is filtered out of the summation to avoid the problem. The subscript  $i$  varies from 0 to 238 because block 4 is connected to block 1 at the left and no physical boundary exists there. The subscript 4 indicates that all  $(i, j)$  cells corresponds to a local reference in block 4, although  $(x_i, y_j)$  and  $(x_P, y_P)$  refer to the unique reference (see Table 2.3).

For blocks 0 to 3, the integrals  $(Q)_0$ ,  $(Q)_1$ ,  $(Q)_2$  and  $(Q)_3$  are defined as equations 3.108, 3.109, 3.110 and 3.111.

$$(Q)_0 = -\frac{1}{\pi} \sum_{i=0}^{79} \sum_{j=1}^{78} \left\{ \left[ Q_{(i,j)} A_{(i,j)} \ln \left( \frac{1}{r_{(i,j)}} \right) \right]_{(x_i, y_j) \neq (x_P, y_P)} \right\}_0 \quad (3.108)$$

$$(Q)_1 = -\frac{1}{\pi} \sum_{i=0}^{79} \sum_{j=1}^{79} \left\{ \left[ Q_{(i,j)} A_{(i,j)} \ln \left( \frac{1}{r_{(i,j)}} \right) \right]_{(x_i, y_j) \neq (x_P, y_P)} \right\}_1 \quad (3.109)$$

$$(Q)_2 = -\frac{1}{\pi} \sum_{i=0}^{79} \sum_{j=1}^{78} \left\{ \left[ Q_{(i,j)} A_{(i,j)} \ln \left( \frac{1}{r_{(i,j)}} \right) \right]_{(x_i, y_j) \neq (x_P, y_P)} \right\}_2 \quad (3.110)$$

$$(Q)_3 = -\frac{1}{\pi} \sum_{i=0}^{79} \sum_{j=1}^{78} \left\{ \left[ Q_{(i,j)} A_{(i,j)} \ln \left( \frac{1}{r_{(i,j)}} \right) \right]_{(x_i, y_j) \neq (x_P, y_P)} \right\}_3 \quad (3.111)$$

The subscripts 0, 1, 2, 3 indicate that all  $(i, j)$  points refer to a local curvilinear reference in every block around the cylinder (see Figure 2.3) although  $(x_i, y_j)$  and  $(x_p, y_p)$  refer to the unique reference (see Table 2.3).  $A_{(i,j)}$  is the area of the element located at cell  $(i, j)$ .  $i$  varies from 0 to 79 since there are no physical boundaries along the diagonals which join two neighboring blocks. For blocks 0, 2 and 3,  $j$  varies from 1 to 78 since  $j = 0$  corresponds to the cylinder boundary and  $j = 79$  to the outer boundary, e.g. bottom wall for block 0, top wall for block 2 and inlet at the left for block 3. For block 1,  $j$  varies from 1 to 79 because block 1 is connected to block 4 and no physical boundary exists there.

The contour integrals  $(CI_1)$  are given in equations 3.112 to 3.121 and again identified by a subindice indicating the corresponding block and a letter indicating the boundary:  $(CI_1)_{0C}$ ,  $(CI_1)_{0B}$ ,  $(CI_1)_{1C}$ ,  $(CI_1)_{2C}$ ,  $(CI_1)_{2T}$ ,  $(CI_1)_{3C}$ ,  $(CI_1)_{3L}$ ,  $(CI_1)_{4R}$ ,  $(CI_1)_{4T}$  and  $(CI_1)_{4B}$ . Here  $L, R, T, B$  and  $C$  indicate left, right, top, bottom and cylinder respectively. Note that there is only one boundary in block 1, two boundaries in blocks 0, 2 and 3, and, three boundaries in block 4. The equations for block 4 are derived from Equations 3.89, 3.91 and 3.93.

$$(CI_1)_{4R} = \frac{1}{2\pi} \sum_{j=1}^{78} \left\{ [P_{(239,j)} - P_{(238,j)}] \ln \left( \frac{1}{r_j} \right) \right\}_4 \quad (3.112)$$

$$(CI_1)_{4T} = \frac{1}{2\pi} \sum_{i=0}^{238} \left\{ [P_{(i,79)} - P_{(i,78)}] \ln \left( \frac{1}{r_i} \right) \right\}_4 \quad (3.113)$$

$$(CI_1)_{4B} = \frac{1}{2\pi} \sum_{i=0}^{238} \left\{ [P_{(i,0)} - P_{(i,1)}] \ln \left( \frac{1}{r_i} \right) \right\}_4 \quad (3.114)$$

$$(CI_1)_{0C} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} [P_{(i,79)} - P_{(i,78)}] \ln \left( \frac{1}{r_i} \right) \right\}_0 \quad (3.115)$$

$$(CI_1)_{0B} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{4D/80}{0.5(b_{(i,0)} + b_{(i,1)}) \cos \beta_i} [P_{(i,0)} - P_{(i,1)}] \ln \left( \frac{1}{r_i} \right) \right\}_0 \quad (3.116)$$

$$(CI_1)_{1C} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} [P_{(i,79)} - P_{(i,78)}] \ln \left( \frac{1}{r_i} \right) \right\}_1 \quad (3.117)$$

$$(CI_1)_{2C} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} [P_{(i,79)} - P_{(i,78)}] \ln \left( \frac{1}{r_i} \right) \right\}_2 \quad (3.118)$$

$$(CI_1)_{2T} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{4D/80}{0.5(b_{(i,0)} + b_{(i,1)}) \cos \beta_i} [P_{(i,0)} - P_{(i,1)}] \ln \left( \frac{1}{r_i} \right) \right\}_2 \quad (3.119)$$

$$(CI_1)_{3C} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} [P_{(i,79)} - P_{(i,78)}] \ln \left( \frac{1}{r_i} \right) \right\}_3 \quad (3.120)$$

$$(CI_1)_{3L} = \frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{4D/80}{0.5(b_{(i,0)} + b_{(i,1)}) \cos \beta_i} [P_{(i,0)} - P_{(i,1)}] \ln \left( \frac{1}{r_i} \right) \right\}_3 \quad (3.121)$$

In these equations some factors and constants are not simplified, since their meaning can be more easily understood. For example,  $D\pi/320$ , is the finite difference length corresponding to  $dL$  at the cylinder boundary,  $4D/80$  is the finite difference length corresponding to  $dL$  at the external contour,  $0.5(b_{(i,79)} + b_{(i,78)})$  is the finite difference radial length between the midpoints of cells  $(i, 79)$  and  $(i, 78)$ , and,  $(b_{(i,0)} + b_{(i,1)})$  is the finite difference radial length between the midpoints of cells  $(i, 0)$  and  $(i, 1)$ .

For these equations  $b_{(i,j)}$  represents the mean radial length of cell  $(i, j)$ . The geometric relation which defines the change of length in the radial direction in blocks 0, 1, 2 and 3 with  $f = 0.25$  must be considered when calculating  $b_{(i,j)}$  (see Section 2.3).  $\beta_i$  is the angle between the radial direction and the normal to the boundary. This angle  $\beta_i$  is considered for the external walls in blocks 0, 2 and 3 because for all other walls, including the cylinder,  $\beta_i$  is 0.0 and  $\cos \beta_i$  is 1.0.

Ten contour integrals  $(CI_2)$  are calculated from equations 3.122 to 3.131:  $(CI_2)_{0C}$ ,  $(CI_2)_{0B}$ ,  $(CI_2)_{1C}$ ,  $(CI_2)_{2C}$ ,  $(CI_2)_{2T}$ ,  $(CI_2)_{3C}$ ,  $(CI_2)_{3L}$ ,  $(CI_2)_{4R}$ ,  $(CI_2)_{4T}$  and  $(CI_2)_{4B}$ . Again, subscripts

outside the parenthesis denote the corresponding blocks and letters denote the boundary segment. For block 4, these contour integrals are derived from 3.98, 3.101 and 3.104.

$$(CI_2)_{4R} = -\frac{1}{2\pi} \sum_{j=1}^{78} \left\{ P_{(239,j)} \left[ \ln \left( \frac{1}{r_{(239,j)}} \right) - \ln \left( \frac{1}{r_{(238,j)}} \right) \right]_{(x_{239,y_j}) \neq (x_P, y_P)} \right\}_4 \quad (3.122)$$

$$(CI_2)_{4T} = -\frac{1}{2\pi} \sum_{i=0}^{238} \left\{ P_{(i,79)} \left[ \ln \left( \frac{1}{r_{(i,79)}} \right) - \ln \left( \frac{1}{r_{(i,78)}} \right) \right]_{(x_{i,y_{78}}) \neq (x_P, y_P)} \right\}_4 \quad (3.123)$$

$$(CI_2)_{4B} = -\frac{1}{2\pi} \sum_{i=0}^{238} \left\{ P_{(i,0)} \left[ \ln \left( \frac{1}{r_{(i,0)}} \right) - \ln \left( \frac{1}{r_{(i,1)}} \right) \right]_{(x_{i,y_1}) \neq (x_P, y_P)} \right\}_4 \quad (3.124)$$

$$(CI_2)_{0C} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} P_{(i,79)} \left[ \ln \left( \frac{1}{r_{(i,79)}} \right) - \ln \left( \frac{1}{r_{(i,78)}} \right) \right]_{(x_{i,y_{78}}) \neq (x_P, y_P)} \right\}_0 \quad (3.125)$$

$$(CI_2)_{0B} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{4D/80}{0.5(b_{(i,0)} + b_{(i,1)}) \cos \beta_i} P_{(i,0)} \left[ \ln \left( \frac{1}{r_{(i,0)}} \right) - \ln \left( \frac{1}{r_{(i,1)}} \right) \right]_{(x_{i,y_1}) \neq (x_P, y_P)} \right\}_0 \quad (3.126)$$

$$(CI_2)_{1C} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} P_{(i,79)} \left[ \ln \left( \frac{1}{r_{(i,79)}} \right) - \ln \left( \frac{1}{r_{(i,78)}} \right) \right]_{(x_i, y_{78}) \neq (x_P, y_P)} \right\}_1 \quad (3.127)$$

$$(CI_2)_{2C} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} P_{(i,79)} \left[ \ln \left( \frac{1}{r_{(i,79)}} \right) - \ln \left( \frac{1}{r_{(i,78)}} \right) \right]_{(x_i, y_{78}) \neq (x_P, y_P)} \right\}_2 \quad (3.128)$$

$$(CI_2)_{2T} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{4D/80}{0.5(b_{(i,0)} + b_{(i,1)}) \cos \beta_i} P_{(i,0)} \left[ \ln \left( \frac{1}{r_{(i,0)}} \right) - \ln \left( \frac{1}{r_{(i,1)}} \right) \right]_{(x_i, y_1) \neq (x_P, y_P)} \right\}_2 \quad (3.129)$$

$$(CI_2)_{3C} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{D\pi/320}{0.5(b_{(i,79)} + b_{(i,78)})} P_{(i,79)} \left[ \ln \left( \frac{1}{r_{(i,79)}} \right) - \ln \left( \frac{1}{r_{(i,78)}} \right) \right]_{(x_i, y_{78}) \neq (x_P, y_P)} \right\}_3 \quad (3.130)$$

$$(CI_2)_{3L} = -\frac{1}{2\pi} \sum_{i=0}^{79} \left\{ \frac{4D/80}{0.5(b_{(i,0)} + b_{(i,1)}) \cos \beta_i} P_{(i,0)} \left[ \ln \left( \frac{1}{r_{(i,0)}} \right) - \ln \left( \frac{1}{r_{(i,1)}} \right) \right]_{(x_i, y_1) \neq (x_P, y_P)} \right\}_3 \quad (3.131)$$

In Equations 3.122 to 3.131,  $b_{(i,j)}$  is the mean length in the radial direction,  $r_{(i,j)}$  is the distance from point  $P$ , center of cell  $(i_P, j_P)$ , to the center point of the cell defined by the local reference  $(i, j)$  in the block, which is specified by the global subindice in each equation.  $P_{(i,j)}$  is the pressure at cell  $(i, j)$  in the specific block. As noted before for the contour integrals  $(CI_1)$  in this

section,  $\beta_i$  is the angle between the radial direction and the normal and is applied to the external walls in blocks 0, 2 and 3.

### 3.5. Chapter nomenclature

$\alpha$	angle which gives the location of a cylinder boundary cell
$\beta_i$	angle between the radial direction for cells $(i, j)$ and the normal to external wall
$\Delta$	finite difference
$\pi$	constant = 3.14159265359
$\nu$	kinematic viscosity
$a$	side length of the square cells in the grid
$A$	area
$A_{(i,j)}$	area of the cell $(i, j)$
$a_c$	side of the finite difference cells parallel to the cylinder boundary
$b$	side length of the finite difference cells perpendicular to the cylinder boundary
$B$	subindice meaning Bottom
$b_{(i,j)}$	side length of the finite difference cell $(i, j)$ in the radial direction
$(CI_1)$	first term of contour integral in Equation 3.84
$(CI_1)_{L/R/T/B/C}$	first term of contour integral in Equation 3.84 (left, right, top, bottom or cylinder boundary)
$(CI_1)_{kL/kR/kT/kB/kC}$	first term of contour integral in Equation 3.84 (left, right, top, bottom or cylinder boundary) in block $k$
$(CI_2)$	second term of contour integral in Equation 3.84

$(CI_2)_{L/R/T/B/C}$  second term of contour integral in Equation 3.84 (left, right, top, bottom or cylinder boundary)

$(CI_2)_{kL/kR/kT/kB/kC}$  second term of contour integral in Equation 3.84 (left, right, top, bottom or cylinder boundary) in block  $k$

$D$  diameter of the cylinder

$\bar{F}$  force on the cylinder

$f_i$  body forces  $i = 1,2,3$

$k$  index for the summation

$F_P$  force on the cylinder due to pressure

$\bar{F}_P$  force on the cylinder due to pressure (vector)

$\bar{F}_{P_k}$   $k$ -term of the force on the cylinder due to pressure (vector)

$\bar{F}_{P_{total}}$  total force on the cylinder due to pressure (vector)

$\bar{F}_{P_{total\ drag}}$  total drag force on the cylinder due to pressure (vector)

$\bar{F}_{P_{total\ lift}}$  total lift force on the cylinder due to pressure (vector)

$\bar{F}_{P_x}$  drag force on the cylinder due to pressure (vector)

$\bar{F}_{P_{x_k}}$   $k$ -term of the drag force on the cylinder due to pressure (vector)

$\bar{F}_{P_y}$  lift force on the cylinder due to pressure (vector)

$\bar{F}_{P_{y_k}}$   $k$ -term of the lift force on the cylinder due to pressure (vector)

$F_V$  force on the cylinder due to viscous effects

$\bar{F}_V$  force on the cylinder due to viscous effects (vector)

$\bar{F}_{V_k}$   $k$ -term of the force on the cylinder due to viscous effects (vector)

$\bar{F}_{V_{total}}$  total force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{V_{total\ drag}}$  total drag force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{V_{total\ lift}}$  total drag force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{V_x}$  drag force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{V_{x_k}}$  k-term of the drag force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{V_y}$  lift force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{V_{y_k}}$  k-term of the lift force on the cylinder due to viscous effects (vector)  
 $\bar{F}_{total}$  total force on the cylinder (vector)  
 $\bar{F}_{total\ drag}$  total drag force on the cylinder (vector)  
 $\bar{F}_{total\ lift}$  total drag force on the cylinder (vector)  
 $i$  cell number in the  $x$  direction or its left wall  
 $i$  index 1, 2 or 3  
 $(i_P, j_P)$  cell corresponding to point  $P$   
 $\bar{i}$  unit horizontal vector  
 $j$  cell number in the  $y$  direction or its bottom wall  
 $j$  index 1, 2 or 3  
 $\bar{j}$  unit vertical vector  
 $L$  countour length or subindice meaning Left.  
 $m$  number of cells inside the rectangular subdomain in the vertical direction  
 $\bar{M}$  moment (vector)

$\bar{M}_{total}$	total moment (vector)
$n$	number of cells inside the rectangular subdomain in the horizontal direction
$\bar{n}$	normal vector pointing outside the cylinder
$P$	pressure per unit density or point in an arbitrary domain
$P_{(i,j)}$	pressure per unit density at cell (i,j)
$P_P$	pressure per unit density at point $P$
$Q$	second invariant of the velocity gradient or point in an arbitrary domain
$(Q)$	surface integral in Equation 3.84.
$(Q)_k$	surface integral in Equation 3.84 for block $k$
$Q_Q$	second invariant of the velocity gradient at point $Q$
$Q_{(i,j)}$	second invariant of the velocity gradient at cell (i,j)
$r$	cylinder radius or distance from point $P$ to point $Q$
$\bar{r}$	cylinder radius (vector)
$R$	subindice meaning right
$r_i$	distance from point $P$ to a point defined by a specific boundary and a given $i$
$r_{(i,j)}$	distance from point $P$ to cell (i, j)
$r_j$	distance from point $P$ to a point defined by a specific boundary and a given $j$
$\bar{r}_k$	k-term of the cylinder radius (vector)
$t$	time
$T$	subindice meaning Top
$u$	velocity in the $x$ direction at a point

$u_i$  velocity  $i = 1,2,3$

$u_{tg}$  tangential velocity

$\bar{u}_{tg}$  tangential velocity vector

$v$  velocity in the  $y$  direction at a point

$x$  axis 1

$x_i$  axes  $i = 1,2,3$  or horizontal location of cell  $(i, j)$

$x_p$  =  $x_{i_p}$  , horizontal location of cell  $(i_p, j_p)$

$y$  axis 2

$y_i$  =  $y_{i_p}$  , vertical location of cell  $(i, j)$

$y_p$  vertical location of cell  $(i_p, j_p)$

$z$  thickness

## Chapter 4 . Results and Discussion

A customized graphical computer program, VerFlow-V.01, was developed here for both quantitative and *qualitative* analysis of periodic laminar flow around a 2D cylinder. In VerFlow-V.01 the equations defined in Chapter 3 are applied to the OpenFoam simulation data from Chapter 2.

Unless a different velocity is specified, discussion of results here focus on the problem where the velocity at the inlet is 0.012 [m/s] ( $Re=67.4$ ).

### 4.1. Forces on the cylinder

Various drag coefficients, for flow around a circular cylinder are shown in figure 1.2. Here this figure is used for a *qualitative* description only, where the coefficients of drag are plotted using a logarithmic scale along the y-axis. The figure in Chapter 1 (Feynman, Leighton, & Sands, 1964), gives us a first reference point with drag coefficient 1.54 at the Reynolds number of 67, i.e. the blue line in Figure 4.1. A second source, (Warsi, 1993) is shown as a red line, point (67,1.95), in Figure 4.1.

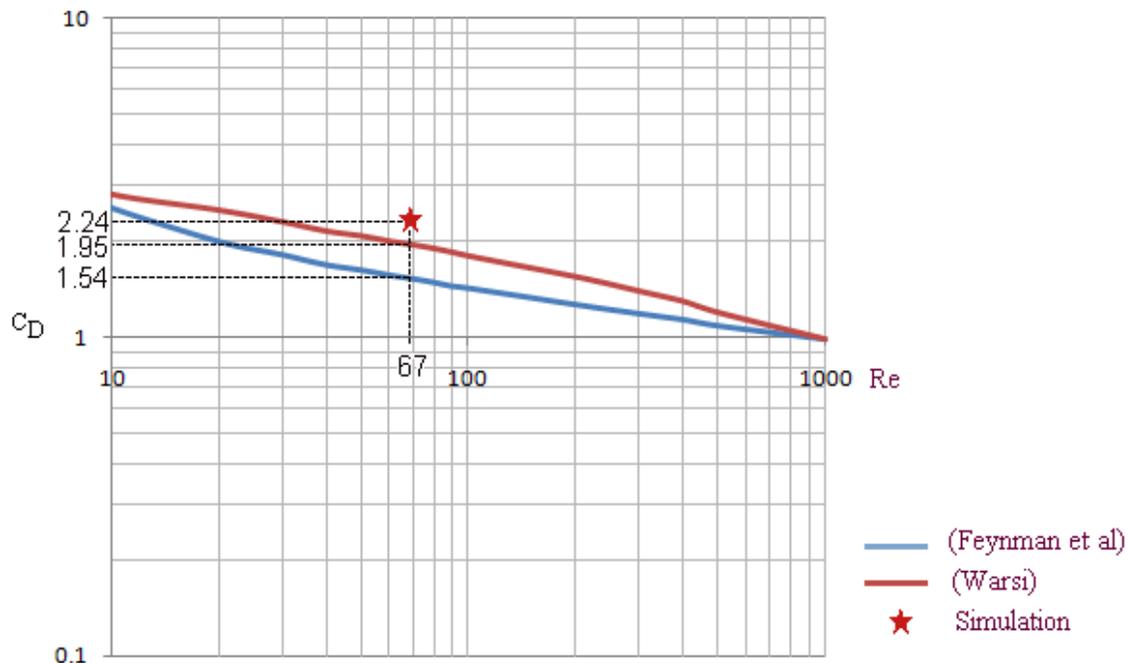


Figure 4.1 Drag coefficient of a cylinder for  $Re=67$

Included in Figure 4.1 are our results (67, 2.24) which are obtained by numerical integration of pressure contributions and viscous forces acting on the cylinder, see equations 1.20, 3.55, 3.68, 3.71, 3.74, 3.77.

The difference in the coefficients of drag, compared in Figure 4.1, is believed to be the result of the flow constriction generated by the top and bottom walls. This constriction also affects the shedding frequency and consequently the Strouhal number.

Equation 1.25 is used to calculate the drag coefficient as a function of time, based on integrating dynamically horizontal (streamwise) forces acting on the cylinder boundary. A similar expression applies for the lift coefficient by integration of vertical forces. This makes sense since the goal is to observe how drag and lift varies with time, although the expected mean value for the lift is zero. For this reason animations are included in this discussion.

Figure 4.2 and its corresponding Animation 4.1 shows the dynamic relationship between components of drag and lift forces and forces acting on the cylinder boundary. All forces are drawn concentric to the center of the cylinder. The pressure force, drawn in orange, is the result of the integration of the pressure along the cylinder boundary. The pressure at each point on the cylinder surface is also represented as acting along radial lines, in cyan. Positive pressure coefficients point away from the yellow cylinder boundary. The blue lines are viscous forces. The black line represents the total force acting on the cylinder. The streamwise velocity field,  $u_x$ , is also shown, using the blue and green color legend shown at the left. The maximum dimensionless force in a complete cycle is 2.29. Note that forces are calculated in dimensionless form, i.e. the drag coefficient,  $C_D$ , and lift coefficient,  $C_L$ .

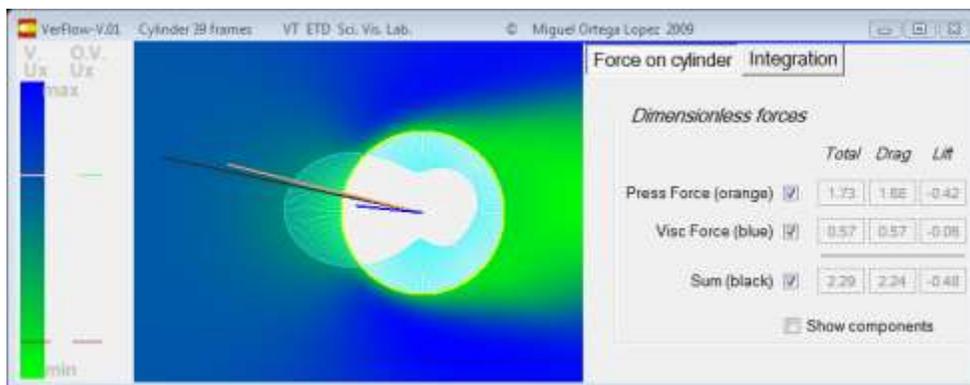
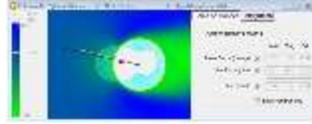


Figure 4.2 Pressure and viscous forces on the cylinder



Animation 4.1 [Forces on the cylinder](#)

Animation 4.1 reveals the relative changes of lift and drag force components in a dynamic sense not seen in Figure 4.2. The dynamic interaction of these forces with the  $u_x$  velocity is especially insightful in a *qualitative* sense. A quantitative discussion on dynamic pressure and viscous force components is given in Section 4.1.3.

#### **4.1.1 General discussion on drag forces**

The total dimensionless drag varies from 2.246 to 2.258. The drag resulting from viscous forces can essentially be considered constant since its variation is very small, from 0.581 to 0.583 which accounts for 25.8% of the drag at this particular Reynolds number. The pressure generates the majority of the drag at 74.2%. In the simulation, the cylinder boundary is divided into four arcs, 0 to 3, which correspond to the grid blocks 0 to 3 in Figure 2.2. The pressure drag contributions are 52.3% from arc 3 (left), 41.7% from arc 1 (right). The difference to complete 100%, is 6%, which is due to the pressure from arcs 0 (bottom) and 2 (top).

Viscous forces from arc 0 (bottom) and arc 2 (top) contribute equally and sum to 77.9% of the total viscous drag. 23.9% is generated at arc 3 (left). Arc 1 (right) actually has a negative effect of -1.7% on the viscous drag, which is the result of “back flow” behind the cylinder.

These results have a logical meaning considering that the pressure acts perpendicularly to the walls, and the viscous forces interact tangentially.

#### **4.1.2 General discussion on lift forces**

The variation in lift forces is the result of pressure and tangential viscous forces acting on the cylinder boundary. The pressure and the tangential velocity changes periodically in such a way that both affect the total lift.

The maximum dimensionless lift is 0.50 where 13.3% is due to viscous forces while 86.7% is due to pressure forces. The lift varies in time from positive to negative and positive again as the flow continues to develop in repeating cycles. The mean lift is zero due to symmetry.

#### 4.1.3 Pressure and viscous forces in one cycle

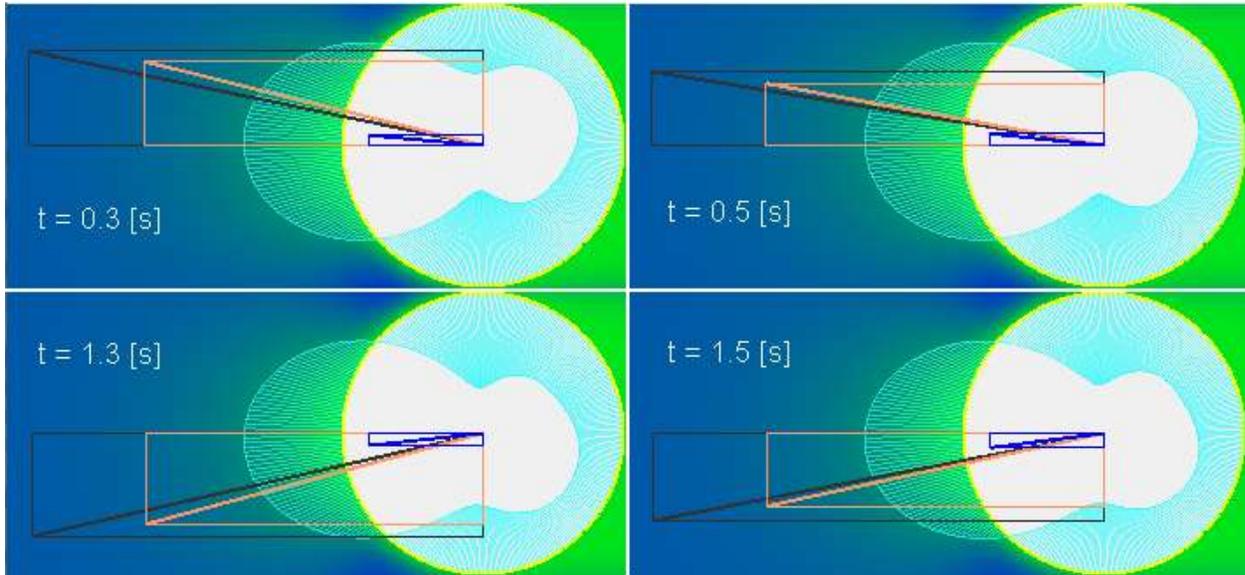


Figure 4.3 Times of maximum forces 0.3[s] top left, 0.5[s] top right, 1.3[s] bottom left and 1.5[s] bottom right

A complete cycle is envisioned within 39 frames for the problem where  $u_{\infty} = 0.012[m/s]$  ( $Re=67.4$ ). To get this result, the same procedure described in Section 2.7.5, is applied. The period is  $0.2[s](10)(39)/40 = 1.95[s]$  and the frequency of the shedding oscillation is  $0.513[Hz]$ .

The detail of how the pressure and viscous forces appear when at their respective maximum values in the cycle is shown in Figure 4.3. The time order of these graphs is top-left (0.3[s]), top-right (0.5[s]), bottom-left (1.3[s]) and bottom-right (1.5[s]). The figures on the left side correspond to maximum pressure forces and the figures on the right side to maximum viscous forces.

The maximum values from both sources do not occur simultaneously. The maximum viscous forces are delayed with respect to the maximum pressure forces by  $0.2[s]$  ( $37^{\circ}$  of the complete cycle). This can be seen in Animation 4.1.

Since the pressure force is considerably larger than the viscous force and the viscous force is delayed with respect to the pressure force, the total force (vectorial sum of both forces) does not reach the maximum angle reached by the pressure force.

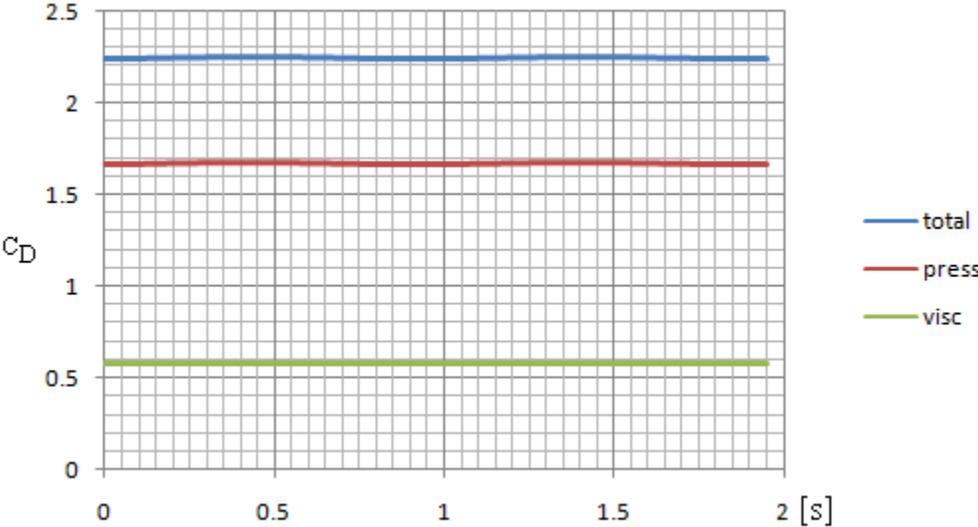


Figure 4.4 Drag forces on the cylinder along one cycle

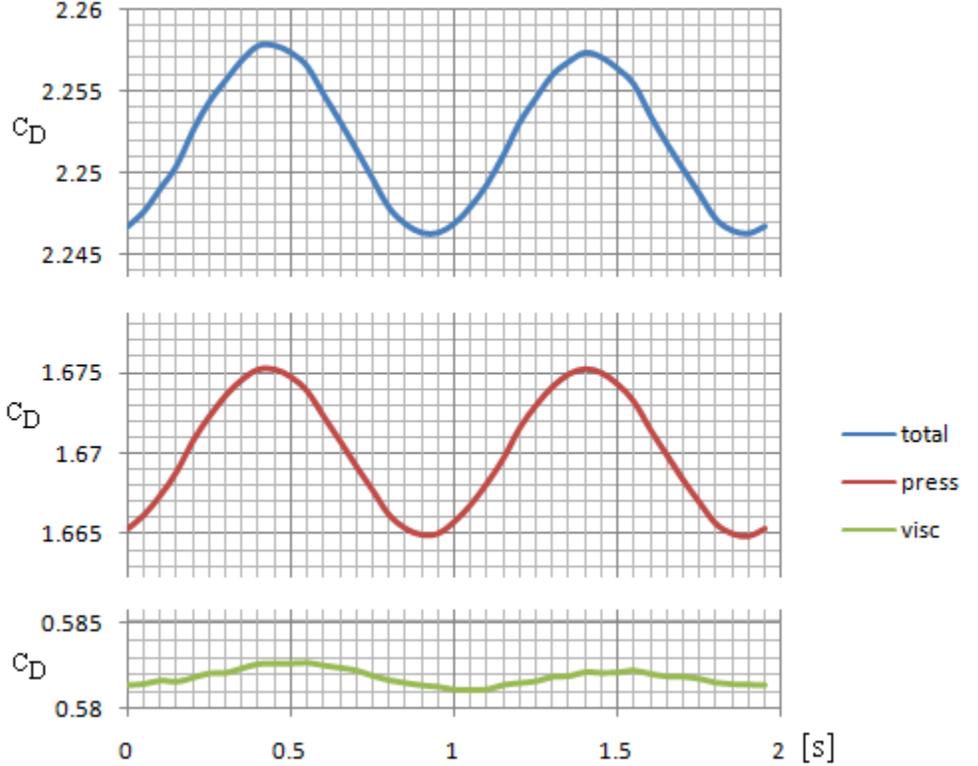


Figure 4.5 Amplification (100 times) of drag forces shown on Figure 4.4 for one complete cycle

In figure 4.4 we confirm what was already stated on 4.1.1 that the drag forces are fairly constant during the entire cycle. Also, viscous forces generate 25.8% of the total drag.

The scale of the vertical axis in figure 4.4 is amplified by 100, and the result is plotted in Figure 4.5. Our results are consistent with Osama Marzouk’s previous result where they observed small variations in the drag with a frequency twice the shedding frequency (Marzouk, 2009).

For the following analysis, the increased time resolution data is not used but rather the results are determined from the original “data” over several cycles.

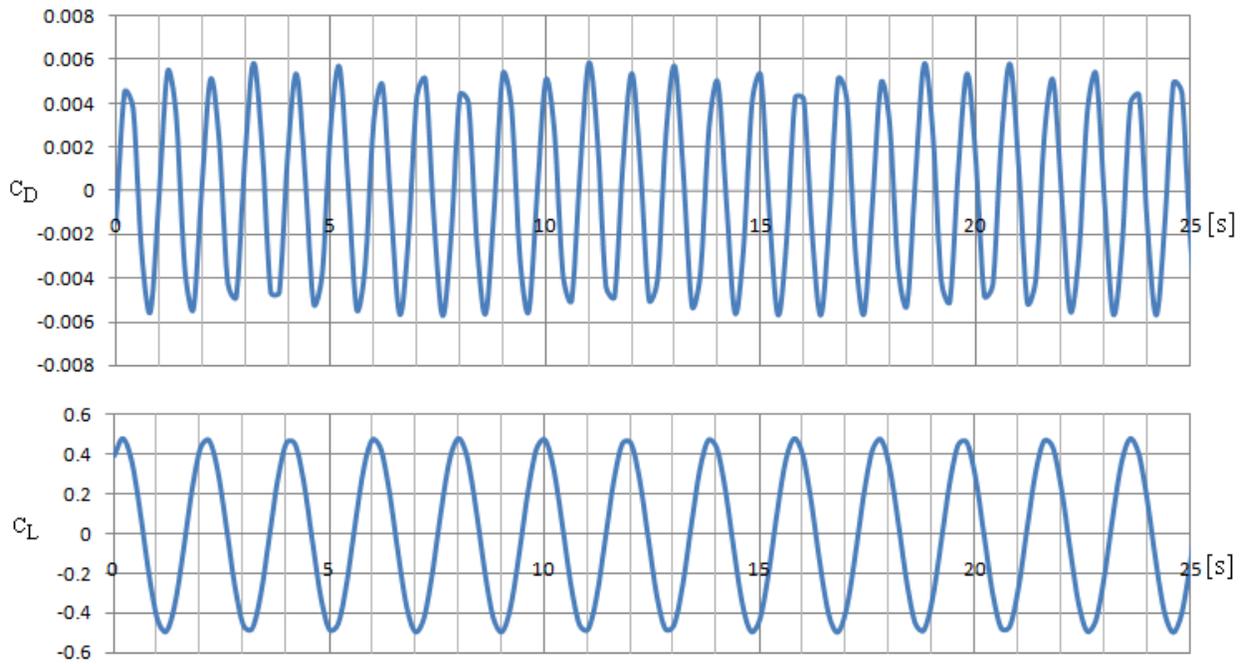


Figure 4.6 Drag (top) and lift (bottom) oscillations for Fast Fourier Transform analysis (see Figure 4.7)

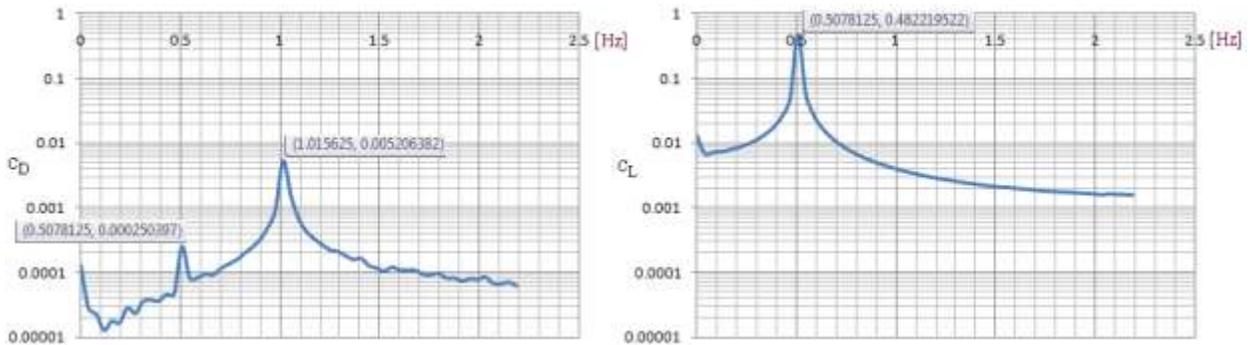


Figure 4.7 Drag and lift oscillations frequency spectrums

Figure 4.7 shows the frequency spectrum resulting from 128 points of the original drag and lift data obtained using VerFlow\_V.01 and shown in Figure 4.6. The variations in the drag coefficient from its mean value have two main frequencies: one coincides with the shedding frequency and the other with twice the shedding frequency. The lift oscillates with the characteristic shedding frequency.

A low frequency pattern can be detected in Figure 4.6. This low frequency is approximately 0.12[Hz] or one fourth of the shedding frequency, and is not identified in the frequency analysis in Figure 4.7 due to the limited number of terms (128 points).

The discussion below returns to the data set associated with the increased time resolution (1 cycle in 39 frames).

Figure 4.8 shows the lift components for the time where the maximum lift forces occur. This graph has the same arbitrary reference used in figure 4.3.

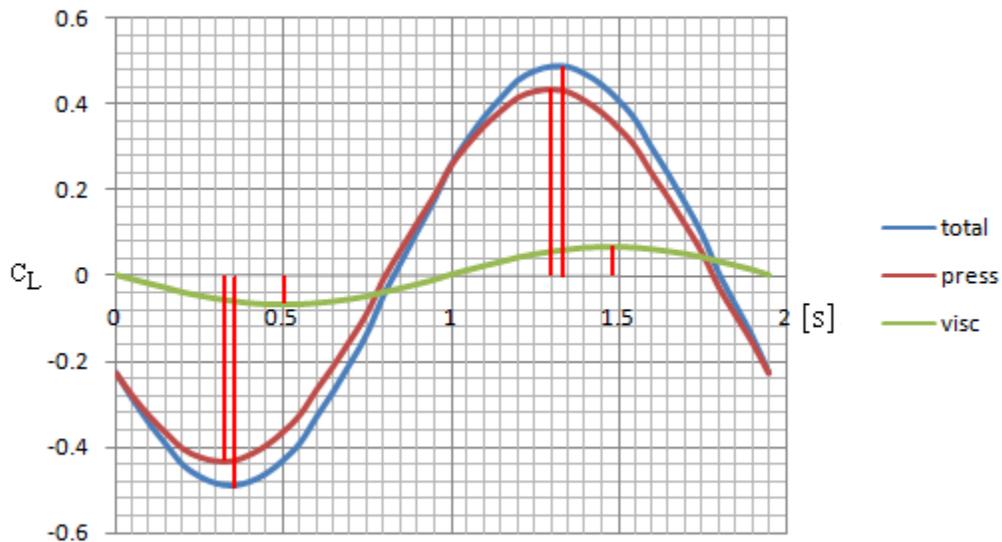


Figure 4.8 Lift forces on the cylinder along one cycle

The delay of maximum viscous forces results in the delay of the total lift with respect to pressure forces. The approximate delays are 0.175[s] (9.0% of the period or 32.3° of the shedding oscillation) for the maximum viscous forces and 0.025[s] (1.3% of the period or a phase lag of 4.6° lag) for the total lift. This difference can also be observed *qualitatively* in Animation 4.1.

#### 4.1.4 Cyclic pressure and viscous forces decomposed into blocks

Grid blocks 0, 1, 2 and 3 defined in Section 2.3 are used here to calculate pressure and viscous force contributions to lift and drag coefficients along arcs corresponding to blocks 0 to 3.

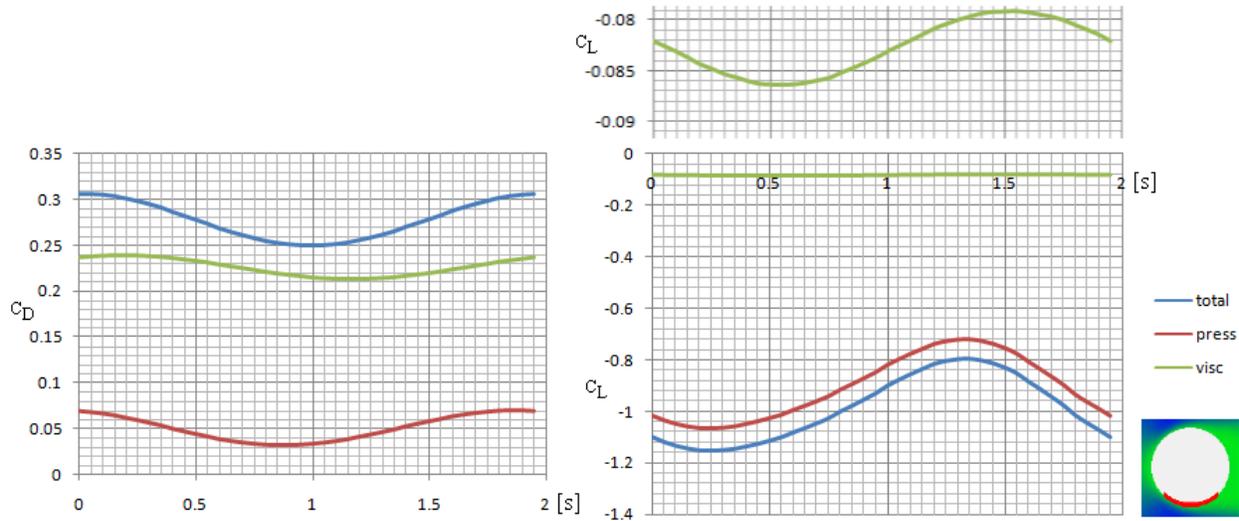


Figure 4.9 Contributions to drag (left) and lift (right) coefficients from arc 0 (in one cycle)

Cyclic results from arc 0 can be seen in Figure 4.9, drag coefficients at the left and lift coefficients at the right. From this arc we find that the pressure contribution to the drag forces cycles from 13.0% to 23.0%. The remaining 77.0% to 67% is associated with the viscous forces. The pressure contribution to the lift varies from 90.0% to 92.7%. Viscous forces act with only a small contribution to the total lift at arc 0. The variation in time, shown at the top right corner in Figure 4.9, is very small where the scale is increased 40 times. Consequently viscous forces contribute only 7.3% to 10.0% of the total lift. Lift contributions from both sources are negative values along this arc.

Figure 4.10 shows the contributions to the drag (left) and lift (right) from arc 1, which is behind the cylinder. The drag, that originates from viscous forces, is negative, negligible and oscillates at double the frequency compared to figure 4.9. These negligible values oscillate around -0.01 and are expected to actually have a mean value of zero because of symmetry. This is an acceptable error when considering various numeric calculations including the OpenFoam simulation “data”.

Cyclic results for arc 1 reveal 100% of the drag is due to pressure forces. The doubling of the frequency due to pressure dominates, since the other viscous effects are only visible when the scale on the vertical axis is increased by 25, see the bottom left corner of Figure 4.10. Although the lift for arc 1 has small values when compared with the lift from other arcs, it has significant influence from both pressure and viscous forces.

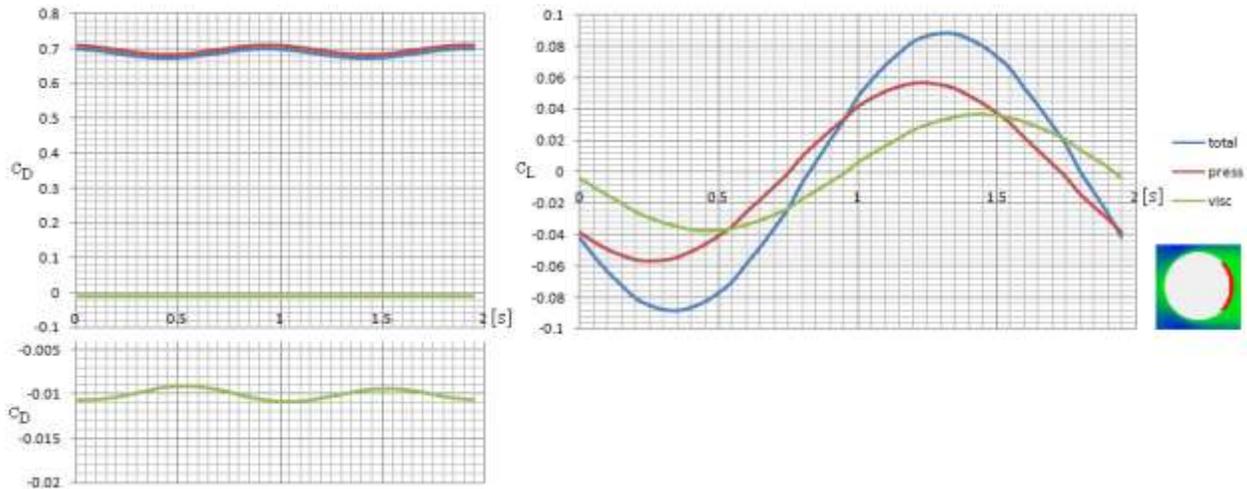


Figure 4.10 Contributions to drag (left) and lift (right) coefficients from arc 1 (in one cycle)

In Figure 4.11, we see the cyclic drag and lift results for arc 2. As expected the drag in this graph has the same maximum, minimum and mean values compared to Figure 4.09, because of the symmetry. The cyclic drag effects from arcs 0 and 2 are displaced a half cycle from each other.

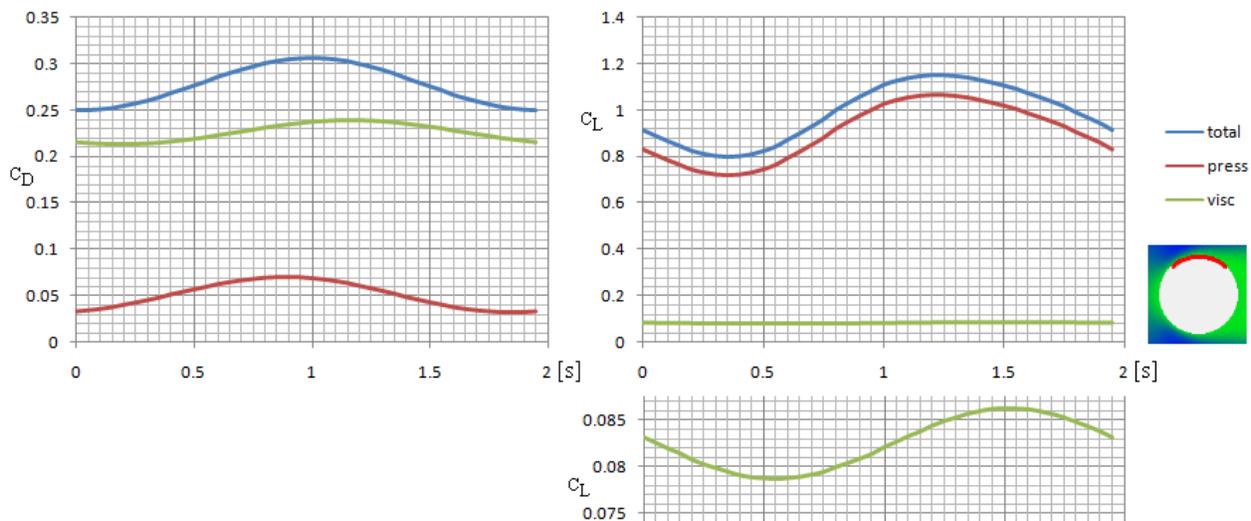


Figure 4.11 Contributions to drag (left) and lift (right) coefficients from arc 2 (in one cycle)

The sum of lift coefficients from arc 2 and arc 0 increases the individual oscillatory effect although the mean value goes to zero. The lift from viscous effects on arc 2 is again observed to have negligible oscillatory variations with the shedding frequency, see the bottom right corner in Figure 4.11 where the vertical scale is increased 40 times.

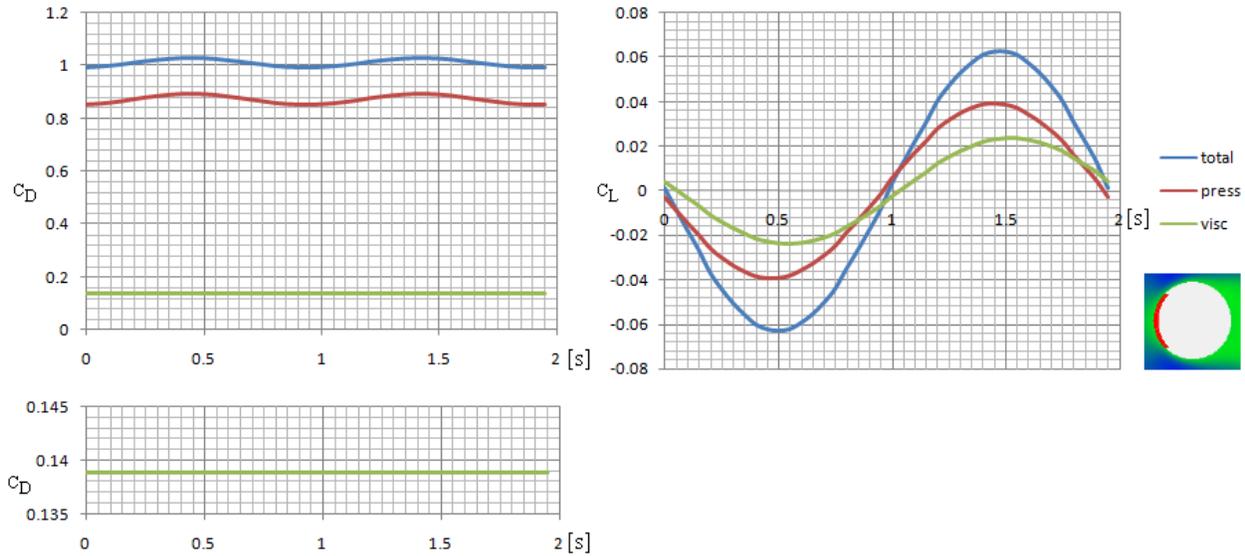


Figure 4.12 Contributions to drag (left) and lift (right) coefficients from arc 3 (in one cycle)

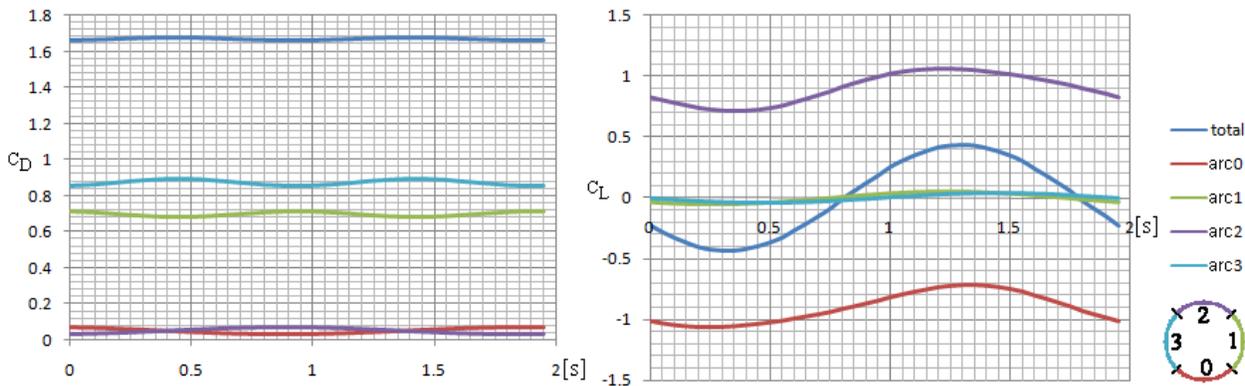


Figure 4.13 Drag (left) and lift (right) coefficient contributions from arcs 0, 1, 2 and 3 due to pressure (in one cycle)

The arc that encounters the flow directly is arc 3 where we observe the cyclic drag (left) and lift (right) contributions in Figure 4.12. A constant value of drag coefficient, 0.139, is due to the viscous effect. Again the pressure generates a doubled frequency, which was already pointed out for arc 1. The sum of arcs 1 and 3 also have a cancelling effect since the oscillation for arc 3 is inverted with respect to the same oscillation observed for arc 1. In arc 3 the lift is due to

significant contributions from pressure and viscous forces. The total lift coefficient varies from -0.0627 to 0.0625. These are reasonable values as we expected these same magnitudes because of the symmetry.

Figure 4.13 reveals a global picture of drag and lift coefficient components due to pressure from each arc. The most important contributions to the drag come from arc 3 and 1 (in that order), similarly but for the lift, the results from arcs 0 and 2 are the most important. The effects of both individual oscillations are summed, which increases the amplitude of the cyclic lift due to pressure.

In Figure 4.14 we show that the total drag at the left is the sum of the drag components from arc 3 and arc 1 and the sum of the drag components from arc 0 and arc 2, where the vertical scale is equally spaced for comparison purposes. However these are only partial results due to pressure.

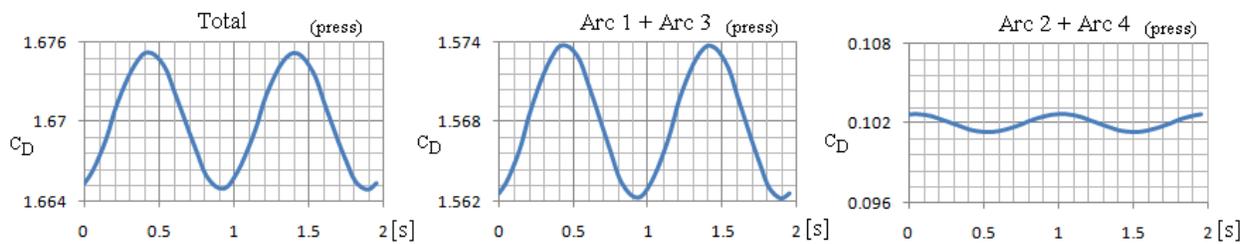


Figure 4.14 Pressure drag coefficients in one cycle: Total (left), sum arcs 3 and 1 (center), and, sum arcs 0 and 2 (right)

Arcs 1 and 3 are observed to not only contribute 93.9% of the total pressure drag magnitude, but are also responsible for the final oscillatory variations. The total pressure drag variations result from the oscillatory behavior from arcs 1 and 3, which is attenuated by arcs 0 and 2 by 9.6%.

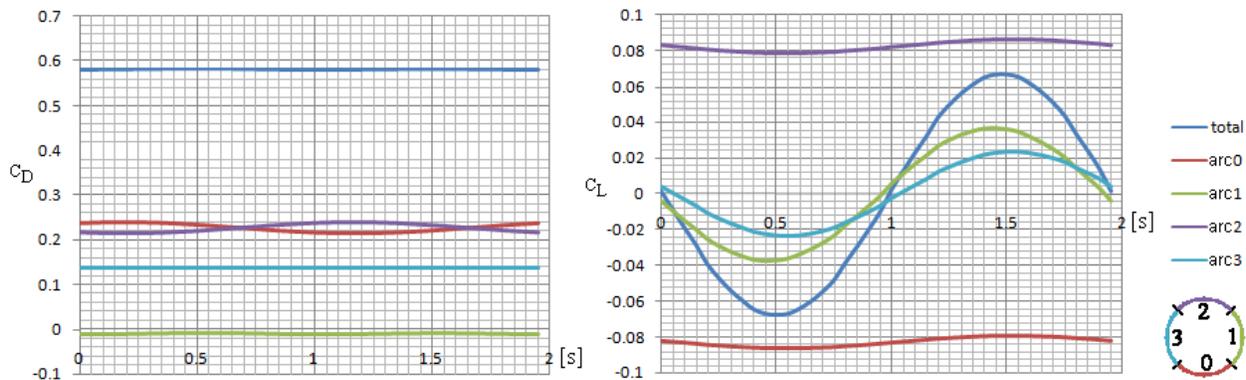
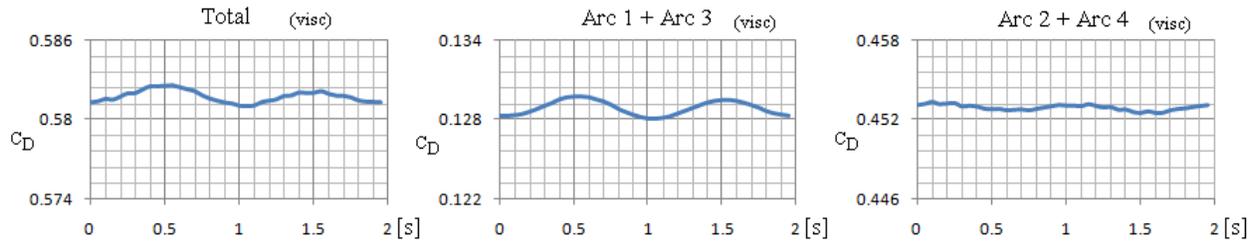


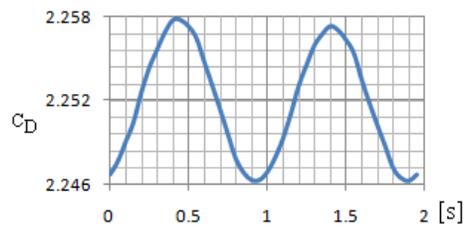
Figure 4.15 Drag (left) and lift (right) coefficient contributions from arcs 0, 1, 2 and 3 due to viscous flow (in one cycle)

In Figure 4.15 drag and lift coefficients from each arc are compared. The drag due to viscous forces is negligible from arc 0, is small from arc 3 (23%), and shows an oscillation from arcs 0 and 2 (77%). The coefficient of lift due to viscous forces is in general small and oscillates from -0.064 and 0.064.



**Figure 4.16** Viscous drag coefficients in one cycle: Total (left), sum arcs 3 and 1 (center), and, sum arcs 0 and 2 (right)

Figure 4.16 shows the viscous effect by combining arc 1 with arc 3 and arc 0 with arc 2. The same amplified vertical scale in Figure 4.14 is used in Figure 4.16 for comparison.



**Figure 4.17** Total drag coefficient in one cycle

Figure 4.17 shows again that the total drag coefficient is the sum of both pressure and viscous forces, using the same scale amplification as that used in Figures 4.16 and 4.14. When Figure 4.17 is compared with Figure 4.14 the oscillation for the total drag precisely coincides in amplitude and phase with the oscillation for the combined action of pressure drag from arcs 3 and 1. The remaining oscillations cancel each other. Will this result hold for other Reynolds numbers? Or is this result just a coincidence?

One additional flow example is considered for comparison with an inlet velocity  $u_\infty = 0.01[cm/s]$  ( $Re=56.2$ ). For this simulation, OpenFoam “data” is extracted and results analyzed using VerFlow-V.01. Figure 4.18 shows again a good prediction of the oscillation. We realize that although some noise exists, viscous forces again affect the oscillation in the total drag, but

the general picture is similar to the pressure drag sum from arcs 1 and 3. Both maintain the same frequency and the same average amplitude, shown in the shaded regions.

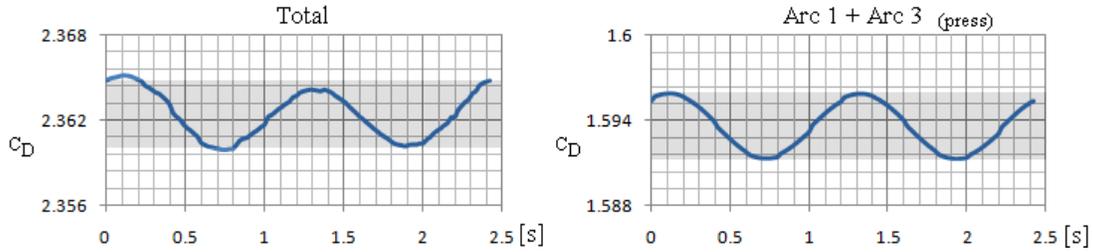


Figure 4.18 Total drag (left) and pressure drag sum from arcs 1 and 3 (right) for  $u_x = 0.01[cm/s]$  in one cycle

## 4.2. Prediction of the pressure at a point

In this section the horizontal (streamwise) velocity at the inlet is  $u_\infty = 0.012[m/s]$  ( $Re=67.4$ ). The calculations shown here and in Section 4.3, are based on Equation 3.84. The objective of this section is to understand how each term in Equation 3.84 contributes to pressure at any arbitrary point,  $P_p$ , using analytic and graphic models.

$$P_p = -\frac{1}{\pi} \iint_A Q_Q \ln\left(\frac{1}{r}\right) dA + \frac{1}{2\pi} \oint_L \ln\left(\frac{1}{r}\right) \frac{\partial P}{\partial n} dL - \frac{1}{2\pi} \oint_L P \frac{\partial \ln\left(\frac{1}{r}\right)}{\partial n} dL \quad (4.1)$$

Each term in this equation is used by VerFlow-V.01 to interpret OpenFoam “data” both quantitatively as a calculation and *qualitatively* as a graphical image. For simplicity each term in Equation 4.1 is assigned a label that is used in the VerFlow-V.01 program interface and defined as follows:

$$(Q) = -\frac{1}{\pi} \iint_A Q_Q \ln\left(\frac{1}{r}\right) dA \quad (4.2)$$

$$(CI_1) = \frac{1}{2\pi} \oint_L \ln\left(\frac{1}{r}\right) \frac{\partial P}{\partial n} dL \quad (4.3)$$

$$(CI_2) = -\frac{1}{2\pi} \oint_L P \frac{\partial \ln\left(\frac{1}{r}\right)}{\partial n} dL \quad (4.4)$$

See also Equations 3.85 to 3.131 in Sections 3.4.1 and 3.4.2 for more detail.

The surface integral is labeled as  $(Q)$  in equation 4.2 and assigned to the *Integration* tab in the VerFlow-V.01 program interface in Figure 4.19 (see also Figure 4.21 where interface is enlarged), where different values are compared under the *Integration* tab, and  $(Q)$  is a red horizontal line. The value of  $(Q)$  is written next to the title “Surface integral  $(Q)$  (red)”, again see Figure 4.19 and 4.21.

The contour integrals  $(CI_1)$  and  $(CI_2)$  have different values for each part of the boundary. The results are written in a table below the title “contour integral 1 (black) 2 (gray)”. In the image portion of Figure 4.19, these integrals are drawn in black or gray under the correspondent boundary identifier: left (L), right (R), top (T), bottom (B) and cylinder (Cy), see also Figure 4.21 for this interface layout.

The reference value for the pressure  $(P)$  at a point comes directly from the OpenFoam pressure file. This reference value is presented as a blue horizontal line below  $(P)$  under the *Integration* tab. A light orange horizontal long line represents the zero pressure. The predicted value for the pressure is shown also in blue below the  $(+)$  label to the right of label  $(P)$ .

The solution for the pressure at any arbitrary point requires a domain where the second invariant of the velocity gradient  $Q$  is known over the whole region and where the pressure and pressure gradients are also known along the boundaries.

We have solved the problem for two different domains. The first is a *rectangular* domain chosen arbitrarily inside the block 4 (the channel to the right of the cylinder). The second is the *entire* flow domain.

The color legend at the left in Figure 4.19 can be used also for Figure 4.20 and 4.22. The same time,  $0.4[s]$ , is used for all these figures. The primary variable labeled  $(V.)$  represents in the figure the second invariant of the velocity gradient,  $Q$ , and because of the transparency setting effects, the contour lines of this variable appears orange or brown. In this same figure the overlapped variable  $(O.V.)$  is the pressure per unit density,  $P$ , where contour lines are drawn with a black color.

### 4.2.2 Rectangular domains

In order to determine the pressure at an arbitrary point using Equation 4.1, a point within the rectangular domain must be selected. The rectangular domain can itself be a subdomain limited to a region within block 4. See Section 3.4.1 for definitions and equations. With VerFlow-V.01 the user can select any arbitrary rectangular region downstream of the cylinder for analysis in block 4.

The example here shows an instantaneous frame in Figure 4.19. An animation is linked to the small figure below and to the Animation 4.2 title.

White, partially transparent lines, inside the rectangle, connect the selected point to each point on the boundary.

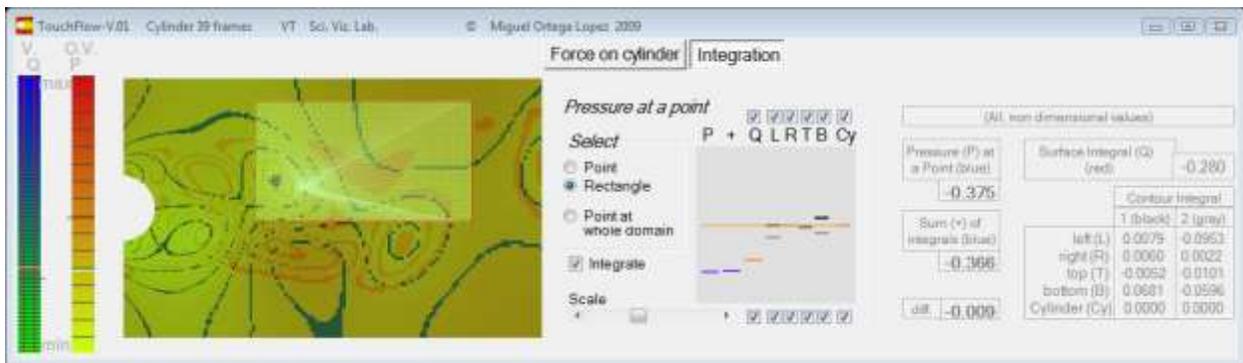


Figure 4.19 Pressure predicted at an arbitrary point in a rectangular domain at time 0.4(s)



Animation 4.2 [Pressure at a point on a rectangular domain](#)

For this particular example the reference pressure is -0.375 (calculated by the OpenFoam code) and the predicted pressure from the Poisson integration in VerFlow-V.01 is -0.366. These values differ by 0.009 (2.4%). Note also that the contributions from the right and top boundaries are negligible. The contour integral ( $CI_2$ ) at the left boundary has an important contribution predicting the pressure at this particular time, rectangle, and point. The two contour integrals at the bottom are not negligible by themselves, but their combined action is not as significant as

other terms, because they have similar magnitudes but opposite signs. The surface integral yields the highest contribution.

In summary for this example, the pressure is the result of the contributions from the surface integral and from the contour integral ( $CI_2$ ) at the left boundary.

For the largest rectangle permitted by VerFlow-V.01, e.g. the entire block 4, and with the same arbitrary point and time, a new value is predicted for the pressure. See Figures 4.20 and 4.21.

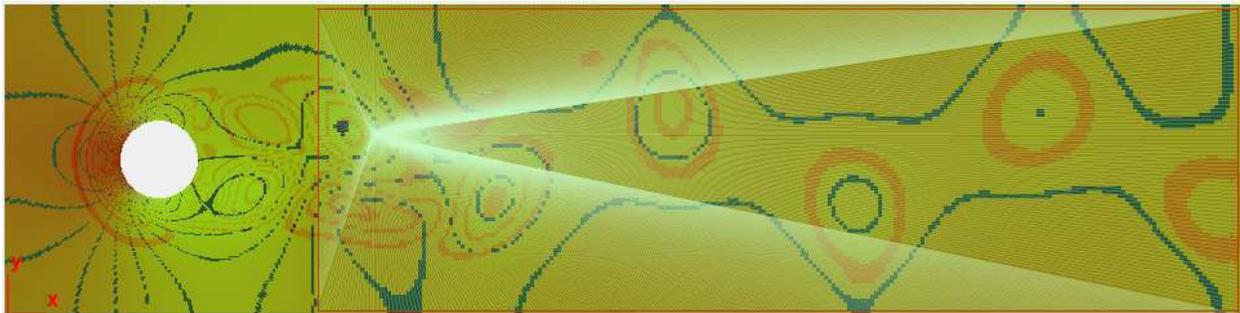


Figure 4.20 Arbitrary point in the biggest rectangle at block 4, see color legend in Figure 4.19.

Again the pressure has contributions both from the second invariant  $Q$  and from the contour integral  $CI_2$  at the left boundary. This is expected for a gross calculation, all contributions from other boundaries can be ignored. This is possible because at these boundaries, right, top and bottom, the pressure gradient has small magnitudes in the normal direction and for integral  $CI_2$  the point is far from the boundaries.

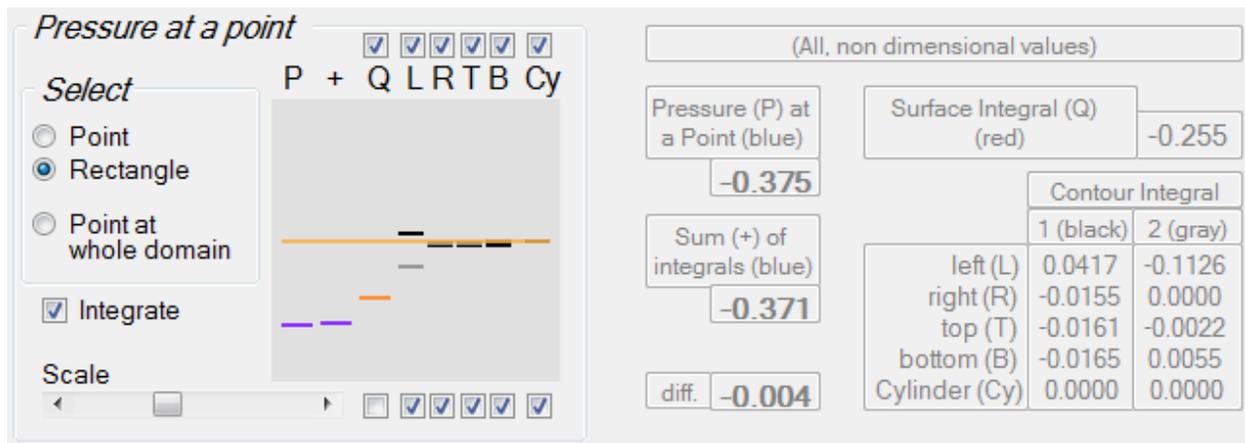


Figure 4.21 Pressure at an arbitrary point for the biggest rectangle in block 4.

### 4.2.3 Entire domain

Once again the same arbitrary point is selected, but now the entire domain is included, blocks 0 to 4. See Section 3.4.2 for definitions and equations. The results can be seen in Figures 4.22 and 4.23 and in Animation 4.3.

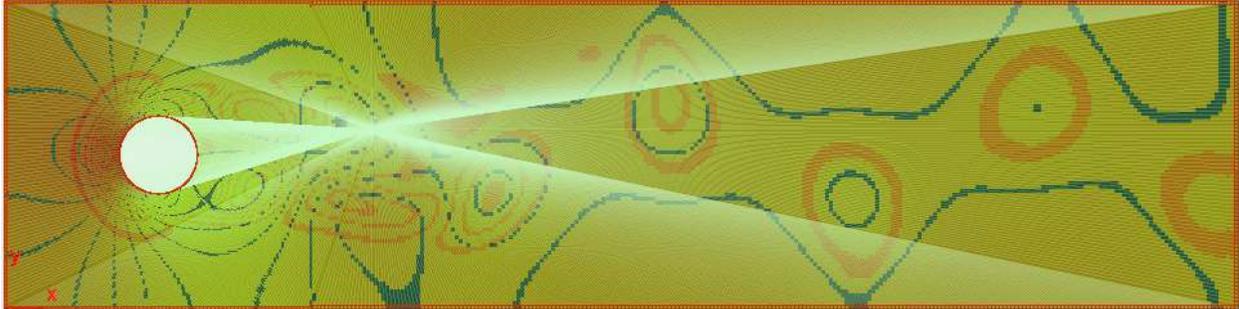


Figure 4.22 Arbitrary point in the entire domain, see color legend in Figure 4.19.

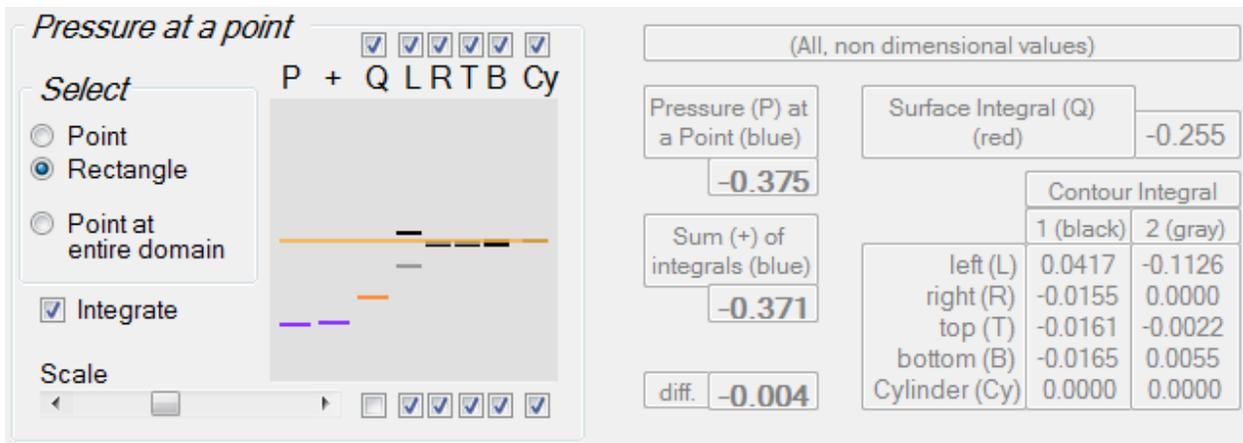


Figure 4.23 Pressure at an arbitrary point for the entire domain



Animation 4.3 [Pressure at a point on the entire domain](#)

Although the contributions from some boundaries are small, when acting together the result becomes important and the prediction of the pressure,  $P_p$ , is very close to the reference value obtained from the OpenFoam simulation. In this case the absolute difference is 0.005. However,

again the most significant contribution, 77%, comes from the surface integral  $\left[-\frac{1}{\pi} \iint_A Q_Q \ln\left(\frac{1}{r}\right) dA\right]$ , equation 4.2.

For this case, the contour integral at the cylinder boundary contributes more than the other boundaries in the prediction of the pressure.

The Animation 4.3 shows how all variables contribute to the pressure changes as the flow moves in a repeated cycle.

Predicted pressures,  $P_p$ , when choosing different points at different times, and selecting different domains, are similar to the pressure per unit density from the OpenFoam simulation, i.e. the difference between both is always less than 6% of the maximum pressure and in most of the cases less than 1% of the maximum pressure. The *quality* of our results is given by the correspondence between the two blue lines, one representing the pressure from OpenFoam and the other the predicted pressure,  $P_p$ . The difference between the two is also given numerically (quantitatively). Both pressures are in good agreement for the large number of tests studied here.

#### ***4.2.4 Filtered domain: envisioning OpenFoam “data” qualitatively***

OpenFoam “data” is filtered for two variables shown in the color legend bars of VerFlow-V.01, see the left side of Figure 4.19. The first variable has a title “V.” and the second “O.V.” which signifies an “Overlapped Variable”. Although both variables, “V.” and “O.V.”, support color legend filters, only the first variable selects the sub-domain. This sub-domain intersection with the domain (rectangular subdomain or entire region) gives the region in which the surface integral given in Equation 4.2 is calculated.

VerFlow-V.01 allows the user to filter the horizontal and vertical velocities, velocity magnitude, pressure, vorticity, vorticity magnitude, second invariant of the velocity gradient and even drag and lift. The filter is accomplished by visually selecting minimum and maximum limits on the color legend bar.

The combination of the capabilities of VerFlow-V.01 results in an extremely useful package that can be customized interactively as the user interprets multiparameter results.

In the following example the first variable, “V.”, is selected as the second invariant of the velocity gradient. A closer view of the *qualitative* and quantitative results of P, +, Q, L, R, T and B in Figure 4.19 is presented in Figure 4.24 where the color legend bar plays a role in the selection of a filtered region in which the surface integral is calculated. In Figure 4.24, the green-blue color legend bar, “V.”  $Q$ , acts as a filtered variable, which appears at the left with two vertical red lines indicating all data are visible and no filtering exists. The overlapped variable, “O.V.” pressure  $P$ , remains as shown in Figure 4.19 where the yellow-red color legend bar is not used. The colors in the visible region of the flow are the result of the combination of both variables, “V.” and “O.V.”, with transparency effects. The shaded region enclosed in the rectangle is the intersected area between the region inside the rectangle and all visible data that was not filtered. In this example the quantitative results at the right show an important contribution of the pressure at the point in red, from the surface integral ( $Q$ ) given by Equation 4.2. In this fashion the user can customize variables of interest and *qualitatively* evaluate the OpenFoam data with respect to both variables, “V.” and “O.V.”, simultaneously.

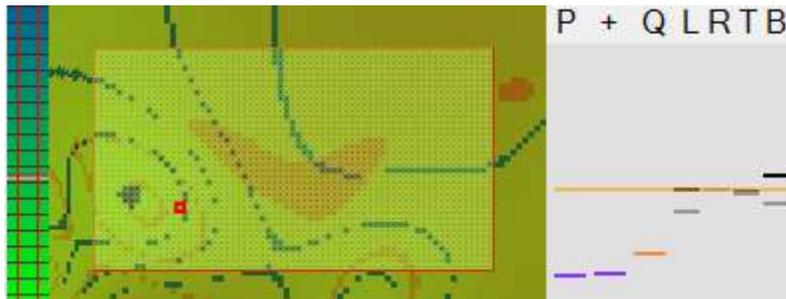


Figure 4.24 Pressure at a point for non-filtered data (rectangle domain)

Recall that ( $P$ ) is the reference pressure at the selected point from OpenFoam, (+) is the sum of all contributions,  $P_p$ , and ( $Q$ ) is the surface integral in the intersected domain where ( $L$ ), ( $R$ ), ( $T$ ) and ( $B$ ) are the contour integrals at the left, right, top and bottom sides respectively.

Here the filter shown in the green-blue color legend bar in Figure 4.25 defines new limits where the variable is visible. Although a smaller region in the color legend bar is selected, the visible region is reduced in the clear zones only (see Figure 4.25). The reference line (zero), inside the selected region, reveals that the magnitude of the variable in the visible region is close to zero. The surface integral ( $Q$ ) given by Equation 4.2 is realized in the intersected shaded region. Note

that the filter only affects the terms (Q) and (+) in the right part of Figures 4.24 and 4.25. The magnitude of this integral is also observed to drop drastically (from 0.280 to 0.062).



Figure 4.25 Pressure at a point for filtered data (rectangle domain)

It would seem reasonable that the regions with small values in the second invariant of the velocity gradient would not contribute to the prediction of pressure, however this assumption is incorrect.



Figure 4.26 Pressure at a point for filtered data (entire domain)

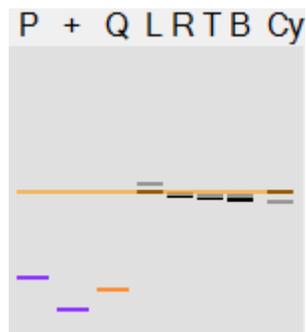


Figure 4.27 Pressure at a point for filtered data (entire domain) - results

Calculations at the same point are shown in Figure 4.27 but for the entire domain shown in Figure 4.26 and using the same color legend filter as in Figure 4.25. No shaded region is shown in Figure 4.26 to realize a cleaner and clearer picture of the entire domain, e.g. blocks 0 to 4.

The quantitative results for this filtered region are shown in Figure 4.27.

These results are compared with Figure 4.23. The surface integral in Figure 4.23 and Figure 4.27 has magnitudes of 0.287 and 0.433 respectively. The resultant magnitude from filtered data is 150% of the resultant magnitude of non-filtered data. Although only small values of the second invariant of the velocity gradient contribute to the surface integral in Figure 4.27 this surface integral magnitude exceeds the reference magnitude for pressure. This means that the prediction of pressure is the result of the summation of large positive and negative surface integral components coming from both large and small magnitudes of the second invariant of the velocity gradient regions.

### 4.3. Point effect on the cylinder boundary

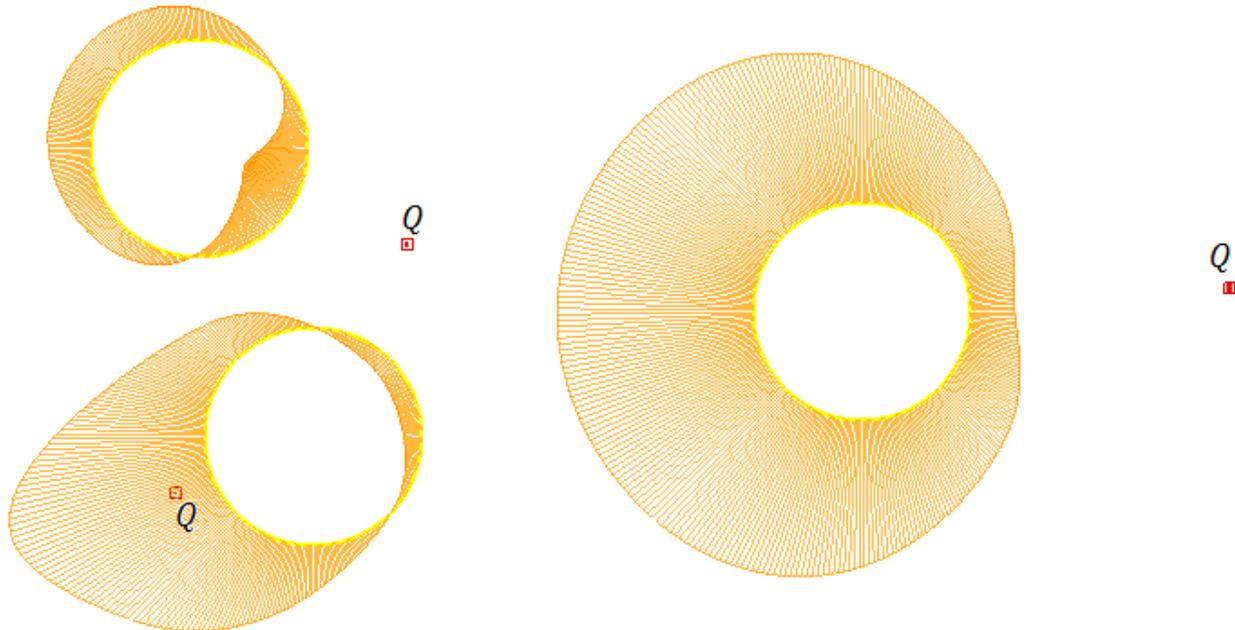
The surface integral,  $(Q)$ , in equation 3.84a (or 4.2, which is the same) can be understood as the sum of differential contributions to the pressure at a point  $P$  from the second invariant of the velocity gradient acting at an infinite number of points  $Q$  in the domain. The differential pressure contribution,  $dP_P = -\frac{1}{\pi} Q_Q \ln\left(\frac{1}{r}\right) dA$ , converted to finite difference form, is given by  $\Delta P_P = -\frac{1}{\pi} Q_Q \ln\left(\frac{1}{r}\right) \Delta A$ , which is the contribution to the pressure at point  $P$ , from each point  $Q$  in the mesh. Intentionally, the point in which the second invariant of the velocity gradient,  $Q$ , is considered was labeled as  $Q$ , and, the point in which the pressure contribution,  $\Delta P$ , is calculated was labeled as  $P$  for all points in each cell along the cylinder boundary in the following example.

Instead of looking at one point  $P$  for all points  $Q$  in the domain, here the objective is to look to one point  $Q$  and several points  $P$  in the domain. These points  $P$  are neighboring cells located along the cylinder boundary. In other words, the contributions to the pressure at each point on the cylinder boundary from the second invariant of the velocity gradient acting at point  $Q$  result in a pressure distribution on the cylinder boundary due to the second invariant of the velocity

gradient at that point  $Q$ . These pressure distributions are dimensionless in VerFlow-V.01 and they are illustrated in Figures 4.28 and 4.29, using an appropriate scale magnification.

Figure 4.28 shows three independent examples of points  $Q$  (in red) and its effect on the pressure along the cylinder boundary. The yellow lines represent the cylinder boundary. The orange radial lines are the pressure contributions from point  $Q$ . Positive contributions point away from the circle into the external region along a radial line. Negative contributions are drawn from the cylinder boundary pointing inward. Point  $Q$  for each example in Figure 4.28 is close to but not touching on the cylinder boundary.

In general there are very different and unbalanced distributions on the cylinder from the different locations within the flow field.



**Figure 4.28 Contribution of red points  $Q$  to the pressure at each cell along the cylinder boundary**

Figure 4.29 shows the contributions of  $Q$  at points very close to the cylinder (third row from the cylinder boundary).

Small variations in the location of point  $Q$  sometimes produce rapid variation in the surface pressure distribution. All seven examples in Figure 4.29 show the same trend. In three of the examples, those with large negative values, the lines point inside the circle and crossover exactly

at the center of the circle to lines pointing outward. These trends contribute to our *qualitative* understanding of property gradients embedded in the data.

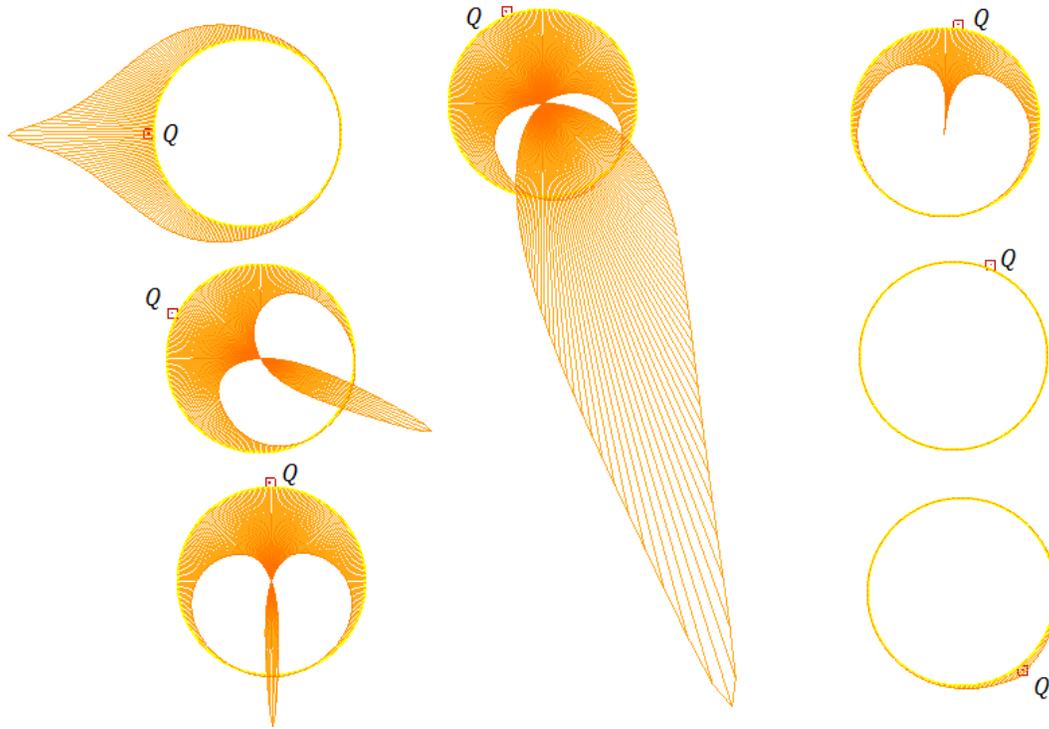


Figure 4.29 Effect of points  $Q$  close to the cylinder on pressure at its boundary

For all examples shown in Figure 4.29 the contributions,  $\Delta P_P = -\frac{1}{\pi} Q_Q \ln\left(\frac{1}{r}\right) \Delta A$ , are zero on the opposite side of point  $Q$ . The reason for this comes from the factor  $\ln\left(\frac{1}{r}\right)$ . The distance  $r$  in the equation for  $\Delta P_P$  is measured between point  $Q$  and point  $P$ . Since this equation is dimensionless,  $r$  is the number of diameters between the two points (the characteristic length is the diameter). When  $P$  and  $Q$  are on opposite sides of the cylinder  $1/r$  is just 1, and its natural logarithm is zero. As a consequence the whole term is zero.

Similar results can be seen for points  $P$  along the cylinder boundary when  $Q$  is located approximately at one diameter from the cylinder boundary, e.g. see two examples at the left in Figure 4.28.

#### 4.4. Contributions from $Q$ to the drag and lift forces

In the previous section, it was found that the velocity field at point  $Q$  affects the pressure at each point on the cylinder boundary. This section presents the individual contributions to drag and lift coefficients from every cell in the grid. To realize drag and lift contributions from an arbitrary cell in the domain where point  $Q$  is located, we use the equations in sections 3.3.1 and 3.3.2, and the use of pressure distributions shown in section 4.3. These calculations are implemented in the VerFlow-V.01 code.

Pressure distributions along the cylinder boundary are calculated for every cell in the grid (point  $Q$ ) domain and the drag and lift are stored separately as new properties in the domain associated with the location and area of the cells corresponding to  $Q$  points. Although drag and lift calculated from the velocity field could be easily normalized to a standard size of the grid cell, the contributions are not normalized since this will only maximize even more the effect in the region close to the cylinder, but the effect is of course considered in the discussion that follows. Note, these drag and lift contributions are exclusively due to pressure distributions from  $Q$  and they are not due to viscous effects. Note also  $Q_d$  is selected for both “V.” and “O.V.” variables.

##### 4.4.1 Contributions to the drag

The result for drag contributions in Figure 4.30 is shown in a close up view in Figure 4.31, where limits are set. The color legend (filtered domain) highlights drag regions: positive in blue and negative in orange.

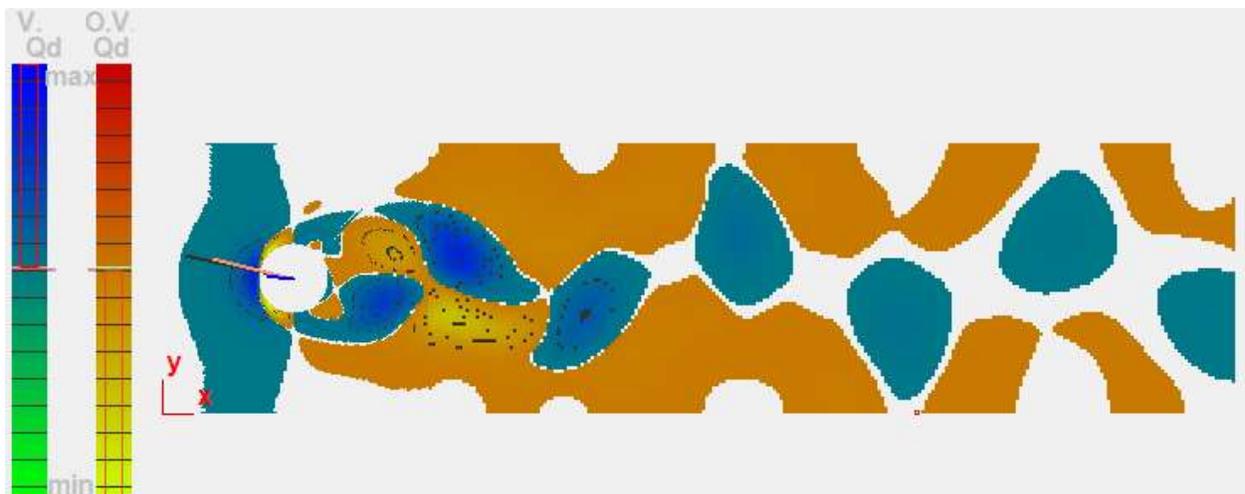


Figure 4.30 Contribution of each point in the domain to the cylinder drag (Min: Yellow, Max: Blue, zero: White)

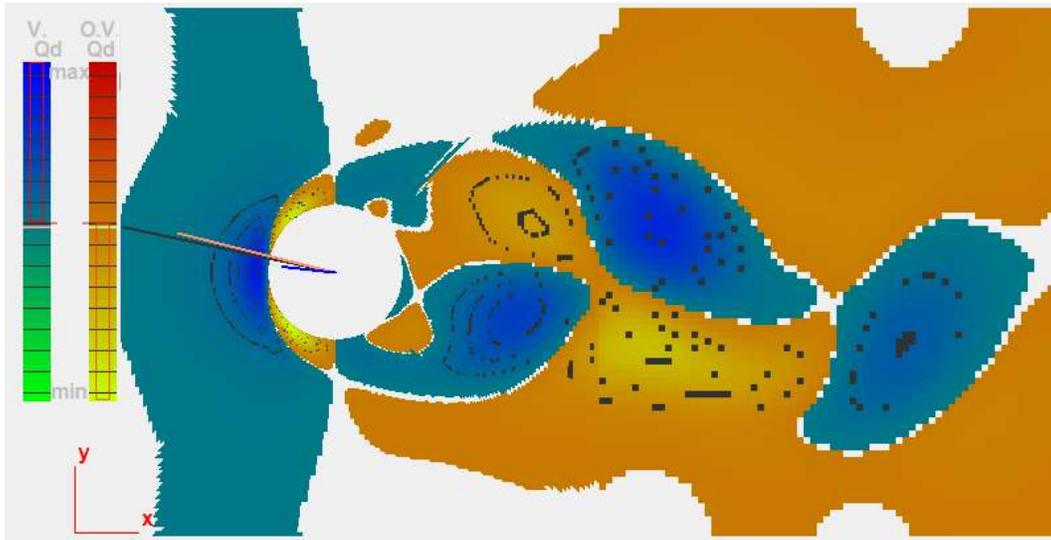
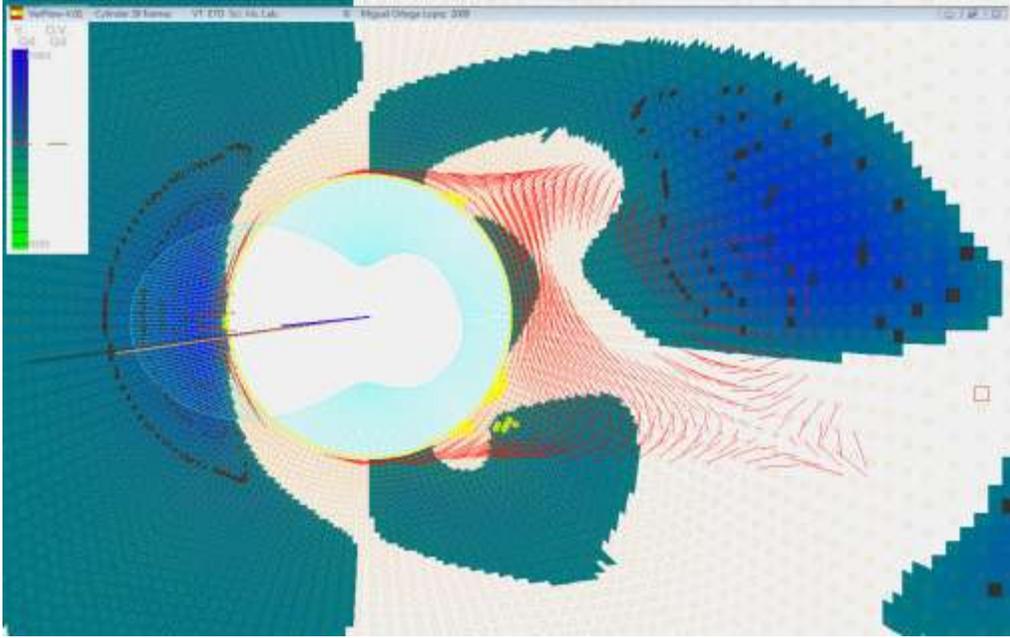


Figure 4.31 Closer view of drag contributions to the cylinder from the second invariant of the velocity gradient field

Limits on the color legend are adjusted to show the white zones as zero drag. The higher positive contributions to drag are located just in front of the cylinder and the higher negatives are at the two frontal lobes. Behind the cylinder, positive and negative regions appear alternated and decrease in intensity rapidly with downstream distance. The distribution of drag in space is similar to the distribution of  $Q$ , where positive regions travel with the flow along the middle of the channel which identifies vortices.

Vortices or eddies are associated with positive contributions to the cylinder drag downstream, while convergence and stream zones are responsible for a reduced drag (see discussion in section 4.5). Regions that contribute to diminish the drag are associated with negative values that tend to travel along relatively high-pressure zones near walls. Around the cylinder there are eight to ten zones of alternating positive and negative drag contributions. These zones are studied in more detail in the following discussion and shown in Figures 4.32 and 4.33. Looking at the contour lines over the positive zones at the right of the cylinder, it is clear that positive zones reach larger magnitudes than the negative zones (comparison is done at similar radii and on similar cells). Although the negative zones cover a larger area, this increased area is not significant since the gradient is clearly very small in the zones, consequently the magnitude of this variable,  $Q_d$ , is nearly zero but negative.



**Figure 4.32 Envisioning multiple instantaneous filtered information over the positive drag contributions from  $Q$ .**

Figure 4.32 shows two simultaneous selected information: (1) the positive drag contributions from the second invariant of the velocity gradient, and (2) the velocity vectors relatively small in magnitude. Both are explained as follows. Local contributions to the drag coefficients due to the second invariant at every point in the domain, are indicated by the label “ $Qd$ ” in the color legend bar at the left. This variable,  $Qd$ , has been filtered and only the positive region is displayed. Black lines are the contour lines of this variable. Instantaneous forces are drawn as straight lines positioned at the cylinder center (pressure, viscous and total forces) and pressure acting on the cylinder is also represented in cyan color (see Section 4.1.3). Red lines are the filtered velocity field, so that only relatively small velocities are drawn to scale in the domain considering the static frame of reference (see section 4.5.3). Yellow points signify where the smallest instantaneous velocities exist and identify stagnation points and sometimes eddy centers and convergence zones (see also section 4.5.3)

At the specific instant of time shown in Figure 4.32, there are four clear positive regions for drag contributions along the cylinder boundary. The main source of drag is located at the left of the cylinder, a small zone close to the first stagnation point (left arc on the cylinder boundary). Black contour lines appear as points that get closer and closer indicating high gradients near the stagnation point. Already noted in the discussion of Figure 4.31, positive zones have a stronger effect on the drag than their negative counterparts. The positive zone directly associated with a

clockwise eddy in Figure 4.32, has a stronger effect than the negative zone associated with the “four way” convergence zone.

In Figure 4.33, considering that the contributions to the drag were not normalized by the area of the cell, the effect on drag from small eddies is small but comparable to the effect on the drag from large eddies.

The previous analysis of drag contributions from the pressure on the cylinder boundary presented in Section 4.1.1, is associated here with regions near the cylinder. Looking at Figure 4.33, the stronger region of positive contributions to the drag is in fact located in block 3 and block 1 has also important positive contributions while blocks 0 and 2 could even be ignored. Animation 4.4 reveals that the strongly positive region at the left of the cylinder is slightly oscillating, e.g. up and down movement, which generates the oscillatory effect shown in Figure 4.13 for arc 3.

Looking to Figure 4.13, the oscillation on the drag due to the pressure on arc 1 attenuates the oscillation on the drag due to the pressure on arc 3 to give the drag shown in Figure 4.14 (center), which will be even more attenuated by the oscillation on the drag due to arcs 0 and 2, see Figure 4.14 at the right. The superposition of the oscillations is shown in Figure 4.14 at the left. A complex behavior in the pressure contributions is observed on the neighboring cells around the cylinder along arc 1 which is shown as an image sequence in Figure 4.33 and also as an Animation 4.4. Along arc 1, only one positive region persists and at two specific instants in time an additional positive region appears at times 0.00[s] and 1.00[s] in Figure 4.33. The contributions to the drag in these regions close to arc 1 are also very close to zero compared with those regions close to arc 3. These observations suggest that the behavior on the cells close to arc 1 generate the oscillations with a doubled shedding frequency in the drag, which attenuates the effect generated by arc 3.



Animation 4.4 [Contributions of  \$Q\$  to the drag](#)

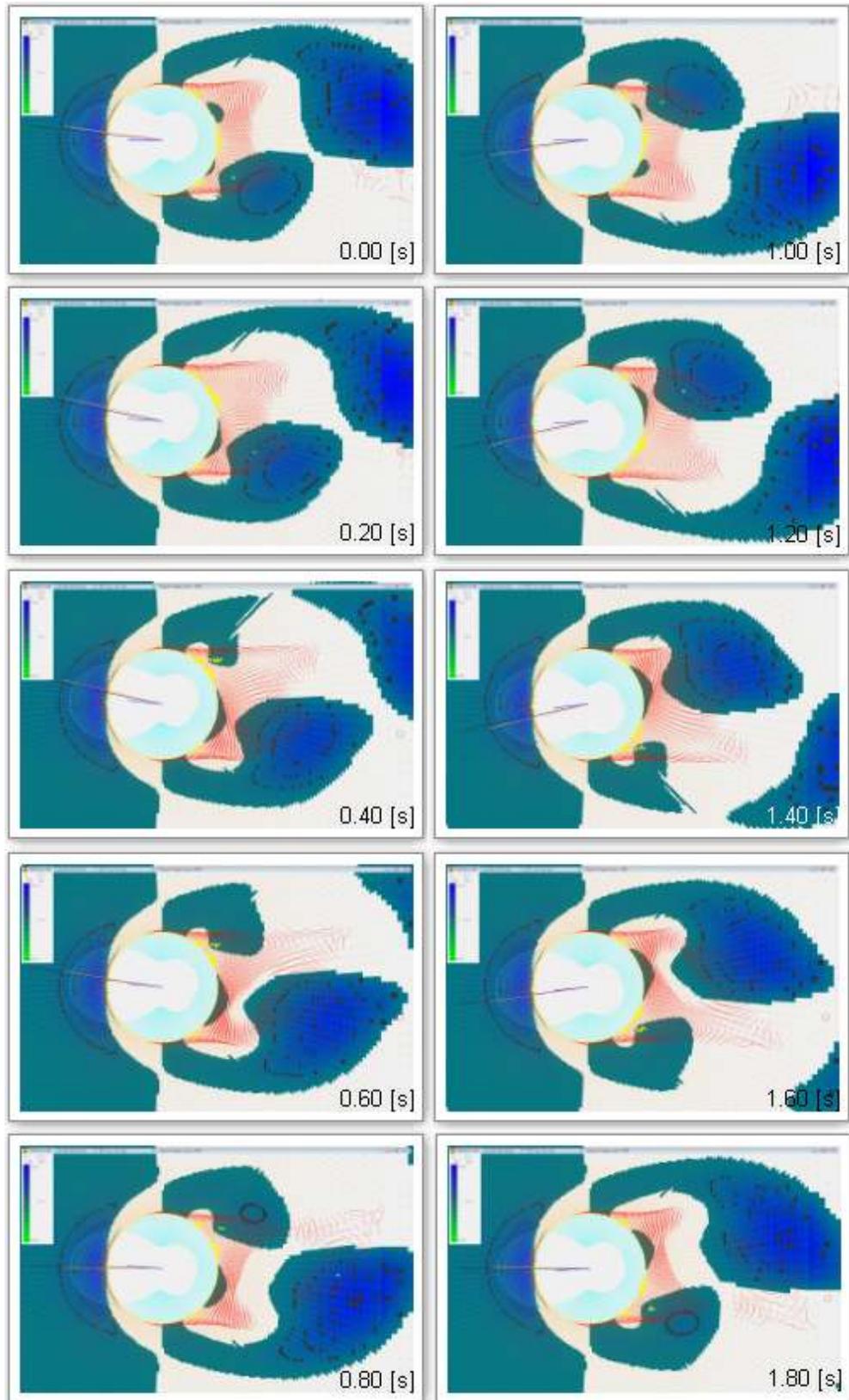


Figure 4.33 Evolution in time of positive contributions of  $Q$  to the drag

#### 4.4.2 Contributions to the lift

Lift contributions are mapped in Figure 4.34, a detailed enlarged view is shown in Figure 4.35.



Figure 4.34 Contribution of each point in the domain to the cylinder lift (Min: Yellow, Max: Blue, zero: White)

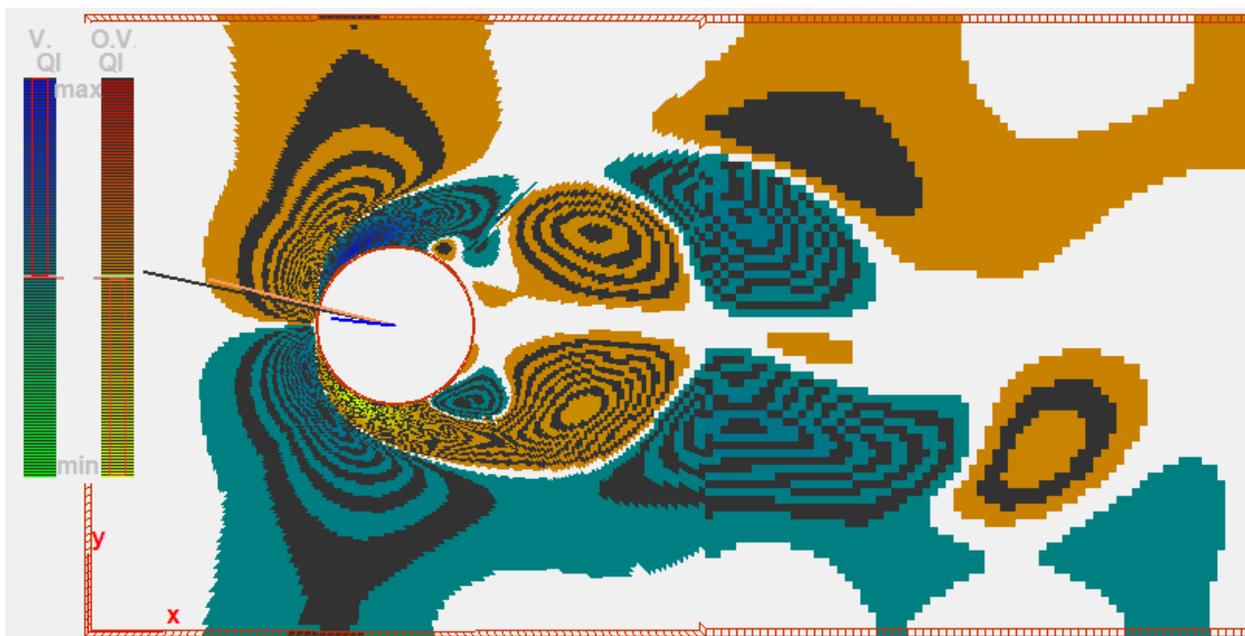
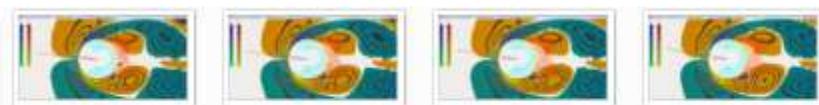


Figure 4.35 Closer view of lift contributions to the cylinder from the second invariant of the velocity gradient field



Animation 4.5 [Contributions of  \$Q\$  to the lift](#)

The large white region at the right downstream is associated with zero values, which indicates the lift decays very fast with distance from the cylinder. The strongest regions are located very close to the cylinder toward the top-left and top-right zones. All regions present a complex oscillatory behavior in time as can be seen in Animation 4.5.

The white region (zero contribution to the lift) appears at some specific locations within the entire region: inlet, walls and in a region emerging from the cylinder at its right boundary. Alternating positive and negative zones in the front are coupled. Behind the cylinder, couples of both negative and positive regions grow alternately traveling with the flow downstream where they soon disappear. When growing, these couples are unbalanced which is associated with a growing vortex (see Animation 4.5) and lift on the cylinder. For example, in Figure 4.35, there are two negative regions of unequal strength traveling downstream approximately one diameter far from the cylinder, a counter clockwise eddy is located at the top left zone of the negative region (see Animation 4.5), which has more contour lines, is strong, and is located below the other negative coupled region; this is directly associated with the negative lift on the cylinder at that instant of time. Both negative regions can be compared since the areas of the cells are also comparable, although the contributions are not normalized with respect to a reference area of the cells. When the coupled regions have traveled downstream to a location two diameters far from the cylinder, the number of contour lines had reached equal values where there is no longer a stronger region in that couple.

What is described for negative couples also exists for positive couples, which are associated with clockwise vortices and positive lift.

#### **4.5. Convergence and eddy zones**

In an effort to better understand the detail of this flow, we take the idea of eddies, streams and convergence zones discussed by Hunt (Hunt, Wray, & Moin, 1988) where two frames of reference are considered. The first is a moving frame traveling at the mean velocity of the fluid (which is essentially the inlet velocity) and the second, is a static frame. Special tools were implemented in VerFlow-V.01 to help the user to *qualitatively* envision these two frames of reference.

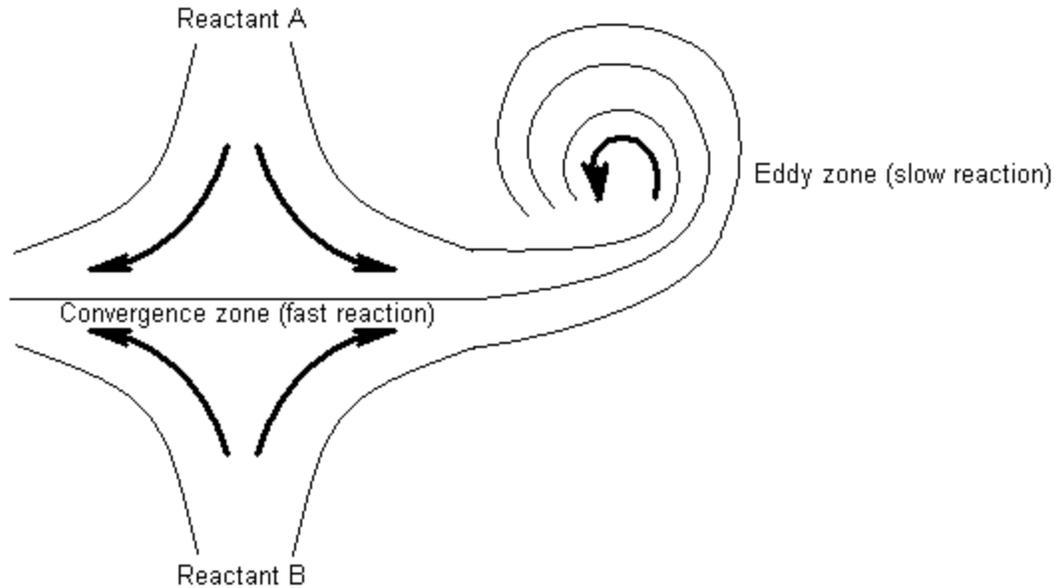


Figure 4.36 Fluid zones adapted from Hunt paper (Hunt, Wray, & Moin, 1988)

#### 4.5.1 Useful zones defined by Hunt

Convergence, eddy, and stream zones were defined by Hunt (Hunt, Wray, & Moin, 1988) (see Figure 4.36 adapted from Hunt). In this figure, Hunt shows how two streams carrying reacting species find each other in a convergence zone and contribute to the formation of an eddy.

Although those zones were developed for turbulent flows, they also exist in the periodic laminar flow around a cylinder.

#### 4.5.2 Moving frame

Consider a frame that is moving to the right at the same mean velocity or inlet velocity. An equivalent point of view occurs when the frame of reference is fixed, the mean velocity of the flow is zero, and the cylinder is moving to the left at the original mean velocity or inlet velocity. Based on the velocity vectors drawn for each cell by VerFlow-V.01, streamlines are shown as narrow lines in Figure 4.37 for this relative velocity field. The shaded circle in Figure 4.37 is the cylinder.

The direction of streamlines is consistent over the whole region. Streams with opposite directions can meet each other only at convergence zones, which are identified and studied here in more detail.

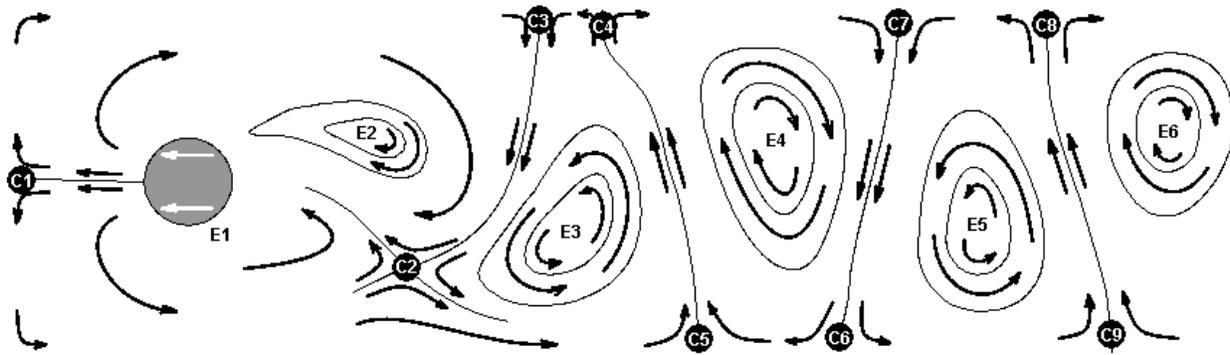


Figure 4.37 Zones in the flow around a cylinder for a frame of reference moving at the mean velocity

At a specific instant in time, flow around a cylinder is composed of convergent, eddy and stream zones which appear in the flow around a cylinder for that instant.

In the general case, a “four way” convergence zone appears where two streams converge on each other in opposite directions and both streams are divided. These are also termed saddle points by some researchers. As a result, two outlet streams go away in opposite directions from the “four way” convergence zones. “Three way” convergence zones are defined when one of the four streams is replaced by a wall.



Figure 4.38 Convergence zone C1 at the inlet (moving frame)

At the left (inlet side), there exists a “three way” convergence zone C1 (Figure 4.37 and yellow points in Figure 4.38). The red lines in Figure 4.38 are scaled velocity vectors with respect to the moving reference frame in zones where the flow has slow velocities. The higher velocities have been filtered out in VerFlow-V.01 to avoid overlap. Small circles identify the origin of each

velocity vector in a cell. The fluid is pushed around the moving cylinder and divided into two streams at the inlet side, one passes over the cylinder and the other below it. The yellow circle indicates the presence of a stagnation zone.

The fluid flow below the cylinder forms into an eddy E1 (see Figures 4.37 and 4.39). This eddy has a “cat’s eye” shape as defined by Hunt (Hunt, Wray, & Moin, 1988), and it is the result of the displacement and deformation of a circular eddy behind the cylinder (this can be seen in the non-moving frame of reference in section 4.5.3). Displacement and deformation of eddy and convergence zones are highlighted when the frame of reference is changed.

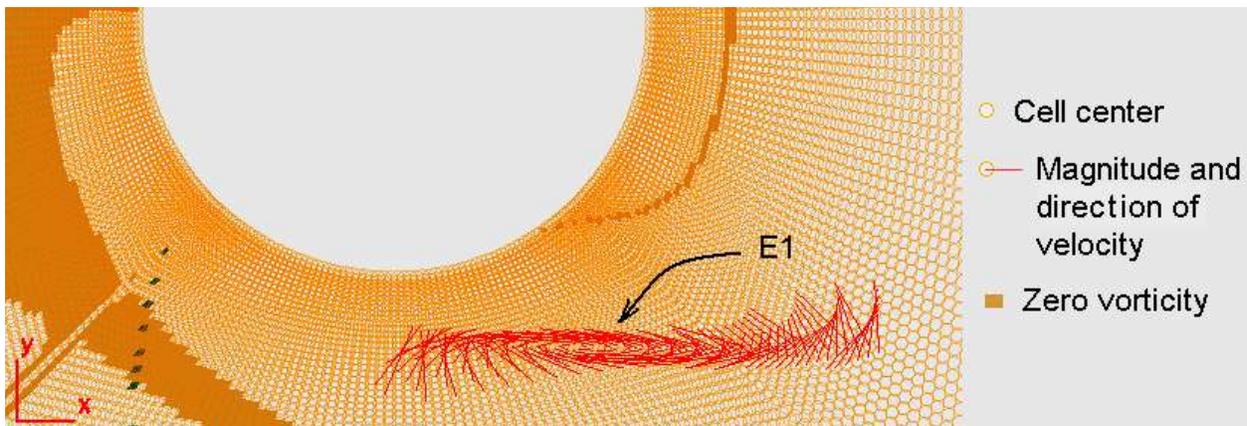


Figure 4.39 Counter clockwise eddy E1 (moving frame)

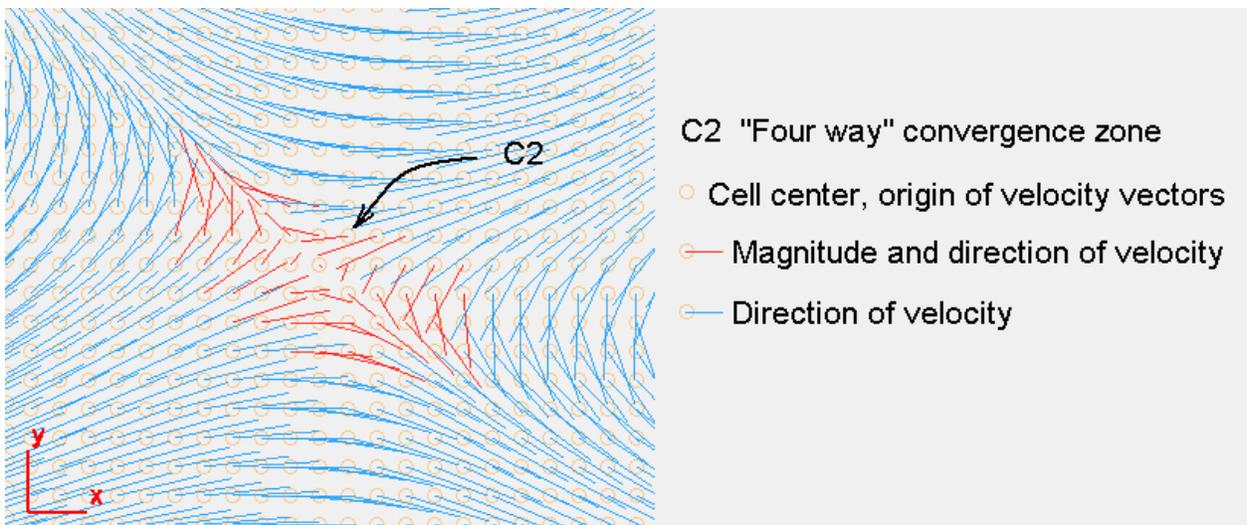


Figure 4.40 Convergence zone C2 (moving frame)

Only part of the fluid that flows below the cylinder takes a path about E1. The flow first finds a “four way” convergence zone C2. Figure 4.40 shows zone C2, the cyan lines shows only the direction of high velocity vectors. If the flow moves to the left at C2, it encloses E1. If the flow moves to the right, it encloses E3 and can return again at C2 behind the cylinder. Note that both E1 and E3 move in a counter clockwise direction.

The flow going over the cylinder is impeded to move to the right because of the “three way” convergent zone C3 at the top wall. The size of the clockwise eddy E2 is increased (see Figure 4.41) as the flow is pulled over the top of the cylinder.



Figure 4.41 eddy zone E2 (moving reference)

When the eddy zone E2 moves far enough, E2 detaches from the cylinder while another appears simultaneously, but with a smaller clockwise eddy and a “four way” convergent zone between E2 and the small eddy. This small eddy is established on the same side of E2, E4 and E6 but close to the cylinder.

Eddy E2 grows rapidly and pushes C2 down until it touches the bottom wall where C2 is transformed into two “three way” convergent zones similar to C5 and C6 but not shown in Figure 4.37 (C5 and C6 were formed when E4 pushed down another “four way” convergent zone). One of these convergent zones (the equivalent to C6) will define an independent region of fluid enclosing E3, which travels at the mean velocity to the right with the moving frame.

Similarly, eddies E4, E5 and E6 are separated regions of fluid traveling at the same inlet velocity. This is envisioned in the moving frame of reference. These eddies are “wrapped” while

growing. When the “four way” convergence zone appears, they basically stop growing and become trapped.

A half of one cycle has been described. The sequence is repeated symmetrically to complete one shedding cycle.

The eddy center and convergence points act as stagnation points, e.g. points C3, C4, C5, C6, C7, C8 and C9. The effect of the wall is to convert a “four way” convergence zone into two “three way” convergence zones. The zones around C3 and C4 are an example of this. In an infinite domain, a “four way” convergence zone would exist instead of C3 and C4 and the location of this zone would not be limited by the walls. So C3, C4, C5, C6, C7, C8 and C9 are the result of the confined walls in the computational domain.

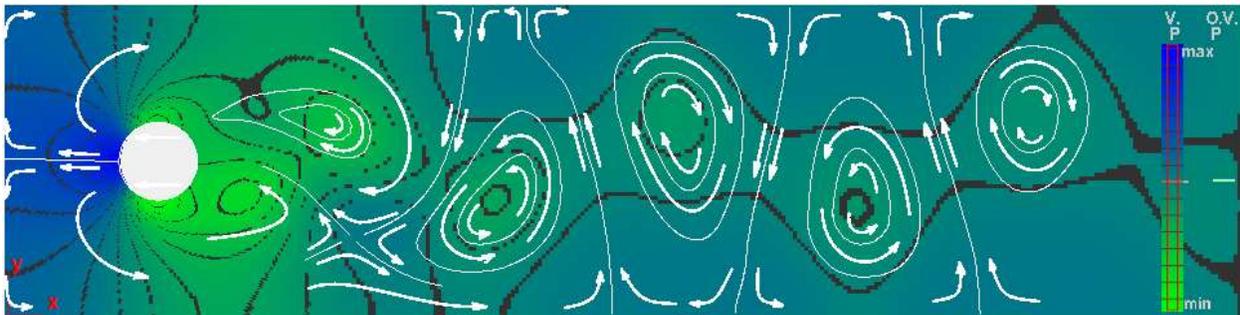


Figure 4.42 Streamlines (moving frame) over pressure (green: Min, blue: Max)

Figure 4.42 shows the instantaneous streamlines in the moving frame. These lines are drawn over the pressure field. Note that eddies’ centers coincide with the lowest pressure for each region of the fluid. Convergence zones occur at high-pressure zones.

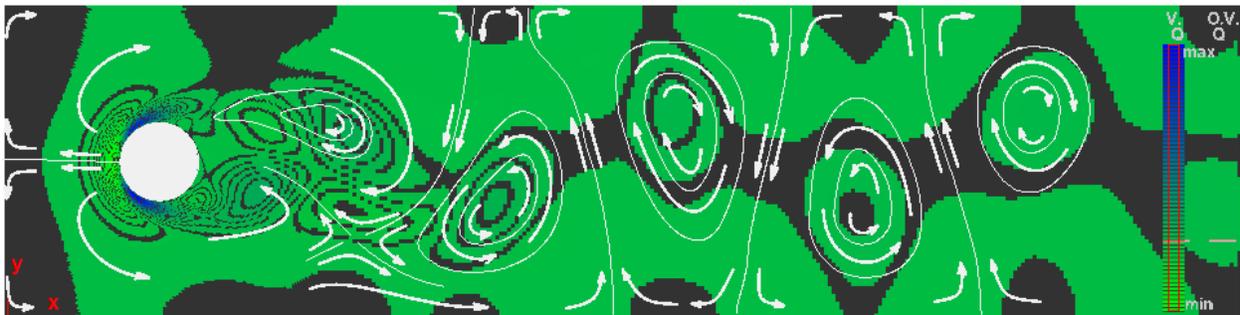


Figure 4.43 Streamlines (moving frame) over second invariant of velocity gradient (green: Min, blue: Max)

“Four way” convergence zones originate where the second invariant of the velocity gradient is negative, see Figure 4.43, while eddies concentrate at positive values of the second invariant of the velocity gradient. Stream zones, in the moving frame, do not coincide with regions of positive  $Q$ .

Figure 4.44 shows the streamlines drawn over the vorticity field with respect to a moving reference frame. Negative vorticity appears in the eddies at the top with a clockwise rotation while positive vorticity appears at the bottom with counter clockwise rotation. The location of the “four way” convergence zone can also be identified looking at the vorticity field when the vortices are detached from the cylinder.



Figure 4.44 Streamlines (moving frame) over vorticity (green: Min, blue: Max)

### 4.5.3 Static frame

The same instant of time used in section 4.5.2 is used in this section.

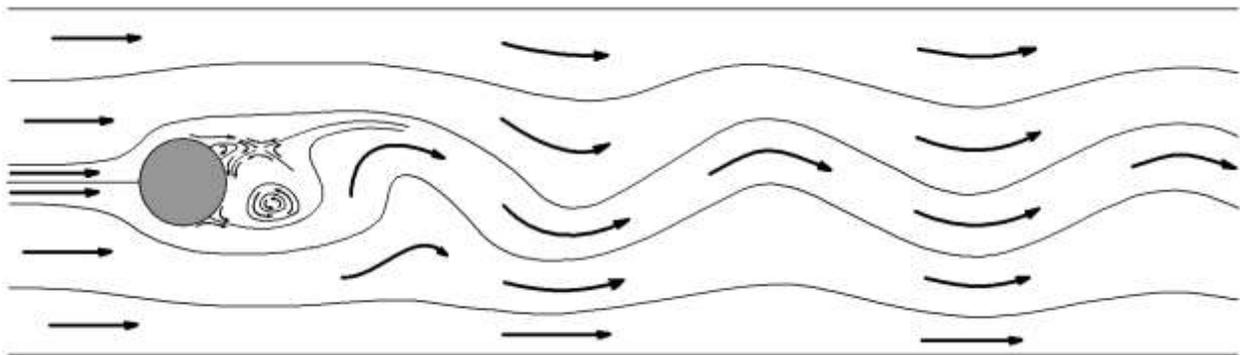


Figure 4.45 Streamlines (static frame)

The general picture of streamlines associated with the static frame of reference is shown in Figure 4.45. A detail of the region behind the cylinder can be seen in Figure 4.46.

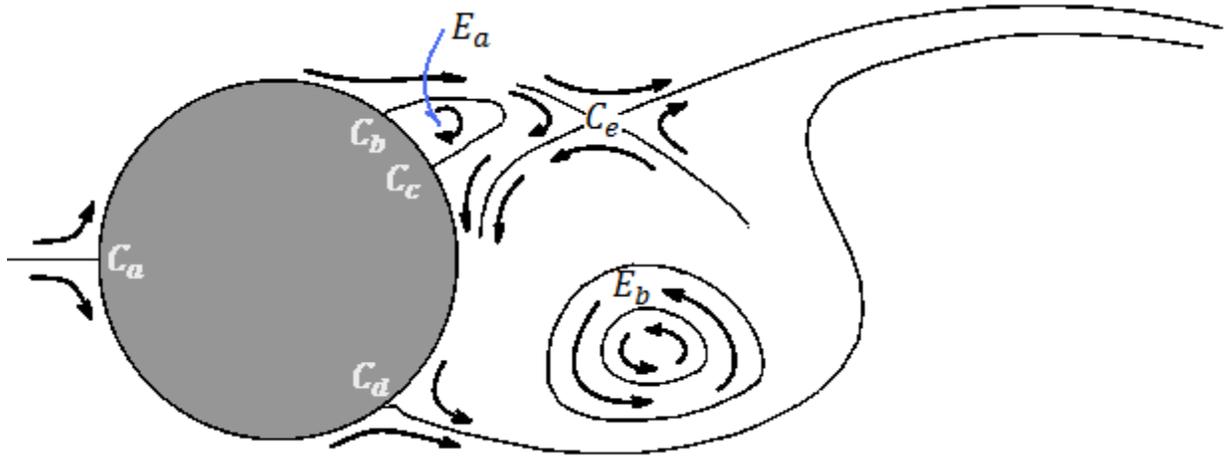


Figure 4.46 Streamlines (static frame) near to the cylinder

At any instant, there are four stagnation points on the cylinder:  $C_a$ ,  $C_b$ ,  $C_c$  and  $C_d$  in Figures 4.46 and 4.47. The key to understanding how vortices or eddies originate is revealed by following one of these points,  $C_c$ , in time and by associating this movement with the formation of an eddy,  $E_a$ , and a “four way” convergence point,  $C_e$ . For the Reynolds numbers used here, the angles between  $C_a$  and  $C_b$ , and between  $C_a$  and  $C_d$ , are approximately  $127^\circ$  while the angle between  $C_b$  and  $C_d$  is  $106^\circ$ . The location of these three stagnation points presents a small oscillation of  $\pm 2^\circ$  all moving in unison simultaneously during the shedding cycle. The other stagnation point,  $C_c$ , oscillates approximately  $\pm 30^\circ$  from the horizontal line during the shedding cycle. The oscillation of  $C_c$  is displaced  $180^\circ$  in the shedding cycle with respect to the oscillation of the other three points, so that when  $C_c$  is moving in a counter clockwise direction,  $C_a$ ,  $C_b$  and  $C_d$  move slightly but in a clockwise direction.

When the distance between  $C_c$  and  $C_b$  reaches a minimum,  $E_a$  and  $C_e$  also appear at the instant of time shown in Figures 4.46 and 4.47. This is also the instant when an older clockwise eddy at the top, but far from the cylinder, is either completely detached from the cylinder in the moving frame or completely absorbed by another “four way” convergence zone at the bottom in the static frame. The eddy  $E_b$  at the bottom finishes growing when the “four way” convergence zone,  $C_e$ , appears over it. What happens to  $E_b$ , is dictated by the future location of  $C_c$ , which completes the discussion. The convergence zone  $C_c$  moves towards  $C_d$  while eddy  $E_a$  grows and eddy  $E_b$  will either be consumed by  $C_e$  in the static frame of reference or simply displaced to the right in the moving frame of reference. While  $C_c$  approaches  $C_d$ , the bottom-left side of  $E_b$  is

horizontally stretched until the point it is detached from the cylinder (viewed in the moving frame) generating simultaneously another clockwise eddy close to  $C_d$  and a “four way” convergence zone in-between.

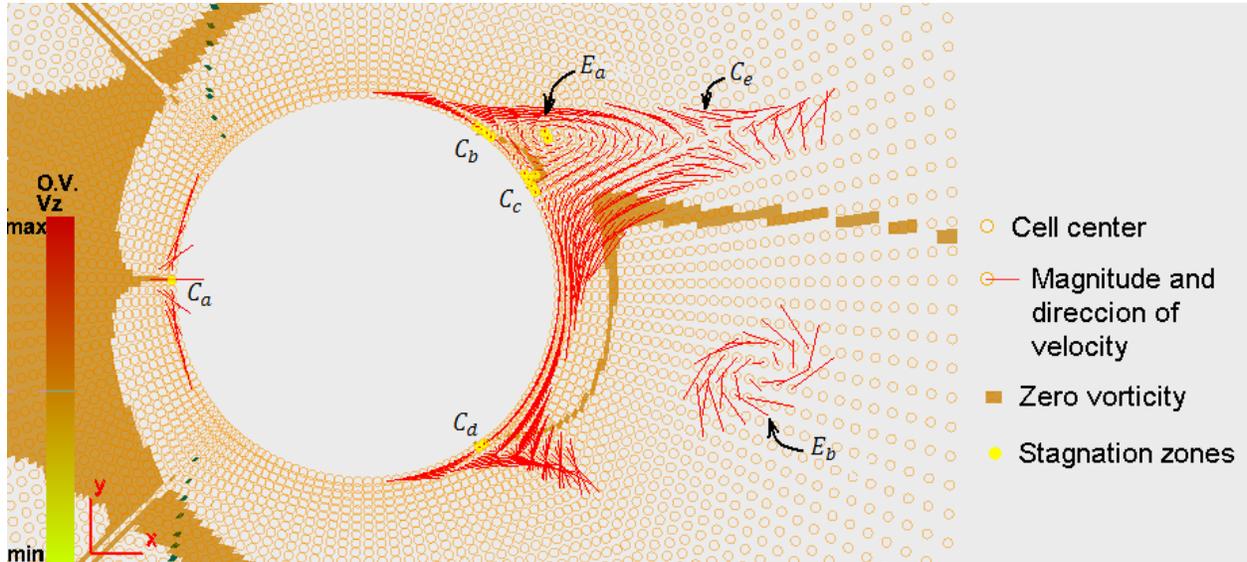


Figure 4.47 Eddies (static frame) and stagnation points at the cylinder



Animation 4.6 [Stagnation points on the cylinder, vorticity field background](#)

Animation 4.6 shows where the stagnation points are located and how they move on the cylinder face. A black contour line is set in VerFlow-V.01 to coincide with zero vorticity. Velocity vectors have been filtered so that only small scaled vectors are drawn using the static frame of reference.

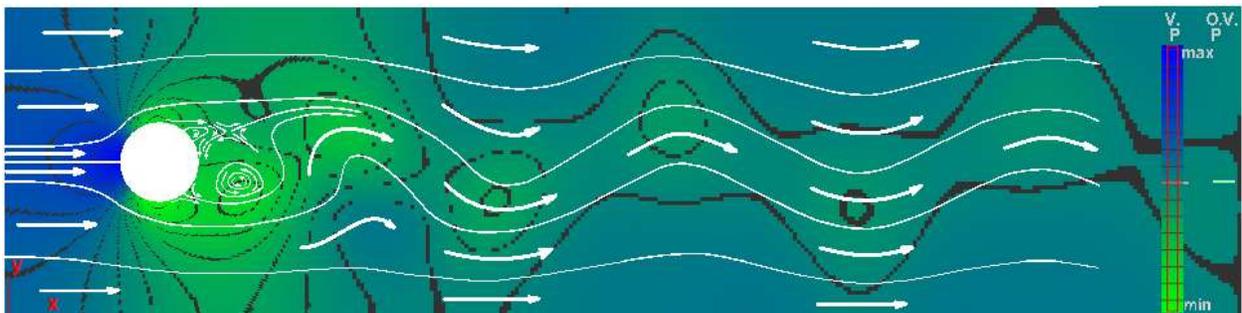


Figure 4.48 Streamlines (static frame) over pressure (green: Min, blue: Max)

Figures 4.48, 4.49 and 4.50 show the streamlines over the pressure, second invariant of the velocity gradient and vorticity fields respectively, in the static frame.

Figure 4.48 shows the streamlines for the same instant of time shown in Figure 4.42, but for the static frame of reference, both over the pressure field. It is clear, in the static frame, eddies closest to the cylinder are revealed since the velocity of the fluid in the wake region is close to zero while downstream the detailed streamline structure of the eddies is obstructed by the downstream motion. The static frame shows streamlines crossing over the low-pressure zones.

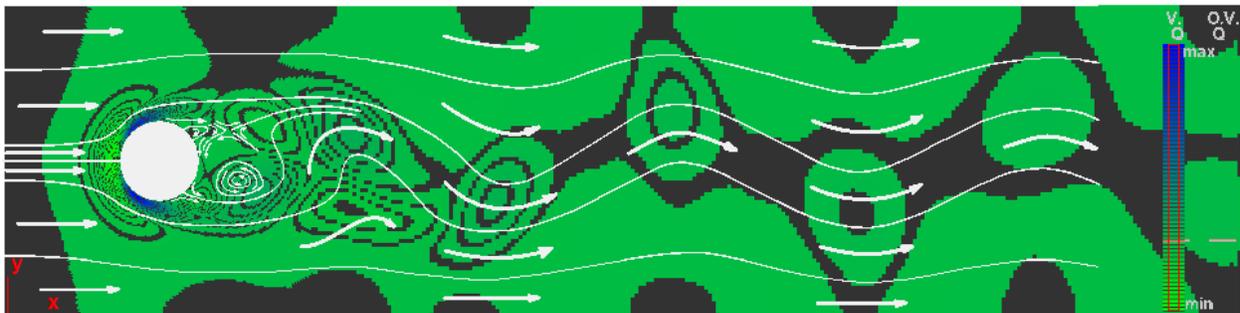


Figure 4.49 Streamlines (static frame) over second invariant of velocity gradient (green: Min, blue: Max)

Figure 4.49 shows the same result but over the second invariant of the velocity gradient  $Q$ . As noted before regions of positive  $Q$  are related with low-pressure regions downstream and with vortices or eddies, which are not envisioned in the static frame.

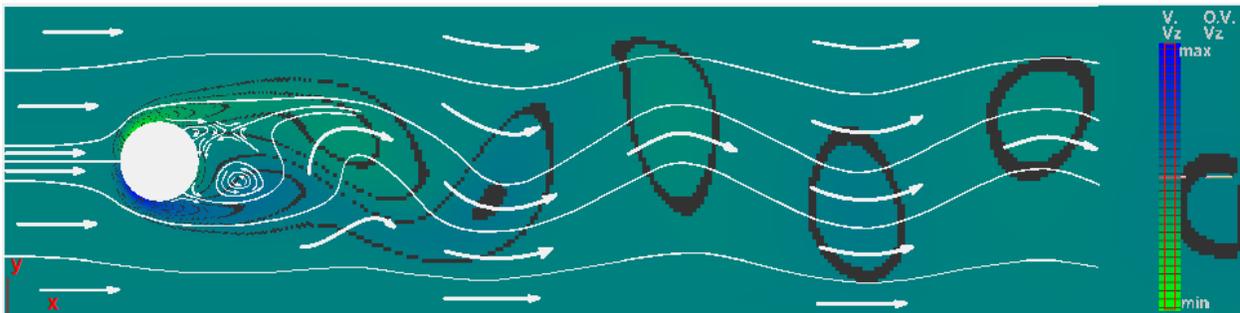


Figure 4.50 Streamlines (static frame) over vorticity (green: Min, blue: Max)

Similarly, Figure 4.50 shows the streamlines over the vorticity field in the static frame, again, for comparison purposes. The color legend bar indicates that blue regions correspond to positive vorticities or clockwise eddies downstream.

#### 4.6. Verification of finite difference forms of $Q$ using VerFlow-V.01

Results in this section and in Section 4.7 show close agreement between the model simulation, the numerical approach developed in this thesis and experimental data. In this section, finite difference forms for the second invariant of the velocity gradient,  $Q$ , are used to calculate this variable based on mathematical derivations realized in Chapter 3, verifying correspondence with the second invariant of the velocity gradient calculated by the OpenFoam simulation.



Figure 4.51 Legend bar for Figures 4.52, 4.53 and 4.54

The second invariant of the velocity gradient is evaluated using the three finite difference forms given in Section 3.2.3. These forms were used in a quantitative analysis in VerFlow-V.01 for block 4 in the mesh (see Figure 2.2). This comparison includes a *qualitative* analysis by envisioning simulation results using the filtered domain described in Section 4.2.4.

Figure 4.51 compares the color legend bar used in Figure 4.52 (yellow-red) with the color legend bar used in Figures 4.53, 4.54, 4.55 and 4.56 (green-blue). The small red rectangles highlighted in the color legend bars of Figure 4.51, indicate the visible region of the second invariant of the velocity gradient in the figures below, where only positive and small magnitudes of the  $Q$  variable are envisioned. These selected filtered domain regions facilitate a *qualitative* comparison of the three different finite difference forms in the figures that follows.

The second invariant of the velocity gradient tensor from the OpenFoam simulation is represented in Figure 4.52 for the entire region and is used also in Figures 4.53, 4.54, 4.55 and 4.56 for blocks 0, 1, 2 and 3 since  $Q$  is calculated for block 4.

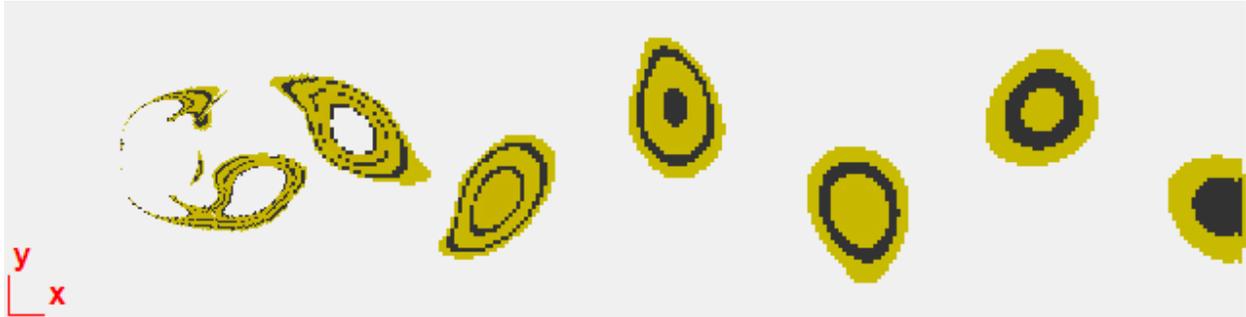


Figure 4.52 Second invariant of the velocity gradient from OpenFoam,  $Q$

The results in block 4 that use the central difference approach given by Equation 3.49 corresponds to Equation 3.35 and is shown in Figure 4.53. An identical reproduction of the OpenFoam result is generated using this approach.

Figure 4.54 and 4.55 show the results of applying Equation 3.36 with two different approaches. In Figure 4.54 a central difference approach given by Equation 3.52 is used while in Figure 4.55 a forward difference approach is used. The central difference gives again an identical result. Very small differences are observed when using the forward difference approach.

Figure 4.56 shows the result of applying a central difference approach to Equation 3.37. This approach is given by Equation 3.54. The result is not identical but very close to that calculated by OpenFoam. The main differences are found at the left and right boundaries of block 4 and must be generated by the extra number of cells required for the calculation (see Figure 1.2).

These results first confirm that these equations are similar and second suggest that OpenFoam uses either Equation 3.35 or Equation 3.36 with a central difference approach. OpenFoam documentation does not describe what finite difference forms are used.

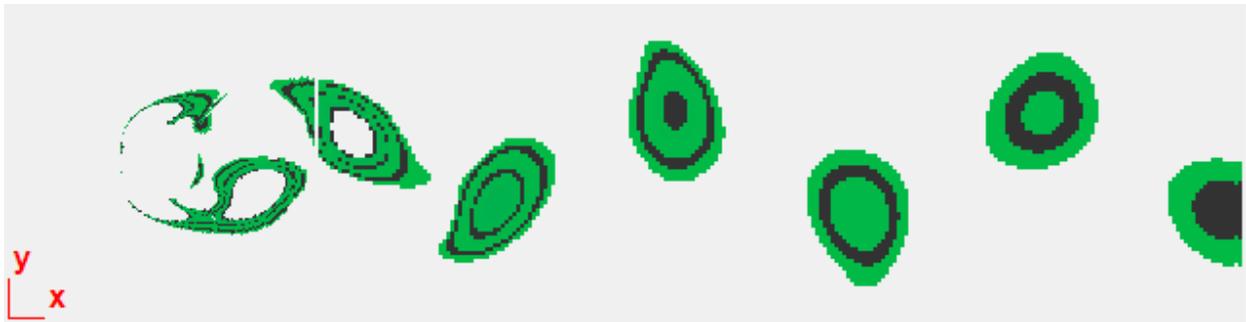


Figure 4.53  $Q$  applying Equation 3.35 with a central difference approach in block 4

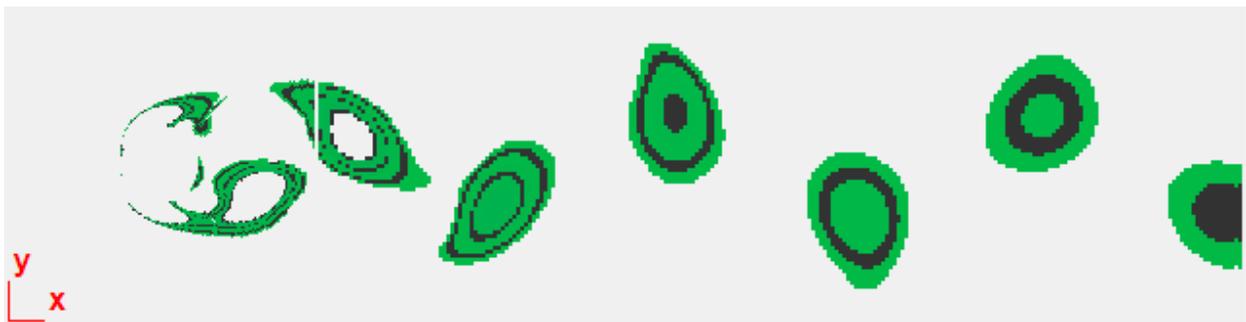


Figure 4.54  $Q$  applying Equation 3.36 with a central difference approach in block 4

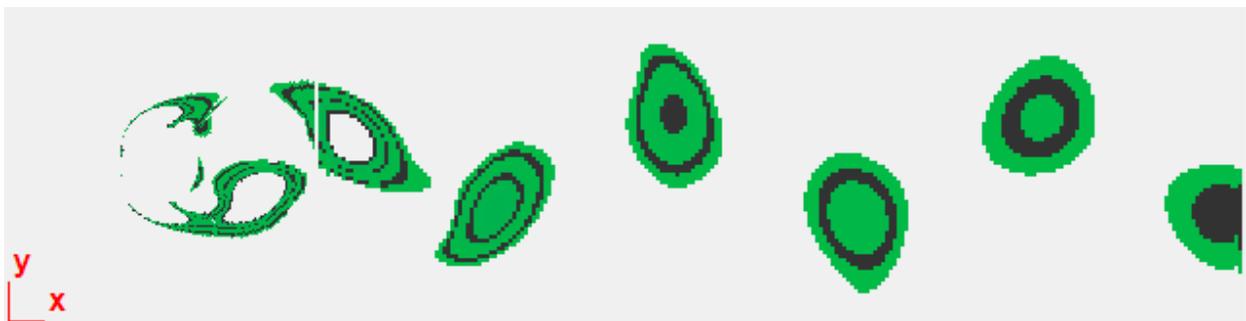


Figure 4.55  $Q$  applying Equation 3.36 with a forward difference approach in block 4



Figure 4.56  $Q$  applying Equation 3.37 with a central difference approach in block 4

## 4.7. Verification of the pressure distribution around the cylinder

Comparison of the pressure distribution from OpenFoam and experimental data for small Reynolds numbers completes the validation of simulation results studied in this thesis.

From the OpenFoam simulation, VerFlow-V.01 is customized to take the pressure at the cells next to the cylinder boundary. This information is plotted as a function of time and angular position, along the cylinder using cyan color. Figure 4.57 shows four instantaneous views of the pressure distribution along the cylinder. Positive pressure coefficients are represented pointing outside and negative pointing inside.

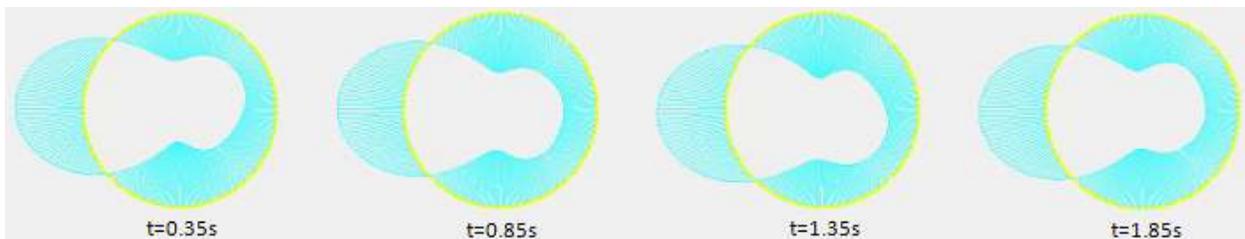


Figure 4.57 Instantaneous pressure distribution along the cylinder boundary envisioned by VerFlow-V.01

These four instantaneous pressure distributions are plotted in Figure 4.58 as a function of angle  $\alpha$ , defined counter clockwise from the horizontal left radius. The mean pressure at each location is also calculated and plotted for comparison.

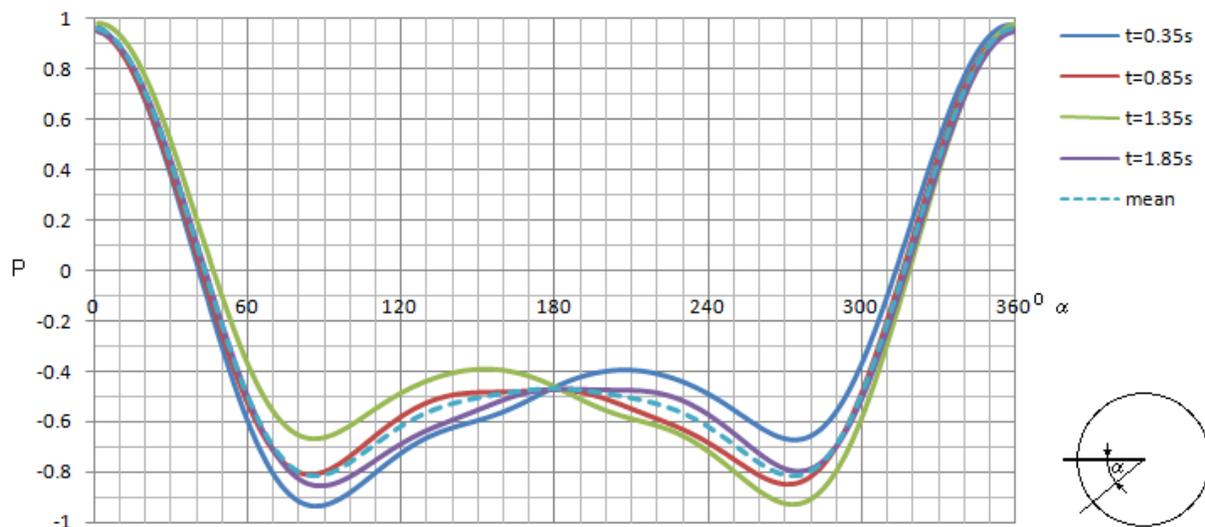


Figure 4.58 Instantaneous pressure distribution as a function of the angle from the horizontal.

The pressure distribution along the cylinder adapted from experimental data (Churchill, 1988) is shown in Figure 4.59 for Reynolds numbers  $Re = 67$  &  $73$  which are the closest to the  $Re = 67.4$  used in this Chapter.

Data from Figures 4.58 and 4.59 is used and plotted in Figure 4.60 for comparison.

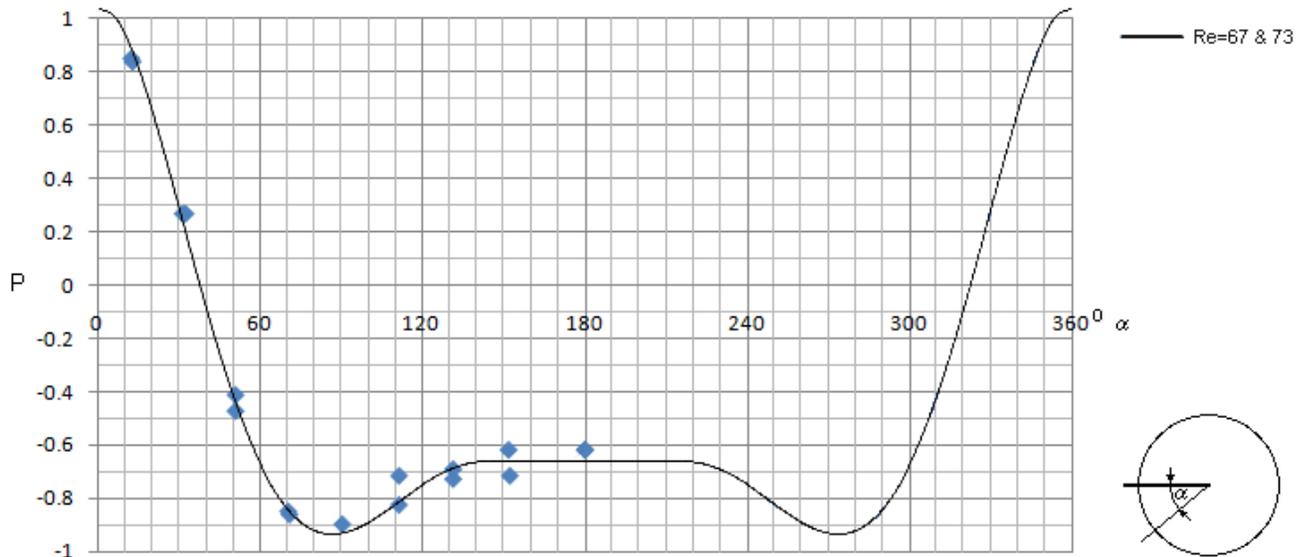


Figure 4.59 Pressure distribution for small Reynolds numbers adapted from Churchill book (Churchill, 1988)

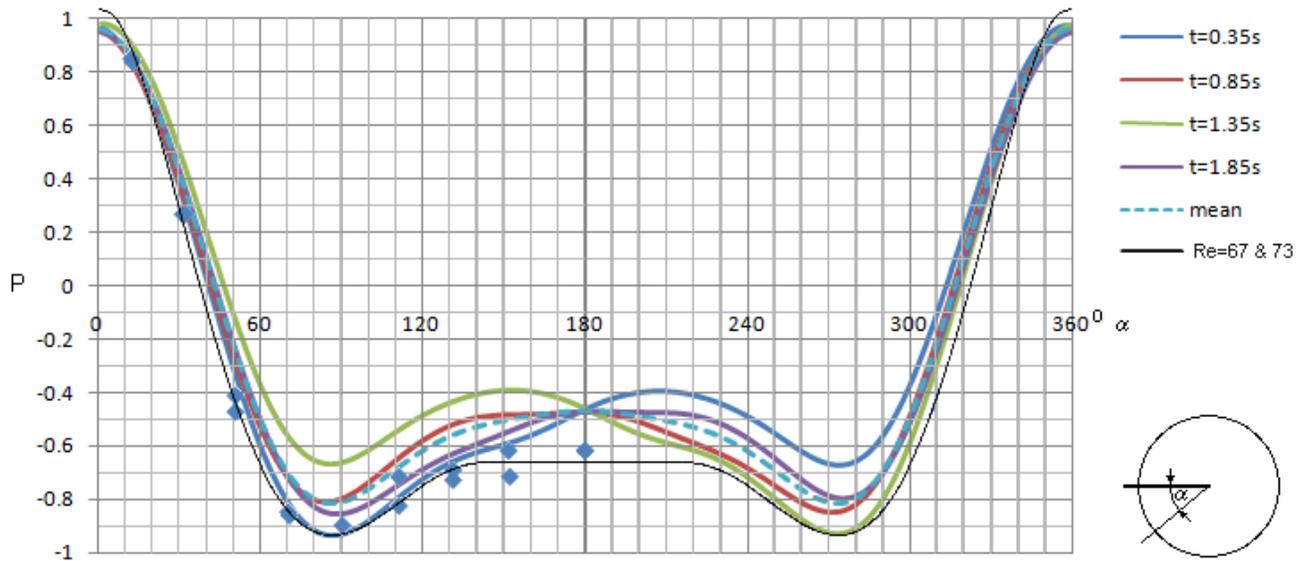


Figure 4.60 Comparison of instantaneous and mean pressure with experimental data

The pressure coefficient mean value for each angle about the cylinder in the simulation and the experimental data show the same general tendency, although the maximum and minimum experimental values are slightly larger with respect to the simulation.

In general good agreement is observed where the pressure variation in time is confirmed in Figure 4.60. This result is significant because pressure variation has been extensively used in this thesis as a comparative point for other calculations.

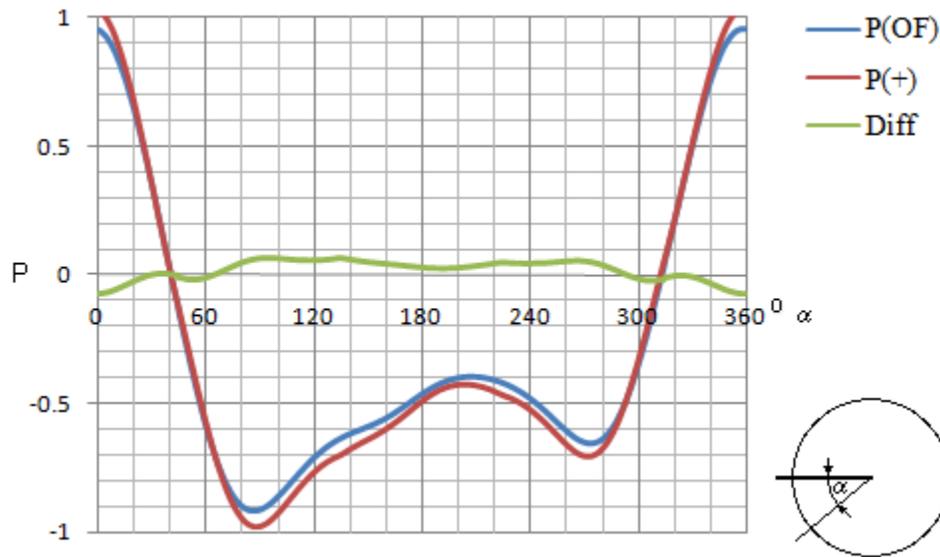


Figure 4.61 Predicted and simulated pressure along the cylinder boundary for  $t = 0.35[s]$

The prediction of the pressure using the integral solution of the Poisson equation,  $P(+)$ , which is given by Equation 4.84, is compared in Figure 4.61 with the pressure from the OpenFoam simulation,  $P(OF)$ . The difference,  $Diff$ , between these pressures is shown for every cell along the cylinder for the simulation time  $t = 0.35[s]$ . Good agreement is observed between both curves.

Along the cylinder boundary and for the same instant of time  $t = 0.35[s]$ , the predicted pressure,  $P(+)$ , is shown in Figure 4.62 as the sum of the integral terms  $(Q)$ ,  $(CI_1)$  and  $(CI_2)$  described in Section 4.2.

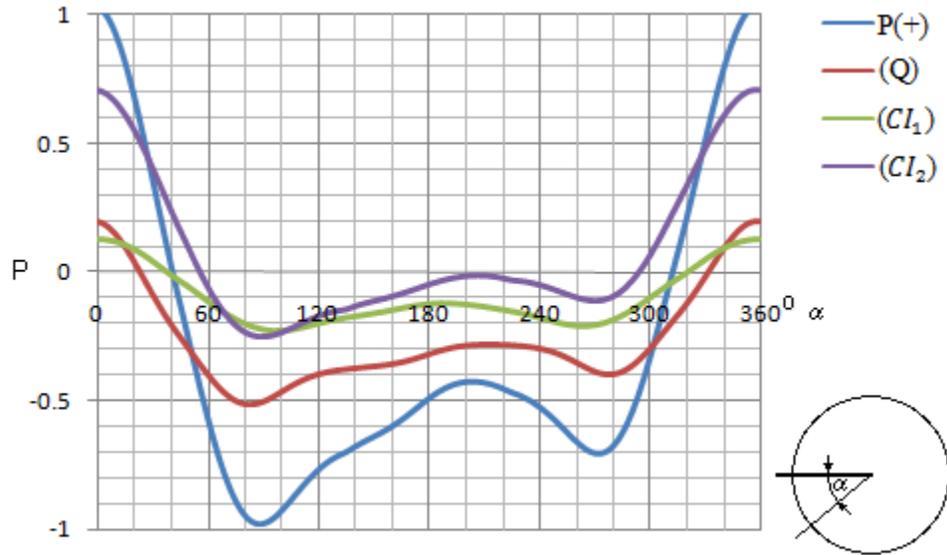


Figure 4.62 Integral components for the pressure along the cylinder boundary for  $t = 0.35[s]$

The drag effect is calculated from the three components of  $P(+)$ . This result is shown in Table 4.1 for times  $0.35[s]$ ,  $0.85[s]$ ,  $1.35[s]$  and  $1.85[s]$  while the period is  $1.95[s]$ . The drag calculated from the surface integral is 29% of the drag due to pressure, while the contribution from the first contour integral is 20% and from the second contour integral is 51%. These values are approximately constants in time.

t [s]	Drag P(+)	Drag (Q)	Drag (CI1)	Drag (CI2)
0.35	1.78	0.51	0.35	0.92
0.85	1.77	0.50	0.35	0.91
1.35	1.79	0.52	0.35	0.92
1.85	1.77	0.50	0.35	0.91

Table 4.1 Drag contributions from integrals in Equation 3.84

The contour integral components ( $CI_1$ ) and ( $CI_2$ ), shown in Figure 4.62, are decomposed into their constituents: the left (L), right (R), top (T), bottom (B) and cylinder (Cy) contour portions. These results are shown in Figures 4.63 and 4.64.

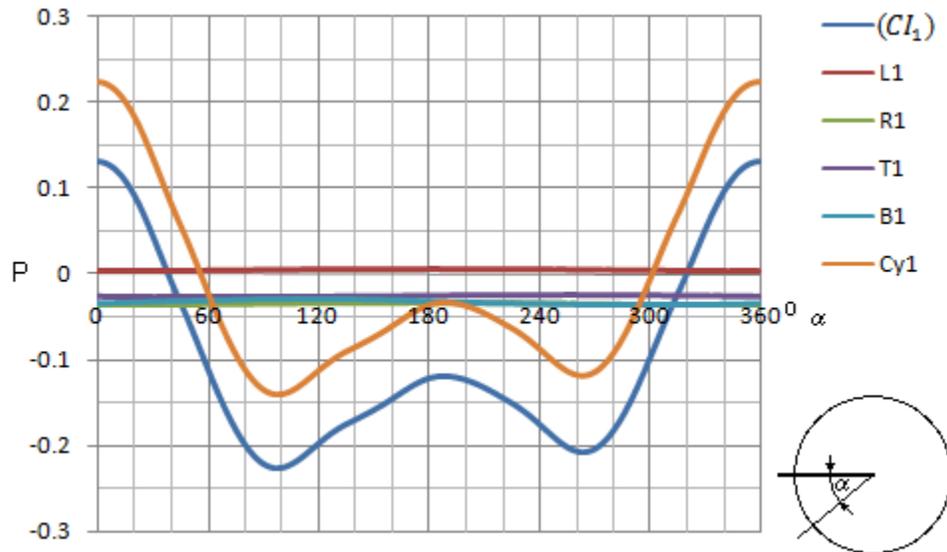


Figure 4.63 Components of  $(CI_1)$  from each boundary for  $t = 0.35[s]$

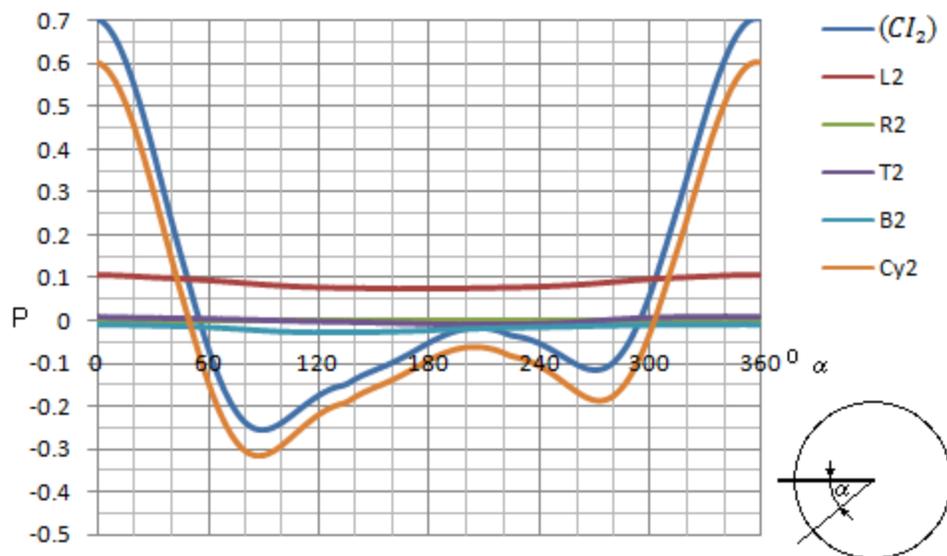


Figure 4.64 Components of  $(CI_2)$  from each boundary for  $t = 0.35[s]$

Figures 4.64 and 4.65 reveal that the general shape of the contour integral curves,  $(CI_1)$  and  $(CI_2)$ , is given by the cylinder boundary while the other boundaries contribute with an approximate constant pressure, resulting in zero drag contributions. As a consequence the drag effect from  $(CI_1)$  and  $(CI_2)$  originates on the cylinder boundary.

#### 4.8. Chapter nomenclature

$\Delta P$  pressure per unit density contribution

$\Delta P_P$	pressure per unit density at point $P$ contribution
$\pi$	constant = 3.14159265359
(+)	predicted value for pressure from sum of integrals
$A$	area
( $B$ )	boundary identifier meaning bottom
$C_D$	drag coefficient
$C\#$	convergence zone number # (moving frame)
$C_\#$	convergence zone number # (static frame)
$C_L$	lift coefficient
( $CI_1$ )	first term of contour integral in Equation 3.84
( $CI_2$ )	second term of contour integral in Equation 3.84
$E\#$	eddie zone number # (moving frame)
$E_\#$	eddie zone number # (static frame)
$L$	contour length
( $L$ )	boundary identifier meaning left
$\bar{n}$	normal vector pointing outside the cylinder
( $O.V.$ )	overlapped variable
$P$	pressure per unit density or point in an arbitrary domain
( $P$ )	pressure per unit density from OpenFoam simulation
$P_P$	pressure per unit density at point $P$
$Q$	second invariant of the velocity gradient or point in an arbitrary domain

- ( $Q$ ) surface integral in Equation 3.84
- $Q_Q$  second invariant of the velocity gradient at point  $Q$
- $r$  cylinder radius or distance from point  $P$  to point  $Q$
- ( $R$ ) boundary identifier meaning right
- Re Reynolds number
- ( $T$ ) boundary identifier meaning top
- $u_\infty$  velocity at the inlet
- $u_x$  streamwise velocity
- ( $V$ .) variable

## Chapter 5 . Conclusions

The discussions in previous chapters include a variety of results and findings. These are organized here into conclusions using a narrative format. Although the 2D periodic laminar flow around a cylinder has been extensively studied, results here have contributed to understanding more about the complexity of this flow. This chapter summarizes the most relevant results, first simulation “data” is validated, followed by the development of VerFlow-V.01 that was used to explore and envision the complexity of periodic laminar flow both quantitatively and *qualitatively*.

### 5.1. Programs

Two computer programs were used: (1) OpenFoam was used to create initial numerical simulation “data” and (2) VerFlow-V.01 was developed and customized for this particular case to interpret the OpenFoam simulation results.

#### 5.1.1 OpenFoam simulation

The OpenFoam simulation models 2D periodic laminar (PL) flow (see Figure 1.4) for the PL range of Reynolds numbers given in Table 2.2. Numerical results were validated using FFT and the Strouhal number. In general the results show good agreement with previous experimental data, such as the general behavior of the velocity field, vorticity field and the pressure distribution along the cylinder boundary (see Figure 4.60). The constriction of the flow due to the walls at the top and bottom increases the drag force in comparison to the unbounded case, usually reported in the literature. Our result for the drag was  $C_D = 2.24$ , while experimental data varies, e.g. between  $C_D = 1.54$  and 1.95 depending upon the source (see Figure 4.1). Constricted flow between the cylinder and wall boundaries may also be responsible for a change in the shedding frequency and as a consequence the Strouhal number. The Strouhal numbers in our simulation were 0.207 for  $Re = 56.2$  and 0.214 for  $Re = 67.4$  (see Equation 2.29) while 3D experiments in unbounded flows gives values around 0.15 for these Reynolds numbers (Churchill, 1988) and asymptotes,  $St = 0.21$  (3D) and  $St = 0.2417$  (2D), are also determined for larger Reynolds numbers  $Re > 200$  (Kundu & Cohen, 2004).

The grid defined in Section 2.3 provides good spatial resolution of the entire flow region and even higher resolution close to the cylinder. The features that can be resolved are on the order of 1% of the cylinder diameter in the region near to the cylinder and on the order of 5% of the cylinder diameter in the far region.

### **5.1.2 VerFlow-V.01**

The OpenFoam simulation results can be studied, with some limitations, in ParaView but ParaView is limited in its versatility. The results reported in this research were possible because of the development of a customized program: VerFlow-V.01, “Ver” is a Spanish word which means “see”, which reflects the enhanced *qualitative* imaging capabilities of the VerFlow-V.01 program. The quantitative and *qualitative* characteristics implemented in VerFlow-V.01, used throughout this thesis are unique, in the sense that these concepts evolved as a consequence of many discussions among the committee members advising this research.

Three differential equations and the corresponding finite difference forms were derived and used to calculate the second invariant of the velocity gradient,  $Q$ . The numerical results of these three finite difference forms were implemented using VerFlow-V.01, and are shown to be consistent with the second invariant of the velocity gradient as calculated by OpenFoam. These results were also used to solve the surface integral in Equation 3.84, which proved to be a significant component in calculating the pressure at any point in the flow field.

### **5.1.3 Time resolution**

An innovative idea for a better use of the available “data” (numerical results from the OpenFoam simulation) was implemented which increased the temporal resolution of the data set. The idea is to rearrange information from several cycles into a single cycle, taking advantage of the real time resolution of the original simulation in OpenFoam (see Section 2.7.5 for details). This increased time resolution was used when investigating the phenomena involved in the shedding nature of this flow.

## **5.2. Forces on the cylinder**

Drag and Lift coefficients, due to pressure and viscous effects acting along the cylinder were calculated in VerFlow-V.01. The calculations were organized with respect to four arcs,

associated with the four blocks or quadrants, enclosing the cylinder. Results are decomposed with respect to each separate quadrant of the cylinder surface.

The most important pressure contributions to the drag are associated with arcs 1 and 3 (see Figure 4.13) while the pressure drag effects associated with arcs 0 and 2 (bottom and top quadrants) can be neglected.

The most important pressure contributions to the lift are associated with arcs 0 and 2 (bottom and top quadrants, see Figure 4.13) whose individual oscillations are in phase and results in a strong oscillation when summed.

The most important viscous contributions to the drag come from arcs 0, 2 and 3 (bottom, top and front quadrants) while viscous effects along arc 1 (back quadrant) can be neglected (see Figure 4.15). The resultant viscous drag is still small when compared with the total drag: 25.8% of the total (see Figure 4.4).

The most important viscous contributions to the lift come from arcs 1 and 3 (back and front quadrants), while the magnitude of lift associated with arcs 0 and 2 (bottom and top quadrants) is significant but effectively cancel when summed (see Figure 4.15). Viscous effects on the lift are also small: 15% (see Figure 4.8). A phase delay in the maximum viscous lift of  $32.3^\circ$  with respect to the pressure lift is associated with the vortex shedding.

The drag exhibits an oscillation at a double the shedding frequency (see Figure 4.5), which is consistent with Osama Marzouk's previous analysis (Marzouk, 2009). The oscillation for the total drag is precisely predicted in amplitude and phase for our case from the oscillation given by the combined action of pressure drag from blocks 1 (right) and 3 (left).

### **5.3. The second invariant of the velocity gradient, $Q$**

The second invariant of the velocity gradient,  $Q$ , is used in this thesis to predict the pressure at any point (see Section 4.2), the contribution of  $Q$  to the pressure distribution on the cylinder boundary associated with any arbitrary cell (see Section 4.3), and the drag and lift contributions on the cylinder for every cell in the entire domain (see section 4.4). Equation 3.84, which is the 2D solution for the Poisson equation, was implemented in VerFlow-V.01 and used to establish the following conclusions.

### ***5.3.1 Pressure at a point***

The prediction of the pressure at a point is possible using VerFlow-V.01 for rectangular domains in block 4 and also for the entire domain. The results in the prediction of the pressure are always very close or equal to the pressure given by OpenFoam in both domains. VerFlow-V.01 allows the user to compare the influence and “weight” of each component contributing to the prediction of the pressure at a point from the integral solution to Poisson’s equation. These components are a surface integral including the second invariant of the velocity gradient, and two contour integrals with contributions from each boundary (left, right, top, bottom or cylinder). For both domains, the surface integral of  $Q$  typically has significant contributions. The contributions from boundaries are sometimes small, especially when these boundaries are far from the point where the pressure is predicted.

VerFlow-V.01 allows the user to limit the surface integration component given in Equation 3.84a. Adjusting the minimum and maximum variable limits on the data acts as a filter for the surface integration. Since this filter can be applied directly to any of the variables, this tool becomes extremely useful for identifying the more important contributions from the velocity field.

### ***5.3.2 Drag and lift contributions***

From the pressure distribution, which was generated along the cylinder boundary for every cell in the domain, drag and lift contributions were also derived and implemented in VerFlow-V.01.

The strongest region, which provides positive drag, is located in front of the cylinder and oscillates slightly up and down from the horizontal line which corresponds to the small doubled shedding frequency variation observed for the drag on the cylinder, although this effect is attenuated by the other regions in the flow (see discussion in Section 4.4), e.g. in particular the effect generated near to the stagnation points behind the cylinder. The study of drag contributions clearly shows that vortices downstream tend to individually contribute to the drag force on the cylinder while regions outside the vortices present negative contributions.

The stronger positive and negative zones which contribute to the lift are located upstream of the cylinder, positive zones exist very close to the top-front cylinder boundary (positive contribution) and negative zones exist very close to the front-bottom cylinder boundary (negative

contribution). These two zones propagate from the cylinder into the wake and alternately shed as eddies (vortices). These eddies grow and move downstream. The extended zone associated with the vortices moves horizontally and couple with another zone of lower intensity, which has the same sign and can be associated with the four way convergence zone.

The surface integral (Q) of equation 3.84 contributes to the cylinder, drag due to pressure with 29% , while the first contour integral (CI1) does so with 20% and the second contour integral (CI2) with 51% for the specific case studied in Chapter 4. Since these values remain fairly constant, it is enough to know the (Q) contribution and divide the result by 0.29 to get the drag due to pressure for this case.

#### **5.4. Additional *qualitative* remarks**

The customized program VerFlow-V.01 facilitates understanding complex flow phenomena associated with the periodic laminar flow around a cylinder. The following three topics were examined in this work and aid in interpretation of the flow field: (1) moving and static frames, (2) eddy, convergence and stream zones, and, (3) stagnation points.

##### **5.4.1 *Moving and static frames***

VerFlow-V.01 was customized to envision the filtered velocity field including direction and magnitude of the vectors for frames of reference moving at a variable velocity. Two frames of reference are used to identify the vortices: the moving and the static. In the moving frame (see Figure 4.37), vortices can be seen traveling downstream while in the static frame (see Figure 4.45), vortices can be seen close to the cylinder because their velocity is also close to zero. The vortices in the static frame grow and disappear when they move out of the reverse flow region. The behavior of the flow in the moving and static frame is described using the definitions introduced by Hunt et al: eddies, convergence and stream zones (Hunt, Wray, & Moin, 1988). Figures 4.37 and 4.45 show the reference streamlines for the flow at a fixed time in the moving frame of reference and in the static frame of reference respectively.

##### **5.4.2 *Eddy, convergence and stream zones:***

“Four way” and “three way” convergence zones are defined in Section 4.5.2 as regions in which currents of flow are either divided or joined. The constriction in the flow because of the walls

transforms a “four way” convergence zone into two “three way” convergence zones at the top or bottom walls. The “four way” convergence zone appears when an eddy is detached from the cylinder boundary, which can be seen in the moving frame of reference. In the static frame, an eddy and a “four way” convergence zone on the opposite side consume each other along the reverse region of the flow in the wake and both disappear. When the eddy is detached from the cylinder, the formation of a “four way” convergence zone is realized and the formation of a small eddy at the same side of the cylinder is also observed very close to the cylinder boundary. A detailed explanation of the vortex or eddy formation is given in section 4.5.3. VerFlow-V.01 helps the user to identify the convergence and eddy regions dynamically in an interactive *qualitative* sense.

### **5.4.3 Stagnation points**

For the Reynolds number of 67.4 flow in the constricted region studied in this simulation, four stagnation points are observed along the cylinder boundary. The first stagnation point is located at the left and presents a slight oscillation  $\pm 2^\circ$ . Two other stagnation points are located at  $127^\circ$  symmetrically at the top and bottom sides. The oscillation of  $\pm 2^\circ$  also affects these two stagnation points which are all in phase. A fourth stagnation point moves down and up behind the cylinder approaching the other two stagnation points in an oscillation of  $\pm 30^\circ$ . The oscillation of this point is not in phase with respect to the oscillation of the other three stagnation points, which are displaced  $180^\circ$ . A detailed explanation is given in Section 4.5.3 where these stagnation points are *qualitatively* envisioned in Animation 4.6, from images originally developed in VerFlow-V.01.

## References

- Churchill, S. W. (1988). *Viscous Flows*. Stoneham: Butterworth Publishers .
- Den Hartog, J. P. (1953). Recent technical manifestations of Von Karman's vortex wake. *Proceedings. N.A.S.* (pp. 155-157). Massachusetts: Department of Mechanical Engineering.
- Feynman, R. P., Leighton, R. B., & Sands, M. (1964). *The Feynman lectures on physics*. Massachusetts: Addison-Wesley Pub. Co.
- Frederick, D., & Chang, T. S. (1972). *Continuum Mechanics*. Cambridge: Scientific Publishers, Inc.
- Fromm, J. E. (1963). *A method for computing nonsteady, incompressible, viscous fluid flows*. New Mexico: Los Alamos, California University.
- Hunt, J., Wray, A., & Moin, P. (1988). Eddies, streams, and convergence zones in turbulent flows. *Center for Turbulence Research* (pp. 193-208). CTR-S88.
- Jeong, J., & Fazole, H. (1994). On the identification of a vortex. *Journal of Fluid Mechanics* , 69-94.
- Kundu, P., & Cohen, I. (2004). *Fluid Mechanics*. Amsterdam; Boston: Elsevier academic Press.
- Marzouk, O. A. (2009, February). Simulation, modeling, and characterization of the wakes of fixed and moving cylinders. Blacksburg, Virginia: ETD PhD Dissertation.
- Mase, G. E. (1970). *Continuum Mechanics*. New York: McGraw-Hill.
- Norberg, C. (1994). An experimental investigation of the flow around a circular cylinder: influence of aspect ratio. *Journal of Fluid Mechanics* , 287-316.
- Pier, B. (2002). On the frequency selection of finite-amplitude vortex shedding in the cylinder wake. *Journal of Fluid Mechanics* , 407-417.
- Sokolnikoff, I., & Redheffer. (1966). *Mathematics of physics and modern engineering*. New York: McGraw-Hill.
- Van Dyke, M. (1982). *An album of fluid motion*. Stanford: The Parabolic Press.
- Warsi, Z. (1993). *Fluid Dynamics*. Boca Raton: CRC Press.

## **Appendix A. General guide to modify the original OpenFoam Simulation**

### **Part I: OpenFoam**

OpenFoam (Open Field Operation and Manipulation) is an open source program for fluid dynamic simulation. The following link connects directly with the corresponding website: <http://www.opencfd.co.uk/openfoam/>. Tutorials are available going through the Documentation User's Guide tabs.

Generating data for an incompressible viscous flow passing around a cylinder requires attention to the following requirements for reproducibility of results.

FoamX can be used as a pre-processing tool that modifies the corresponding files from a user friendly window. One can also modify the files directly by accessing them through an editor such as WinVi in a MS-Windows Operating System (OS), or the vi editor working in a UNIX/Linux OS shell.

The key information to create the grid is stored in the "blockMeshDict" file in the "constant/polyMesh" folder. Vertices, blocks, edges and patches are defined here. OpenFoam always works in a three dimensional space. Blocks are defined by eight vertices. Where the line between two vertices is an arc, it has to be written in the "edges" section. The required information for an arc is the two vertices and the coordinates of any point different from the vertices in the arc. Each block is divided in cells and the cell size can vary gradually depending on selecting the "simpleGrading" option. The inlet, outlet and walls including the cylinder are defined as "patches"

IcoFoam is selected among nine incompressible flow solvers, since it is for transient incompressible, laminar flow of Newtonian fluids. This information must be included in the "ControlDict" file in the "system" folder.

The kinematic viscosity is entered as "nu" in the "transportProperties" file in the "constant" folder.

The "boundary" file in the "constant" folder defines the characteristics of each "patch". Here slip condition is established at the walls and no-slip condition at the cylinder.

The “U” file in the “0” folder contains the initial data for velocities. It must include the velocity uniform value for the inlet boundary condition.

The “P” file in the “0” folder does the same for the “U” file but with the pressure field. In this case, it establishes a “uniform 0” value at the outlet.

The “fvSolution” file in the “system” folder”, specifies the solvers and tolerances used for solving the “P” and “U” fields.

The “fvSchemes” file located in the same folder, indicates the gradient, divergence, interpolation and general schemes used. In general they can remain as the original defaults (Euler, Gauss linear).

To encourage reproducibility of results files listed above have been archived and can be downloaded from the Virginia Tech ETD system.

## **Part II: Data modification**

The objective here is to provide an instructive example where results shown in chapters 3 and 4 can be reproduced and modified as necessary. To do this, change for example the velocity at the inlet, the time step or the solver directly in the original files, run OpenFoam again and change some specific information in VerFlow-V.01 to work with different primary results.

We have used the Constellation PCs: Taurus, Leo and Libra computers at the Laboratory for Scientific Visual Analysis at Virginia Tech to generate the original data from OpenFoam-1.4.1.

VerFlow-V.01 requires the following OpenFoam files:

- “NameCase/constant/polyMesh/blockMeshDict” describes the geometry and should not be altered at all
- “NameCase/Time/U” Velocity field for a specific time
- “NameCase/Time/P” Pressure field for a specific time
- “NameCase/Time/vorticity” Vorticity field for a specific time
- “NameCase/Time/Q” Second invariant of the velocity gradient (Q) field for a specific time

Here is a general description of the procedure to modify the data from one of those already used, which now we call "NameCase":

- Put a copy of the folder case "NameCase" in a folder (Root) inside the tutorials
- Inside NameCase" should only be the folder "0", "constant" and "system". The other folders have to be erased from there)
- Inside the folder "0" should only remain the files "P" and "U" and the other folder and files should be erased)
- Open a terminal
- Move to the tutorials directory: `cd ...run/tutorials/`
- Run FoamX
- Open Case Browser
- Open the Root (in which is already located "NameCase")
- Open the Case "NameCase"
- Edit and modify if necessary the Dictionary Entries
- Check especially the values in parenthesis: `deltaT (0.001)`, `endTime (80)`, `writeInterval (200)`, `writeFormat (ascii)` in the controlDict Dictionary; `nu (8.9E-7)` in transport properties; and, `Uleft (uniform 0.012,0.0,0.0)` in U field. Some of them can be changed, like the velocity at the left to get a new data set for a different Reynolds.
- Run BlockMesh from `FoamUtilities>Mesh>Generation>blockmesh`. This will generate the mesh for the calculations.
- Run the case by pressing the play button "Start Calculation". It will take a long time, probably some days.
- Run Q and vorticity from `FoamUtilities>postprocessing>velocityfield`
- Finally, copy the folder results (the whole folder "NameCase") in VerFlow-V.01 folder.

VerFlow-V.01 requires adjustment from the code in some initial variables in order to process the previous results. Among the required information are the new "NameCase", the number of time steps, the time interval and the general description. Note here that VerFlow-V.01 is fully commented, but should be studied wisely to ensure the changes are not generating wrong results. VerFlow-V.01 was developed for a specific problem and it will require few changes for minor

modifications in the original data but things like inserting a new block in the geometry, for example, will require writing a new program.

## Appendix B. VerFlow-V.01: Main Subroutines

*Ver*, a Spanish word which means “see” was used since it is the primary language of the author and the main objective of VerFlow-V01 was to envision and discover *qualitatively* the intuitive world behind the quantitative calculations which were also included in VerFlow-V.01. VerFlow-V.01 was written in Microsoft Visual Basic 2008. In this Appendix, the Main Subroutines of VerFlow-V.01 are shown in the order indicated under the Code Reference title. All programs are copyright protected Miguel Darío Ortega López under the VerFlow license below.

### VerFlow Software License

Copyright © 2009 by Miguel Darío Ortega López. All rights reserved. Some individual files may be covered by other copyrights (this will be noted in the file itself).

Redistribution and use in source and binary forms are permitted provided that this entire copyright notice is duplicated in all such copies. No charge may be charged for copies, derivations, or distributions of this material without the express written consent of the copyright holder.

THIS SOFTWARE IS PROVIDED “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR ANY PARTICULAR PURPOSE.

IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, LOSS OF USE, DATA, OR PROFITS OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

### CODE REFERENCE

The following list is a reference of the selected VerFlow-V.01 subroutines, which code is shown in the next section of this Appendix.

*Reading velocity field:* Reads and stores velocity field.

*Reading pressure:* Reads and stores pressure field.

*Reading second invariant of the velocity gradient, Q:* Reads and stores Q field.

*Reading vorticity field:* Reads and stores vorticity field.

*Calculating Q, 3 different approaches, block 4:* Calculates Q from 3 finite difference equations.

General geometry:

*Calculating and envisioning flow around a cylinder:* Principal subroutine of VerFlow-V.01. This part of the program includes filtering the variable (V.) and the overlapped variable (O.V.), establishing contour lines, calculating the contour integrals at each boundary and Q integration for rectangular and entire domains, grid and velocity vectors (moving and static frames), contributions to the pressure field from a point, pressure and viscous forces.

Conversion to the unique reference system (five blocks subroutines)

Drag and Lift pressure contributions from every point

## **SELECTED CODE**

### ***Reading velocity field:***

---

```
Sub Vel() 'read velocities and get magnitude
Me.Text = "Reading velocity field"
For it = 0 To nt - 1
    it1 = t0 + it * dt
    PNameFileM = Ptxt2 & Ptxt3 & "\" & it1 & ".u"
    Dim sr As StreamReader = New StreamReader(PNameFileM)
    Do While Ptxt1 <> nn
        Ptxt1 = sr.ReadLine()
    Loop
    Ptxt1 = sr.ReadLine() 'ready to begin with useful data
    For i10 = 0 To 3 'first 4 blocks
        For i = 1 To NR
            For i11 = 1 To NQ
                Ptxt1 = sr.ReadLine()
                Vcomp()
                Vx(i10, i11 - 1, i - 1, it, 0) = u1
                Vx(i10, i11 - 1, i - 1, it, 1) = u2
                Vx(i10, i11 - 1, i - 1, it, 2) = u3
                Vx(i10, i11 - 1, i - 1, it, 4) = uu
                If it = 0 And i10 = 0 And i = 1 And i11 = 1 Then
                    ulmin(0) = u1 : ulmax(0) = u1
                    ulmin(1) = u2 : ulmax(1) = u2
                    ulmin(2) = u3 : ulmax(2) = u3
                End If
            Next i11
        Next i
    Next i10
Next it
```

```

        ulmin(4) = uu : ulmax(4) = uu
    End If
    Next
    Next
    Next i10
    'fifth block
    For i11 = 1 To NL
        For i = 1 To NQ
            Ptxt1 = sr.ReadLine()
            Vcomp()
            Vx4(NL - i11, i - 1, it, 0) = u1
            Vx4(NL - i11, i - 1, it, 1) = u2
            Vx4(NL - i11, i - 1, it, 2) = u3
            Vx4(NL - i11, i - 1, it, 4) = uu
        Next
    Next
    Next
    sr.Close()
Next
End Sub

```

### ***Reading pressure:***

---

```

Sub Pres() 'read pressure
Me.Text = "Reading P"
For it = 0 To nt - 1
    it1 = t0 + it * dt
    PNameFileM = Ptxt2 & Ptxt3 & "\" & it1 & ".p"
    Dim sr As StreamReader = New StreamReader(PNameFileM)
    Do While Ptxt1 <> nn
        Ptxt1 = sr.ReadLine()
    Loop
    Ptxt1 = sr.ReadLine() 'ready to begin with useful data
    For i10 = 0 To 3 'first 4 blocks
        For i = 1 To NR
            For i11 = 1 To NQ
                Ptxt1 = sr.ReadLine()
                Vx(i10, i11 - 1, i - 1, it, 3) = Val(Ptxt1) 'Pressure
                    0-3 blocks
            If it = 0 And i10 = 0 And i = 1 And i11 = 1 Then
                ulmin(3) = Val(Ptxt1) : ulmax(3) = Val(Ptxt1)
            End If
            If Val(Ptxt1) < ulmin(3) Then ulmin(3) = Val(Ptxt1) :
                ulfiltmin(3) = Val(Ptxt1)
            If Val(Ptxt1) > ulmax(3) Then ulmax(3) = Val(Ptxt1) :
                ulfiltmax(3) = Val(Ptxt1)
            Next
        Next
    Next i10
    'fifth block
    For i11 = 1 To NL
        For i = 1 To NQ
            Ptxt1 = sr.ReadLine()

```

```

                Vx4(NL - i11, i - 1, it, 3) = Val(Ptxt1) 'Pressure fifth
                                                    block
            Next
        Next
        sr.Close()
    Next
End Sub

```

### ***Reading second invariant of the velocity gradient, Q:***

---

```

Sub Q()
Me.Text = "Reading Q"
For it = 0 To nt - 1
    it1 = t0 + it * dt
    PNameFileM = Ptxt2 & Ptxt3 & "\" & it1 & ".Q"
    Dim sr As StreamReader = New StreamReader(PNameFileM)
    Do While Ptxt1 <> nn
        Ptxt1 = sr.ReadLine()
    Loop
    Ptxt1 = sr.ReadLine() 'ready to begin with useful data
    For i10 = 0 To 3 'first 4 blocks
        For i = 1 To NR
            For i11 = 1 To NQ
                Ptxt1 = sr.ReadLine()
                Vx(i10, i11 - 1, i - 1, it, 5) = Val(Ptxt1) 'Q 0-3
                                                    blocks
                If it = 0 And i10 = 0 And i = 1 And i11 = 1 Then
                    ulmin(5) = Val(Ptxt1) : ulmax(5) = Val(Ptxt1)
                End If
                If Val(Ptxt1) < ulmin(5) Then ulmin(5) = Val(Ptxt1) :
                    ulfiltmin(5) = Val(Ptxt1)
                If Val(Ptxt1) > ulmax(5) Then ulmax(5) = Val(Ptxt1) :
                    ulfiltmax(5) = Val(Ptxt1)
            Next
        Next
    Next i10
    'fifth block
    For i11 = 1 To NL
        For i = 1 To NQ
            Ptxt1 = sr.ReadLine()
            Vx4(NL - i11, i - 1, it, 5) = Val(Ptxt1) 'Q fifth block
        Next
    Next
    sr.Close()
Next
End Sub

```

### ***Reading vorticity field:***

---

```

Sub Vort()
Me.Text = "Reading vorticity field"
For it = 0 To nt - 1
    it1 = t0 + it * dt
    PNameFileM = Ptxt2 & Ptxt3 & "\" & it1 & "\vorticity"
    Dim sr As StreamReader = New StreamReader(PNameFileM)
    Do While Ptxt1 <> nn
        Ptxt1 = sr.ReadLine()
    Loop
    Ptxt1 = sr.ReadLine() 'ready to begin with useful data
    For i10 = 0 To 3 'first 4 blocks
        For i = 1 To NR
            For i11 = 1 To NQ
                Ptxt1 = sr.ReadLine()
                VortComp()
                Vx(i10, i11 - 1, i - 1, it, 6) = u1v
                Vx(i10, i11 - 1, i - 1, it, 7) = u2v
                Vx(i10, i11 - 1, i - 1, it, 8) = u3v
                Vx(i10, i11 - 1, i - 1, it, 9) = uuv
                If it = 0 And i10 = 0 And i = 1 And i11 = 1 Then
                    ulmin(6) = u1v : ulmax(6) = u1v
                    ulmin(7) = u2v : ulmax(7) = u2v
                    ulmin(8) = u3v : ulmax(8) = u3v
                    ulmin(9) = uuv : ulmax(9) = uuv
                End If
            Next
        Next
    Next i10
    'fifth block
    For i11 = 1 To NL
        For i = 1 To NQ
            Ptxt1 = sr.ReadLine()
            VortComp()
            Vx4(NL - i11, i - 1, it, 6) = u1v
            Vx4(NL - i11, i - 1, it, 7) = u2v
            Vx4(NL - i11, i - 1, it, 8) = u3v
            Vx4(NL - i11, i - 1, it, 9) = uuv
        Next
    Next
    sr.Close()
Next
Me.Text = Ptxt6
End Sub

```

### ***Calculating Q, 3 different approaches, block 4:***

---

```

Sub Qbb()
'Must select proper equation to calculate Q=Vx4b(i11 - 1, i - 1, it)
Me.Text = "Calculating Qbb (own)"
For it = 0 To nt - 1
    'fifth block
    For i11 = 2 To NL - 1
        For i = 2 To NQ - 1

```

```

Qb0(0) = Vx4(i11 - 1 - 1, i - 1, it, 0) 'u left (u i-1,j)
Qb0(1) = Vx4(i11 - 1 + 1, i - 1, it, 0) 'u right(u i+1,j)
Qb0(2) = Vx4(i11 - 1, i - 2, it, 0) 'u bottom (u i,j-1)
Qb0(3) = Vx4(i11 - 1, i, it, 0) 'u top (u i,j+1)
Qb0(4) = Vx4(i11 - 1 - 1, i - 1, it, 1) 'v left (v i-1,j)
Qb0(5) = Vx4(i11 - 1 + 1, i - 1, it, 1) 'v right(v i+1,j)
Qb0(6) = Vx4(i11 - 1, i - 2, it, 1) 'v bottom (v i,j-1)
Qb0(7) = Vx4(i11 - 1, i, it, 1) 'v top (v i,j+1)
Qb0(8) = Vx4(i11 - 1, i - 1, it, 0) 'u (u i,j)
Qb0(9) = Vx4(i11 - 1, i - 1, it, 1) 'v (v i,j)

Qb1(0) = -((Qb0(1) - Qb0(0)) ^ 2) / 8 / dx ^ 2
Qb1(1) = -(Qb0(3) - Qb0(2)) * (Qb0(5) - Qb0(4)) / 4
                                         / dx ^ 2
Qb1(2) = -((Qb0(7) - Qb0(6)) ^ 2) / 8 / dx ^ 2
'Vx4b(i11 - 1, i - 1, it) = Qb1(0) + Qb1(1) + Qb1(2)
'***this MATCHES perfectly with OpenFoam
'central difference and eqn Q=[-.5(dudx)2-dvdx.dudy-
                                         .5(dvdy)2], equations(3.35) (3.49)

Qb1(0) = (Qb0(1) - Qb0(0)) * (Qb0(7) - Qb0(6)) / 4
                                         / dx ^ 2
Qb1(1) = (Qb0(3) - Qb0(2)) * (Qb0(5) - Qb0(4)) / 4
                                         / dx ^ 2
'Vx4b(i11 - 1, i - 1, it) = Qb1(0) - Qb1(1)
'***this MATCHES perfectly with OpenFoam
'central difference and eqn Q=(dudx.dvdy-dvdx.dudy)
                                         Equations(3.36) (3.52)

Qb1(0) = (Qb0(1) - Qb0(8)) * (Qb0(7) - Qb0(9)) / dx ^ 2
Qb1(1) = (Qb0(3) - Qb0(8)) * (Qb0(5) - Qb0(9)) / dx ^ 2
'Vx4b(i11 - 1, i - 1, it) = Qb1(0) - Qb1(1)
'***this is almost the same as OpenFoam
'forward difference and eqn Q=(dudx.dvdy-dvdx.dudy)
                                         Equation (3.36)
'*** ** ** central difference and eqn Q=(-.5d2u2dx2-
.5d2v2dy2-d2uvdxdy)
                                         Equations(3.37) (3.54)
Qb0(0) = Vx4(i11 - 1 - 1, i - 1, it, 0) 'u left (u i-1,j)
Qb0(1) = Vx4(i11 - 1 + 1, i - 1, it, 0) 'u right(u i+1,j)
Qb0(2) = Vx4(i11 - 1, i - 2, it, 0) 'u bottom (u i,j-1)
Qb0(3) = Vx4(i11 - 1, i, it, 0) 'u top (u i,j+1)
Qb0(4) = Vx4(i11 - 1 - 1, i - 1, it, 1) 'v left (v i-1,j)
Qb0(5) = Vx4(i11 - 1 + 1, i - 1, it, 1) 'v right(v i+1,j)
Qb0(6) = Vx4(i11 - 1, i - 2, it, 1) 'v bottom (v i,j-1)
Qb0(7) = Vx4(i11 - 1, i, it, 1) 'v top (v i,j+1)
'*** ** ** continue
Qb0(8) = Vx4(i11 - 1, i - 1, it, 0) 'u (u i,j)
Qb0(9) = Vx4(i11 - 1, i - 1, it, 1) 'v (v i,j)
'*** ** ** continue
Qb0(10) = Vx4(i11 - 1 + 1, i, it, 0) 'u (u i+1,j+1)
Qb0(11) = Vx4(i11 - 1 - 1, i, it, 0) 'u (u i-1,j+1)
Qb0(12) = Vx4(i11 - 1 - 1, i - 2, it, 0) 'u (u i-1,j-1)
Qb0(13) = Vx4(i11 - 1 + 1, i - 2, it, 0) 'u (u i+1,j-1)
Qb0(14) = Vx4(i11 - 1 + 1, i, it, 1) 'v (v i+1,j+1)
Qb0(15) = Vx4(i11 - 1 - 1, i, it, 1) 'v (v i-1,j+1)

```

```

Qb0(16) = Vx4(i11 - 1 - 1, i - 2, it, 1) 'v (v i-1,j-1)
Qb0(17) = Vx4(i11 - 1 + 1, i - 2, it, 1) 'v (v i+1,j-1)
'*** ** ** continue
Qb0(18) = (Qb0(8) + Qb0(10) + Qb0(3) + Qb0(1)) / 4
'u (u i+1/2,j+1/2)
Qb0(19) = (Qb0(8) + Qb0(11) + Qb0(3) + Qb0(0)) / 4
'u (u i-1/2,j+1/2)
Qb0(20) = (Qb0(8) + Qb0(12) + Qb0(2) + Qb0(0)) / 4
'u (u i-1/2,j-1/2)
Qb0(21) = (Qb0(8) + Qb0(13) + Qb0(2) + Qb0(1)) / 4
'u (u i+1/2,j-1/2)
Qb0(22) = (Qb0(9) + Qb0(14) + Qb0(7) + Qb0(5)) / 4
'v (v i+1/2,j+1/2)
Qb0(23) = (Qb0(9) + Qb0(15) + Qb0(7) + Qb0(4)) / 4
'v (v i-1/2,j+1/2)
Qb0(24) = (Qb0(9) + Qb0(16) + Qb0(6) + Qb0(4)) / 4
'v (v i-1/2,j-1/2)
Qb0(25) = (Qb0(9) + Qb0(17) + Qb0(6) + Qb0(5)) / 4
'v (v i+1/2,j-1/2)

'*** ** ** continue
'*** ** ** Sum of a2Q
Qb2(0) = 0.5 * Qb0(1) ^ 2 + 0.5 * Qb0(0) ^ 2 - Qb0(8) ^ 2
Qb2(1) = 0.5 * Qb0(7) ^ 2 + 0.5 * Qb0(6) ^ 2 - Qb0(9) ^ 2
Qb2(2) = Qb0(18) * Qb0(22)
Qb2(3) = -Qb0(19) * Qb0(23)
Qb2(4) = Qb0(20) * Qb0(24)
Qb2(5) = -Qb0(21) * Qb0(25)
'*** ** ** continue
Qb2(6) = -Qb2(0) - Qb2(1) - Qb2(2) - Qb2(3) - Qb2(4) -
Qb2(5)

'*** ** ** continue
'Vx4b(i11 - 1, i - 1, it) = Qb2(6) / dx ^ 2
'*** ** ** end

If Vx4b(i11 - 1, i - 1, it) > ulmax(5) Then
    ulmax(5) = Vx4b(NL - i11, i - 1, it)
If Vx4b(i11 - 1, i - 1, it) < ulmin(5) Then
    ulmin(5) = Vx4b(NL - i11, i - 1, it)

Next
Next
Next
Me.Text = Ptxt6
End Sub

```

### **General geometry:**

---

```

Sub Geom() 'initial calculations for the grid geometry around the
cylinder (0-3 blocks)
For ig = 0 To NQ
    x1e = ig * dx
    c = Math.Sqrt((W / 2 - x1e) ^ 2 + (W / 2) ^ 2)
    x1i = W / 2 - rr * (W / 2 - x1e) / c
    y1e = 0

```

```

yli = W / 2 - rr * W / 2 / c
xl(ig, 0) = xle
yl(ig, 0) = yle
Sfj = 0 : Sfj2 = 0
PG01 = 1 / f 'r first respect to last (0.25)
PG02 = (1 / PG01) ^ (1 / (NQ - 1)) 'r consecutive values
PG03 = (c - rr) * (1 - PG02) / (1 - PG02 ^ NQ) 'first value
For jg = 1 To NR ' j geom (nodes)
    PG04 = PG03 * PG02 ^ (jg - 1) 'element by element
    xl(ig, jg) = xl(ig, jg - 1) + PG04 * (W / 2 - xle) / c
    yl(ig, jg) = yl(ig, jg - 1) + PG04 * W / 2 / c
Next
'*****taking the viscous force between the first row
                                and the boundary
ddrFiner(ig) = ((xl(ig, NR) - xl(ig, NR - 1)) ^ 2 + (yl(ig, NR) -
                                yl(ig, NR - 1)) ^ 2) ^ 0.5
'*****taking the viscous force between the two first
                                rows around the cylinder
ddrFiner(ig) = ((xl(ig, NR) - xl(ig, NR - 1)) ^ 2 + (yl(ig, NR) -
                                yl(ig, NR - 1)) ^ 2) ^ 0.5
Next
For ig = 0 To NQ - 1 'cell center
    For jg = 0 To NR - 1
        x10(ig, jg) = (xl(ig, jg) + xl(ig + 1, jg) + xl(ig, jg + 1) +
                                xl(ig + 1, jg + 1)) / 4
        y10(ig, jg) = (yl(ig, jg) + yl(ig + 1, jg) + yl(ig, jg + 1) +
                                yl(ig + 1, jg + 1)) / 4
    Next
Next
End Sub

```

### ***Calculating and envisioning flow around a cylinder***

---

```

Private Sub PictureBox1_Paint(ByVal sender As Object, ByVal e As
    System.Windows.Forms.PaintEventArgs) Handles PictureBox1.Paint
    QMaxMinInit()
    Dim lPen3a As New Pen(Drawing.Color.FromArgb(255, 255, 0, 0), 1)
    For blk = 0 To 3
        For j = 0 To NR - 1
            For i = 0 To NQ - 1
                ii = i : jj = j
                Dim lPen1a As New Pen(Drawing.Color.FromArgb(255, 0, 0,
                    0), 1)
                c1Filt = Int(255 * (Vx(blk, i, j, ttt, vvv) -
                    ulfiltmin(vvv)) / (ulfiltmax(vvv) - ulfiltmin(vvv)))
                If c1Filt <= PVto And c1Filt >= PVfrom Then
                    c1 = Int(255 * (Vx(blk, i, j, ttt, vvv) - ulmin(vvv))
                        / (ulmax(vvv) - ulmin(vvv)))
                    c2 = 255 - c1
                    MoireEffect1()
                    If CheckBox1.Checked Then MoireEffect2()
                    Dim bBrush As New
                        SolidBrush(Drawing.Color.FromArgb(255, c3, c2, c1))
                End If
            Next i
        Next j
    Next blk
End Sub

```

```

        blk000 = blk : blk0123()

        Dim point1 As New Point(xx1, yy1)
        Dim point2 As New Point(xx2, yy2)
        Dim point3 As New Point(xx3, yy3)
        Dim point4 As New Point(xx4, yy4)
        Dim points As Point() = {point1, point2, point3,
                                point4}

        e.Graphics.FillPolygon(bBrush, points)
    End If
Next
Next
Next
For j = 0 To NQ - 1
    For i = 0 To NL - 1
        ii = i : jj = j
        c1Filt = Int(255 * (Vx4(i, j, ttt, vvv) - ulfiltmin(vvv)) /
                    (ulfiltmax(vvv) - ulfiltmin(vvv)))
        If c1Filt <= PVto And c1Filt >= PVfrom Then
            c1 = Int(255 * (Vx4(i, j, ttt, vvv) - ulmin(vvv)) /
                    (ulmax(vvv) - ulmin(vvv)))

            c2 = 255 - c1
            MoireEffect1()
            If CheckBox1.Checked Then MoireEffect2()
            Dim bBrush As New SolidBrush(Drawing.Color.FromArgb(255,
                                                                c3, c2, c1))

            blk4()
            Dim rect As New Rectangle(xx1, yy1, zm, zm)
            e.Graphics.FillRectangle(bBrush, rect) 'draw no solid
        End If
    Next
Next
Next
For blk = 0 To 3
    For j = 0 To NR - 1
        For i = 0 To NQ - 1
            ii = i : jj = j
            Dim lPen1a As New Pen(Drawing.Color.FromArgb(255, 0, 0,
                                                         0), 1)

            c1Filt = Int(255 * (Vx(blk, i, j, ttt, vvv2) -
                                ulfiltmin(vvv2)) / (ulfiltmax(vvv2) - ulfiltmin(vvv2)))
            If c1Filt <= PVto2 And c1Filt >= PVfrom2 Then
                c2 = Int(255 * (Vx(blk, i, j, ttt, vvv2) -
                                ulmin(vvv2)) / (ulmax(vvv2) - ulmin(vvv2)))
                c1 = 255 - c2
                c2 = 200
                MoireEffect1()
                If CheckBox1.Checked Then MoireEffect22()
                Dim bBrush As New
                    SolidBrush(Drawing.Color.FromArgb(trns, c2, c1, c3))
                blk000 = blk : blk0123()
                Dim point1 As New Point(xx1, yy1)
                Dim point2 As New Point(xx2, yy2)
                Dim point3 As New Point(xx3, yy3)
                Dim point4 As New Point(xx4, yy4)
                Dim points As Point() = {point1, point2, point3,
                                        point4}
            End If
        Next
    Next
Next

```

```

        e.Graphics.FillPolygon(bBrush, points)
    End If
Next
Next
Next
For j = 0 To NQ - 1
    For i = 0 To NL - 1
        ii = i : jj = j
        c1Filt = Int(255 * (Vx4(i, j, ttt, vvv2) - ulfiltmin(vvv2)) /
                    (ulfiltmax(vvv2) - ulfiltmin(vvv2)))
        If c1Filt <= PVto2 And c1Filt >= PVfrom2 Then
            c2 = Int(255 * (Vx4(i, j, ttt, vvv2) - ulmin(vvv2)) /
                    (ulmax(vvv2) - ulmin(vvv2)))

            c1 = 255 - c2
            c2 = 200
            MoireEffect1()
            If CheckBox1.Checked Then MoireEffect22()
            Dim bBrush As New SolidBrush(Drawing.Color.FromArgb(trns,
                                                                c2, c1, c3))

            blk4()
            Dim rect As New Rectangle(xx1, yy1, zm, zm)
            e.Graphics.FillRectangle(bBrush, rect) 'draw no solid
        End If
    Next
Next
Next
'reference axis
e.Graphics.DrawLine(lPen3a, 2, zm * (NR - 1) + 2, zm * 10, zm * (NR -
                                                                1) + 2)
e.Graphics.DrawString("x", sfnt, Brushes.Red, zm * 10, zm * (NR - 1)
                                                                + 2 - 20)
e.Graphics.DrawLine(lPen3a, 2, zm * (NR - 1) + 2, 2, zm * (NR - 1 -
                                                                10) + 2)
e.Graphics.DrawString("y", sfnt, Brushes.Red, 2, zm * (NR - 1 - 10) +
                                                                2 - 20)

'DP.n/r fifth block integration Note this is simplified into this:
                                                                / (W/NQ) * (W/NQ) quadrangular elements
DPsum01 = 0 : DPsum02 = 0 : DPsum03 = 0 : DPsum04 = 0
DPrsum01 = 0 : DPrsum02 = 0 : DPrsum03 = 0 : DPrsum04 = 0
DPsumCC = 0 : DPrsumCC = 0
Vel2 = Vx(3, 40, 1, 0, 0)

'contour integrals
If CheckBox9.Checked Or RadioButton21.Checked Or
    RadioButton22.Checked Or RadioButton23.Checked Then 'draw point
    Dim lPen2a As New Pen(Drawing.Color.FromArgb(255, 200, 50, 0), 1)
    If Qx05 < NQ Then 'block 0123
        If bl090 > NQ Then bl090 = NQ - 3 'NQ - 1
        ii = bl050 : jj = bl090
        If bl040 = 0 Then blk0()
        If bl040 = 1 Then blk1()
        If bl040 = 2 Then blk2()
        If bl040 = 3 Then blk3()
        Qx05 = (xx1 + xx2 + xx3 + xx4) / 4 / zm
        Qy05 = (yy1 + yy2 + yy3 + yy4) / 4 / zm
        Dim bBrush As New SolidBrush(Drawing.Color.FromArgb(255, 255,
                                                                0, 0))
    End If
End If

```

```

        Dim point1 As New Point(xx1, yy1)
        Dim point2 As New Point(xx2, yy2)
        Dim point3 As New Point(xx3, yy3)
        Dim point4 As New Point(xx4, yy4)
        Dim points As Point() = {point1, point2, point3, point4}
        e.Graphics.FillPolygon(bBrush, points)
    End If
    xx1 = zm * Qx05 - zm / 2 ' selected area from
    yy1 = zm * Qy05 - zm / 2
    xx2 = zm 'selected area width and height
    yy2 = zm
    e.Graphics.DrawRectangle(lPen2a, xx1, yy1, xx2, yy2) 'draw the
                                                    point
End If
If RadioButton23.Checked = False Then 'integration on rectangle
    Qx030 = Qx03
    Qy030 = Qy03
    Qx040 = Qx04
    Qy040 = Qy04
Else 'integration on the entire domain
    Qx030 = 80 + 1 - 1
    Qy030 = 1
    Qx040 = 320 - 1 - Qx030
    Qy040 = 80 - 2
End If
'left and right block 4 (contour integrals)
For j = Qy030 To Qy040 + Qy030 - 1
    ii = Qx030 - NQ - 1 : jj = j 'ii is just outside at the left
    blk4()
    Qx01rrb()
    Qx01rrbl()
    Qx01rrc()
    If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And Not
        RadioButton23.Checked) And CheckBox7.Checked Then
        DPsum01 = DPsum01 + 1 / 2 / pi * Math.Log(1 / Qrr02) *
            (Vx4(ii, NQ - 1 - j, ttt, 3) - Vx4(ii + 1, NQ - 1 -
                j, ttt, 3)) / (Vel2) ^ 2 'left (out-rect. inside
                                                    block 5)
        DPsum01 = DPsum01 - 1 / 2 / pi * Vx4(ii, NQ - 1 - j, ttt,
            3) / (Vel2) ^ 2 * (Math.Log(1 / Qrr) - Math.Log(1 /
                Qrr01)) 'left (out-rect. inside block 5th)
    If CheckBox14.Checked Then
        Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
            250, 200), 1)
        e.Graphics.DrawLine(lPen2aa, Qx01r, Qy01b, Qx05 * zm,
            Qy05 * zm) 'draw lines
    End If
End If
ii = Qx040 + Qx030 - NQ : jj = j 'ii is just outside at the right
blk4()
Qx01rrb()
Qx01rrbr()
Qx01rrc()
If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And CheckBox8.Checked
    Then
        DPsum02 = DPsum02 + 1 / 2 / pi * Math.Log(1 / Qrr02) *
            (Vx4(ii, NQ - 1 - j, ttt, 3) - Vx4(ii - 1, NQ - 1 -

```

```

        j, ttt, 3)) / (Vel2) ^ 2 'right (out-rect. inside
        block 5th)
    DPrsum02 = DPrsum02 - 1 / 2 / pi * Vx4(ii, NQ - 1 - j, ttt,
        3) / (Vel2) ^ 2 * (Math.Log(1 / Qrr) - Math.Log(1 /
        Qrr01)) 'right (out-rect. inside block 5th)
    If CheckBox15.Checked Then
        Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
            250, 200), 1)
        e.Graphics.DrawLine(lPen2aa, Qx01r, Qy01b, Qx05 * zm,
            Qy05 * zm) 'draw lines
    End If
End If
Next

'top and bottom block 4 (contour integrals)
For i = Qx030 - NQ To Qx040 + Qx030 - NQ - 1
    ii = i : jj = Qy030 - 1 'jj is just outside at the top
    blk4()
    Qx01rrb()
    Qx01rrbt() 'top
    Qx01rrc()
    If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And CheckBox10.Checked
        Then
            DPsum03 = DPsum03 + 1 / 2 / pi * Math.Log(1 / Qrr02) *
                (Vx4(i, NQ - 1 - jj, ttt, 3) - Vx4(i, NQ - 1 - (jj +
                1), ttt, 3)) / (Vel2) ^ 2 'bottom (block 5th)
            DPrsum03 = DPrsum03 - 1 / 2 / pi * Vx4(i, NQ - 1 - jj, ttt,
                3) / (Vel2) ^ 2 * (Math.Log(1 / Qrr) - Math.Log(1 /
                Qrr01)) 'bottom (block 5)
            If CheckBox16.Checked Then
                Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
                    250, 200), 1)
                e.Graphics.DrawLine(lPen2aa, Qx01r, Qy01b, Qx05 * zm,
                    Qy05 * zm) 'draw lines
            End If
        End If
        ii = i : jj = Qy040 + Qy030 'jj is just outside at the bottom
        blk4()
        Qx01rrb()
        Qx01rrbb() 'bottom
        Qx01rrc()
        If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And CheckBox11.Checked
            Then
                DPsum04 = DPsum04 + 1 / 2 / pi * Math.Log(1 / Qrr02) *
                    (Vx4(i, NQ - 1 - jj, ttt, 3) - Vx4(i, NQ - 1 - (jj -
                    1), ttt, 3)) / (Vel2) ^ 2 'bottom (block 5th)
                DPrsum04 = DPrsum04 - 1 / 2 / pi * Vx4(i, NQ - 1 - jj, ttt,
                    3) / (Vel2) ^ 2 * (Math.Log(1 / Qrr) - Math.Log(1 /
                    Qrr01)) 'bottom (block 5th)
                If CheckBox17.Checked Then
                    Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
                        250, 200), 1)
                    e.Graphics.DrawLine(lPen2aa, Qx01r, Qy01b, Qx05 * zm,
                        Qy05 * zm) 'draw lines
                End If
            End If
        End If
Next

```

```

'blocks 0,1,2,3 (countour integrals)
For blk = 0 To 3 'on cylinder
  For i = 0 To NQ - 1
    ii = i : jj = NQ - 1
    blk000 = blk : blk0123()
    xc1 = (xx1 + xx2 + xx3 + xx4) / 4
    yc1 = (yy1 + yy2 + yy3 + yy4) / 4
    dL1 = ((xx1 - xx2) ^ 2 + (yy1 - yy2) ^ 2) ^ 0.5
    xc3 = (xx1 + xx2) / 2 : yc3 = (yy1 + yy2) / 2
    ii = i : jj = NQ - 2
    blk000 = blk : blk0123()
    xc2 = (xx1 + xx2 + xx3 + xx4) / 4
    yc2 = (yy1 + yy2 + yy3 + yy4) / 4
    dr1 = ((xc1 - xc2) ^ 2 + (yc1 - yc2) ^ 2) ^ 0.5
    Qrr = dx * Math.Sqrt((xc1 / zm - Qx05) ^ 2 + (yc1 / zm -
      Qy05) ^ 2) '(outside cell) distance to the point [m]
    Qrr01 = dx * Math.Sqrt((xc2 / zm - Qx05) ^ 2 + (yc2 / zm -
      Qy05) ^ 2) '(inside cell) distance to the point [m]
    Qx01rrc()
    If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And
      CheckBox12.Checked And RadioButton23.Checked Then
      DPsumCC = DPsumCC + 1 / 2 / pi * Math.Log(1 / Qrr02) *
        (Vx(blk, i, NQ - 1, ttt, 3) - Vx(blk, i, NQ - 2,
          ttt, 3)) / (Vel2) ^ 2 / dr1 * dL1 'cylinder
      DPrsumCC = DPrsumCC - 1 / 2 / pi * Vx(blk, i, NQ - 1,
        ttt, 3) / (Vel2) ^ 2 * (Math.Log(1 / Qrr) -
          Math.Log(1 / Qrr01)) / dr1 * dL1 'cylinder
    If CheckBox18.Checked Then
      Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60,
        200, 250, 200), 1)
      e.Graphics.DrawLine(lPen2aa, xc1, yc1, Qx05 * zm,
        Qy05 * zm) 'draw lines
    End If
  End If
Next
Next
For i = 0 To NQ - 1
  'bottom
  ii = i : jj = 0
  blk0() 'bottom just outside
  xc1 = (xx1 + xx2 + xx3 + xx4) / 4
  yc1 = (yy1 + yy2 + yy3 + yy4) / 4
  ii = i : jj = 1
  blk0() 'bottom just inside (fixed)
  xc2 = (xx1 + xx2 + xx3 + xx4) / 4
  yc2 = (yy1 + yy2 + yy3 + yy4) / 4
  dL1 = ((xx1 - xx2) ^ 2 + (yy1 - yy2) ^ 2) ^ 0.5 'horizontal
  distance
  xc3 = (xx1 + xx2) / 2 : yc3 = (yy1 + yy2) / 2
  If i < NQ / 2 Then
    ii = i + 1 : jj = 0
    blk0() 'bottom just outside to center value
    xc1b = (xx1 + xx2 + xx3 + xx4) / 4
    yc1b = (yy1 + yy2 + yy3 + yy4) / 4
    f1 = (xc1b - xc2) / (xc1b - xc1)
    P001 = f1 * Vx(0, i, 0, ttt, 3) + (1 - f1) * Vx(0, i + 1, 0,
      ttt, 3) 'replace external pressure
  End If
Next

```

```

Else
    ii = i - 1 : jj = 0
    blk0() 'bottom just outside to center value
    xc1b = (xx1 + xx2 + xx3 + xx4) / 4
    yc1b = (yy1 + yy2 + yy3 + yy4) / 4
    f1 = (xc1b - xc2) / (xc1b - xc1)
    P001 = f1 * Vx(0, i, 0, ttt, 3) + (1 - f1) * Vx(0, i - 1, 0,
        ttt, 3) 'replace external pressure
End If
dr1 = Math.Abs(yc1 - yc2) ' just vertical distance
Qrr = dx * Math.Sqrt((xc2 / zm - Qx05) ^ 2 + (yc1 / zm - Qy05) ^
    2) '(outside moved cell) distance to the point [m]
Qrr01 = dx * Math.Sqrt((xc2 / zm - Qx05) ^ 2 + (yc2 / zm - Qy05)
    ^ 2) '(inside cell) distance to the point [m]
Qx01rrc()
If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And CheckBox11.Checked
    And RadioButton23.Checked Then
    DPsum04 = DPsum04 + 1 / 2 / pi * Math.Log(1 / Qrr02) * (P001
        - Vx(0, i, 1, ttt, 3)) / (Vel2) ^ 2 / dr1 * dL1
        'bottom square
    DPrsum04 = DPrsum04 - 1 / 2 / pi * P001 / (Vel2) ^ 2 *
        (Math.Log(1 / Qrr) - Math.Log(1 / Qrr01)) / dr1 *
        dL1 'bottom square
    If CheckBox17.Checked Then
        Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
            250, 200), 1)
        e.Graphics.DrawLine(lPen2aa, xc2, yc1, Qx05 * zm, Qy05 *
            zm) 'draw lines
    End If
End If
'top
ii = i : jj = 0
blk2() 'top just outside
xc1 = (xx1 + xx2 + xx3 + xx4) / 4
yc1 = (yy1 + yy2 + yy3 + yy4) / 4
ii = i : jj = 1
blk2() 'top just inside (fixed)
xc2 = (xx1 + xx2 + xx3 + xx4) / 4
yc2 = (yy1 + yy2 + yy3 + yy4) / 4
dL1 = ((xx1 - xx2) ^ 2 + (yy1 - yy2) ^ 2) ^ 0.5
xc3 = (xx1 + xx2) / 2 : yc3 = (yy1 + yy2) / 2
If i < NQ / 2 Then
    ii = i + 1 : jj = 0
    blk2() 'top just outside (to center value)
    xc1b = (xx1 + xx2 + xx3 + xx4) / 4
    yc1b = (yy1 + yy2 + yy3 + yy4) / 4
    f1 = (xc1b - xc2) / (xc1b - xc1)
    P001 = f1 * Vx(2, i, 0, ttt, 3) + (1 - f1) * Vx(2, i + 1, 0,
        ttt, 3) 'replace external pressure
Else
    ii = i - 1 : jj = 0
    blk2() 'top just outside (to center value)
    xc1b = (xx1 + xx2 + xx3 + xx4) / 4
    yc1b = (yy1 + yy2 + yy3 + yy4) / 4
    f1 = (xc1b - xc2) / (xc1b - xc1)
    P001 = f1 * Vx(2, i, 0, ttt, 3) + (1 - f1) * Vx(2, i - 1, 0,
        ttt, 3) 'replace external pressure

```

```

End If
dr1 = Math.Abs(yc1 - yc2) ' just vertical distance
Qrr = dx * Math.Sqrt((xc2 / zm - Qx05) ^ 2 + (yc1 / zm - Qy05) ^
2) '(outside moved cell) distance to the point [m]
Qrr01 = dx * Math.Sqrt((xc2 / zm - Qx05) ^ 2 + (yc2 / zm - Qy05)
^ 2) '(inside cell) distance to the point [m]

Qx01rrrc()
If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And CheckBox10.Checked
And RadioButton23.Checked Then
    DPsum03 = DPsum03 + 1 / 2 / pi * Math.Log(1 / Qrr02) * (P001
- Vx(2, i, 1, ttt, 3)) / (Vel2) ^ 2 / dr1 * dL1
    'top square
    DPrsum03 = DPrsum03 - 1 / 2 / pi * P001 / (Vel2) ^ 2 *
(Math.Log(1 / Qrr) - Math.Log(1 / Qrr01)) / dr1 *
dL1 'top square
    If CheckBox16.Checked Then
        Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
250, 200), 1)
        e.Graphics.DrawLine(lPen2aa, xc2, yc1, Qx05 * zm, Qy05 *
zm) 'draw lines
    End If
End If
'left
ii = i : jj = 0
blk3() 'left just outside
xc1 = (xx1 + xx2 + xx3 + xx4) / 4
yc1 = (yy1 + yy2 + yy3 + yy4) / 4
ii = i : jj = 1
blk3() 'left just inside (fixed)
xc2 = (xx1 + xx2 + xx3 + xx4) / 4
yc2 = (yy1 + yy2 + yy3 + yy4) / 4
dL1 = ((xx1 - xx2) ^ 2 + (yy1 - yy2) ^ 2) ^ 0.5
xc3 = (xx1 + xx2) / 2 : yc3 = (yy1 + yy2) / 2
If i < NQ / 2 Then
    ii = i + 1 : jj = 0
    blk3() 'left just outside (to center value)
    xc1b = (xx1 + xx2 + xx3 + xx4) / 4
    yc1b = (yy1 + yy2 + yy3 + yy4) / 4
    f1 = (yc1b - yc2) / (yc1b - yc1)
    P001 = f1 * Vx(3, i, 0, ttt, 3) + (1 - f1) * Vx(3, i + 1, 0,
ttt, 3) 'replace external pressure
Else
    ii = i - 1 : jj = 0
    blk3() 'left just outside (to center value)
    xc1b = (xx1 + xx2 + xx3 + xx4) / 4
    yc1b = (yy1 + yy2 + yy3 + yy4) / 4
    f1 = (yc1b - yc2) / (yc1b - yc1)
    P001 = f1 * Vx(3, i, 0, ttt, 3) + (1 - f1) * Vx(3, i - 1, 0,
ttt, 3) 'replace external pressure
End If
dr1 = Math.Abs(xc1 - xc2) ' just horizontal distance
Qrr = dx * Math.Sqrt((xc1 / zm - Qx05) ^ 2 + (yc2 / zm - Qy05) ^
2) '(outside moved cell) distance to the point [m]
Qrr01 = dx * Math.Sqrt((xc2 / zm - Qx05) ^ 2 + (yc2 / zm - Qy05)
^ 2) '(inside cell) distance to the point [m]

```

```

Qx01rrc()
If Qrr02 <> 0 And Qrr <> 0 And Qrr01 <> 0 And CheckBox7.Checked
    And RadioButton23.Checked Then
    DPsum01 = DPsum01 + 1 / 2 / pi * Math.Log(1 / Qrr02) * (P001
        - Vx(3, i, 1, ttt, 3)) / (Vel2) ^ 2 / dr1 * dL1
        'left square
    DPrsum01 = DPrsum01 - 1 / 2 / pi * P001 / (Vel2) ^ 2 *
        (Math.Log(1 / Qrr) - Math.Log(1 / Qrr01)) / dr1 *
        dL1 'left square
    If CheckBox14.Checked Then
        Dim lPen2aa As New Pen(Drawing.Color.FromArgb(60, 200,
            250, 200), 1)
        e.Graphics.DrawLine(lPen2aa, xc1, yc2, Qx05 * zm, Qy05 *
            zm) 'draw lines
    End If
End If
Next
DPsum05 = DPsum01 + DPsum02 + DPsum03 + DPsum04
DPrsum05 = DPrsum01 + DPrsum02 + DPrsum03 + DPrsum04

'Q integration (integral over the rectangular or the entire domain)
If RadioButton21.Checked Or RadioButton22.Checked Then 'rectangle
    Dim lPen2a As New Pen(Drawing.Color.FromArgb(255, 200, 50, 0), 1)
    xx1 = zm * Qx03 'selected area from...
    yy1 = zm * Qy03
    xx2 = zm * Qx04 'selected area width and height
    yy2 = zm * Qy04
    e.Graphics.DrawRectangle(lPen2a, xx1, yy1, xx2, yy2) 'rectangle
End If

If CheckBox9.Checked Then
    Qsum03 = 0 'Sum of Element Areas at the intersected area
    Qsum04 = 0 'Sum of Q/Qrr*dA summation at the intersected area

    For blk = 0 To 3
        For j = 0 To NR - 1
            For i = 0 To NQ - 1
                ii = i : jj = j
                blk000 = blk : blk0123()

                TF = RadioButton23.Checked 'draw border (entire case)
                TF = (TF And (j < 1 Or j > 78) And blk <> 1) Or (TF
                    And (j > 78 Or (j < 1 And (i < 1 Or i > 78)))
                    And blk = 1)
                If TF Then
                    Dim lPen2aa As New
                        Pen(Drawing.Color.FromArgb(255, 200, 50, 0), 1)
                    Dim point1 As New Point(xx1, yy1)
                    Dim point2 As New Point(xx2, yy2)
                    Dim point3 As New Point(xx3, yy3)
                    Dim point4 As New Point(xx4, yy4)
                    Dim points As Point() = {point1, point2, point3,
                        point4}
                    e.Graphics.DrawPolygon(lPen2aa, points) 'draw
                        border (entire case)
                End If
            End For
        End For
    End For
End If

```

```

xa1 = (xx1 + xx2) / 2 : xa2 = (xx2 + xx3) / 2 : xa3 =
      (xx3 + xx4) / 2 : xa4 = (xx4 + xx1) / 2
ya1 = (yy1 + yy2) / 2 : ya2 = (yy2 + yy3) / 2 : ya3 =
      (yy3 + yy4) / 2 : ya4 = (yy4 + yy1) / 2
ha2 = ((xa4 - xa2) ^ 2 + (ya4 - ya2) ^ 2) ^ 0.5
'perpendicular distance:
ha1 = Math.Abs((xa4 - xa2) * (ya2 - ya3) - (xa2 -
      xa3) * (ya4 - ya2)) / ((xa4 - xa2) ^ 2
      + (ya4 - ya2) ^ 2) ^ 0.5
ha1 = ha1 + Math.Abs((xa4 - xa2) * (ya2 - ya1) - (xa2
      - xa1) * (ya4 - ya2)) / ((xa4 - xa2) ^
      2 + (ya4 - ya2) ^ 2) ^ 0.5
Q01 = (dx / zm) ^ 2 * ha1 * ha2 'Area [m2]
Q01 = Q01 / (2 * rr) ^ 2 ' dimensionless area
Q02 = Vx(blk, i, j, ttt, 3) 'Pressure
Q02 = Q02 / Vel2 ^ 2 'Non-D Pressure
Q03 = Vx(blk, i, j, ttt, 5) 'Q
Q03 = Q03 / Vel2 ^ 2 * (2 * rr) ^ 2 'No dimensional Q
Qx01p = (xx1 + xx2 + xx3 + xx4) / 4 'center polyg app
Qy01p = (yy1 + yy2 + yy3 + yy4) / 4
rp = ((yy2 - yy3) ^ 2 + (xx2 - xx3) ^ 2) ^ 0.5 / 2
      'approx. radio of polygon
Qx02 = zm * Qx05 - zm / 2
Qy02 = zm * Qy05 - zm / 2
'Qx02b = zm * Qx05 + 2 * zm
'Qy02b = zm * Qy05 + 2 * zm
Qx02b = zm * Qx05 + zm / 2
Qy02b = zm * Qy05 + zm / 2
rp2 = (((Qy02 + Qy02b) / 2 - Qy01p) ^ 2 + ((Qx02 +
      x02b) / 2 - Qx01p) ^ 2) ^ 0.5
TF2 = rp2 < 0.1 * rp
If TF2 Then 'point itself then no Q summation here
      Qpp0 = Q02 'Non-D pressure at the point
Else
      c1Filt = Int(255 * (Vx(blk, i, j, ttt, vvv) -
      ulfiltmin(vvv)) / (ulfiltmax(vvv) -
      ulfiltmin(vvv)))
      TF3 = c1Filt <= PVto And c1Filt >= PVfrom And
      RadioButton23.Checked = True And
      CheckBox6.Checked
      TF3 = TF3 And (j <> 0 Or (blk = 1 And Not ((i = 0
      And j = 0) Or (i = NQ - 1 And j = 0))))
      If TF3 Then
          Qrr = dx * Math.Sqrt((Qx01p / zm - Qx05) ^ 2
          + (Qy01p / zm - Qy05) ^ 2) 'distance to
          the point [m]
          Qrr = Qrr / (2 * rr) ' dimensionless radius
          Qsum03 = Qsum03 + Q01 'non-D Area summation
          Qsum04 = Qsum04 - 1 / pi * Q03 * Math.Log(1 /
          Qrr) * Q01 '1/pi*Q*ln(1/Qrr)*dA non-D
          summation
          If CheckBox13.Checked Then
              Dim lPen2aa As New
                  Pen(Drawing.Color.FromArgb(100,
                  200, 250, 100), 1)
              e.Graphics.DrawRectangle(lPen2aa,
                  Int(Qx01p - 1), Int(Qy01p - 1), 2, 2)

```

```

                                'draw rectangles
                                End If
                                MinMax001()
                                End If
                            End If
                        Next
                    Next
                Next
            For j = 0 To NQ - 1
                For i = 0 To NL - 1
                    Vel2 = Vx(3, 40, 1, 0, 0)
                    ii = i : jj = j
                    blk4()
                    TF = RadioButton23.Checked
                    'TF = TF And (i > 237 Or j < 2 Or j > 77)
                    TF = TF And (i > 238 Or j < 1 Or j > 78)
                    If TF Or CheckBox19.Checked Then
                        Dim lPen2aa As New Pen(Drawing.Color.FromArgb(255,
                                                                    200, 50, 0), 1)
                        e.Graphics.DrawRectangle(lPen2aa, xx1, yy1, zm, zm)
                        'draw border (entire case)
                    End If
                    Q01 = dx ^ 2 'Area [m2]
                    Q01 = Q01 / (2 * rr) ^ 2 ' dimensionless area
                    Q02 = Vx4(i, j, ttt, 3) 'Pressure
                    Q02 = Q02 / Vel2 ^ 2 'Non-D Pressure
                    '*****veryimportant*****
                    Q03 = Vx4(i, j, ttt, 5) 'Q from OpenFoam
                    'Q03 = Vx4b(i, j, ttt) 'Q own = Q OpenFoam
                    '*****veryimportant*****
                    Q03 = Q03 / Vel2 ^ 2 * (2 * rr) ^ 2 ' dimensionless Q
                    Qx01center()
                    Qx02 = zm * Qx05 - zm / 2
                    Qy02 = zm * Qy05 - zm / 2
                    'Qx02b = zm * Qx05 + 2 * zm
                    'Qy02b = zm * Qy05 + 2 * zm
                    Qx02b = zm * Qx05 + zm / 2
                    Qy02b = zm * Qy05 + zm / 2
                    If Qx01r >= Qx02 And Qx01r < Qx02b And Qy01r >= Qy02 And
                        Qy01r < Qy02b Then 'just in the point
                        z01 = 0
                        Qpp0 = Q02 'Non-D pressure at the point
                    Else
                        z01 = 1 'Sum Q at this location
                    End If
                    If Not (RadioButton23.Checked) Then 'integration of Q in
                                                                the rectangle
                        Qx02 = zm * Qx03
                        Qy02 = zm * Qy03
                        Qx02b = zm * (Qx03 + Qx04)
                        Qy02b = zm * (Qy03 + Qy04)
                    Else 'integration of Q in the entire domain
                        Qx02 = zm * NQ
                        Qy02 = zm
                        Qx02b = zm * (NL - 1 + NQ)
                        Qy02b = zm * (NQ - 1)
                    End If
                End If
            End If
        End If
    End Sub

```

```

If Qx01r >= Qx02 And Qx01r < Qx02b And Qy01r >= Qy02 And
    Qy01r < Qy02b Then
    c1Filt = Int(255 * (Vx4(i, j, ttt, vvv) -
        ulfiltmin(vvv)) / (ulfiltmax(vvv) -
        ulfiltmin(vvv)))
    If c1Filt <= PVto And c1Filt >= PVfrom And z01 = 1
        And CheckBox6.Checked Then
        Qrr = dx * Math.Sqrt((Qx01r / zm - Qx05) ^ 2 +
            (Qy01r / zm - Qy05) ^ 2) 'distance to the
            point [m]
        Qrr = Qrr / (2 * rr) ' dimensionless radius
        Qsum03 = Qsum03 + Q01 'non-D Area summation
        Qsum04 = Qsum04 - 1 / pi * Q03 * Math.Log(1 /
            Qrr) * Q01 '1/pi*Q*ln(1/Qrr)*dA non-D sum
        If CheckBox13.Checked Then
            Dim lPen2aa As New
                Pen(Drawing.Color.FromArgb(100,
                    200, 250, 100), 1)
            e.Graphics.DrawRectangle(lPen2aa, Int(Qx01r -
                1), Int(Qy01r - 1), 2, 2) 'rect
        End If
        MinMax001()
    End If
End If
Next
Next
End If

If CheckBox19.Checked Or CheckBox20.Checked Or CheckBox31.Checked
    Then 'grid & velocity vector blk 0123
    For blk = 0 To 3
        For j = 0 To NR - 1
            For i = 0 To NQ - 1
                ii = i : jj = j
                blk000 = blk : blk0123()

                If CheckBox19.Checked Then
                    Dim lPen2aa As New
                        Pen(Drawing.Color.FromArgb(105, 200,
                            50, 0), 1)

                    Dim point1 As New Point(xx1, yy1)
                    Dim point2 As New Point(xx2, yy2)
                    Dim point3 As New Point(xx3, yy3)
                    Dim point4 As New Point(xx4, yy4)
                    Dim points As Point() = {point1, point2, point3,
                        point4}
                    e.Graphics.DrawPolygon(lPen2aa, points) 'grid
                End If
                Qx01p = (xx1 + xx2 + xx3 + xx4) / 4 'cent. polyg. app
                Qy01p = (yy1 + yy2 + yy3 + yy4) / 4
                If i / SclVvD = Int(i / SclVvD) And j / SclVvD =
                    Int(j / SclVvD) And (CheckBox20.Checked Or
                    CheckBox31.Checked) Then 'velocity vectors
                    Vel3 = Vel2 * Vel3a / 100
                    MagVel = ((Vx(blk, i, j, ttt, 0) - Vel3) ^ 2 +
                        Vx(blk, i, j, ttt, 1) ^ 2) ^ 0.5 + 0.00001
                    xv1 = Qx01p + 1.5 * SclVv * zm * (Vx(blk, i, j,

```

```

        ttt, 0) - Vel3) / Vel2
yv1 = Qy01p - 1.5 * SclVv * zm * Vx(blk, i, j,
        ttt, 1) / Vel2
xv2 = Qx01p + 3 * zm * (Vx(blk, i, j, ttt, 0) -
        Vel3) / MagVel
yv2 = Qy01p - 3 * zm * Vx(blk, i, j, ttt, 1) /
        MagVel
Dim lPen4a As New Pen(Drawing.Color.FromArgb(200,
        255, 0, 0), 1)
Dim lPen4b As New Pen(Drawing.Color.FromArgb(100,
        255, 150, 0), 1)
Dim lPen4c As New Pen(Drawing.Color.FromArgb(200,
        0, 150, 255), 1)
Dim cBrush As New
        SolidBrush(Drawing.Color.FromArgb(200,
        255, 255, 0))
If CheckBox20.Checked Then
    e.Graphics.DrawEllipse(lPen4b, Qx01p - zm /
        4, Qy01p - zm / 4, zm / 2, zm / 2)
        'draw circle (blk 0123)
End If
If MagVel < CvgEddy And CheckBox31.Checked Then
    e.Graphics.FillEllipse(cBrush, Qx01p - zm /
        4, Qy01p - zm / 4, zm / 2, zm / 2)
        'draw circle (blk 0123)
End If
If MagVel * SclVv / 2 / Vel2 < 1 And
        CheckBox20.Checked Then
    e.Graphics.DrawLine(lPen4a, Qx01p, Qy01p,
        xv1, yv1) 'draw scaled velocity vector
Else
    If CheckBox20.Checked And Not
        CheckBox31.Checked Then
        e.Graphics.DrawLine(lPen4c, Qx01p, Qy01p,
            xv2, yv2) 'draw velocity vector
    End If
End If
End If
Next
Next
Next
For j = 0 To NQ - 1 'grid & velocity vector blk 4
    For i = 0 To NL - 1
        ii = i : jj = j
        blk4()
        TF = RadioButton23.Checked
        'TF = TF And (i > 237 Or j < 2 Or j > 77)
        TF = TF And (i > 238 Or j < 1 Or j > 78)
        If TF Or CheckBox19.Checked Then
            Dim lPen2aa As New Pen(Drawing.Color.FromArgb(105,
                200, 50, 0), 1)
            e.Graphics.DrawRectangle(lPen2aa, xx1, yy1, zm, zm)
                'draw border (entire case)
        End If
        Qx01center()
        If ((i + SclVvD / 2) / SclVvD = Int((i + SclVvD / 2) /
            SclVvD) And (j + SclVvD / 2) / SclVvD = Int((j +

```

```

        SclVvD / 2) / SclVvD) Or SclVvD = 1) And
        (CheckBox20.Checked Or CheckBox31.Checked) Then
        'velocity vectors
        '***doing Eulerian reference by varying a relative
            horizontal velocity
        'New Relative (moving) Velocity Magnitude
        MagVel = ((Vx4(i, j, ttt, 0) - Vel3) ^ 2 + Vx4(i, j,
            ttt, 1) ^ 2) ^ 0.5 + 0.00001
        xv1 = Qx01r + 1.5 * SclVv * zm * (Vx4(i, j, ttt, 0) -
            Vel3) / Vel2
        yv1 = Qy01r - 1.5 * SclVv * zm * Vx4(i, j, ttt, 1) /
            Vel2
        xv2 = Qx01r + 3 * zm * (Vx4(i, j, ttt, 0) - Vel3) /
            MagVel
        yv2 = Qy01r - 3 * zm * Vx4(i, j, ttt, 1) / MagVel

        Dim lPen4a As New Pen(Drawing.Color.FromArgb(200,
            255, 0, 0), 1)
        Dim lPen4b As New Pen(Drawing.Color.FromArgb(100,
            255, 150, 0), 1)
        Dim lPen4c As New Pen(Drawing.Color.FromArgb(200, 0,
            150, 255), 1)

        Dim cBrush As New
            SolidBrush(Drawing.Color.FromArgb(200, 255, 255, 0))
        If CheckBox20.Checked Then
            e.Graphics.DrawEllipse(lPen4b, Qx01r - zm / 4,
                Qy01r - zm / 4, zm / 2, zm / 2) 'circle (blk 4)
        End If
        If MagVel < CvgEddy And CheckBox31.Checked Then
            e.Graphics.FillEllipse(cBrush, Qx01r - zm / 4,
                Qy01r - zm / 4, zm / 2, zm / 2) 'circle (blk 4)
        End If
        If MagVel * SclVv / 2 / Vel2 < 1 And
            CheckBox20.Checked Then
            e.Graphics.DrawLine(lPen4a, Qx01r, Qy01r, xv1,
                yv1) 'draw scaled velocity vector
        Else
            If CheckBox20.Checked And Not CheckBox31.Checked
                Then
                e.Graphics.DrawLine(lPen4c, Qx01r, Qy01r,
                    xv2, yv2) 'draw velocity vector
            End If
        End If
    End If
Next
Next
End If

If CheckBox22.Checked Then 'Contribution from point
    ContFromPoint()
End If
If CFyn Or CFVyn Or CFSyn Or CheckBox21.Checked Or CheckBox22.Checked
    Then
    CFx = 0 : CFy = 0
    CFVM = 0 : CFVx = 0 : CFVy = 0
    dd = 2 * pi * rr / NQ / 4
    ddr = dx * f

```

```

For blk = 0 To 3
  If blkbol01(blk) Then
    alfa = 5 * pi / 4 + blk * pi / 2 - dd / 2 / rr
    jj = NR - 1
    'VB reference located at left-top corner of the object
    For i = 0 To NQ - 1
      ii = i
      alfa = alfa + dd / rr
      'Pressure
      CFPijj = Vx(blk, i, jj, ttt, 3) 'Pressure
      CFPx1 = CFPijj * Math.Cos(alfa) 'Pressure in x (hor.
        initial point (arrow to center) VBref)
      CFPy1 = -CFPijj * Math.Sin(alfa) 'Pressure in y (ver.
        initial point (arrow to center) VBref)
      'Q contribution Pressure
      CFPQijj = QCP(blk, i, ttt) 'Q contribution
      CFPQx1 = CFPQijj * Math.Cos(alfa) 'Pressure in x
        (hor. initial point (arrow to center) VBref)
      CFPQy1 = -CFPQijj * Math.Sin(alfa) 'Pressure in y
        (ver. initial point (arrow to center) VBref)
      'Pressure force
      CFijj = -CFPijj * dd 'F/rho/zleng [m3/s2];
      CFx = CFx + CFijj * Math.Cos(alfa) 'Press. forces sum
      CFy = CFy + CFijj * Math.Sin(alfa)

      'Viscous Force
      '*****FIRST taking the viscous force
        between the first row and the boundary
      'Vt1 = -Vx(blk, i, jj, ttt, 0) * Math.Sin(alfa)
        'tangential component of Vx
      'Vt2 = Vx(blk, i, jj, ttt, 1) * Math.Cos(alfa)
        'tangential component of Vy
      '*****SECOND taking the viscous force
        between the two first rows around the cylinder
      'Vt1 = -(Vx(blk, i, jj - 1, ttt, 0) - Vx(blk, i, jj,
        ttt, 0)) * Math.Sin(alfa) 'tangential component of Vx
      'Vt2 = (Vx(blk, i, jj - 1, ttt, 1) - Vx(blk, i, jj,
        ttt, 1)) * Math.Cos(alfa) 'tangential component of Vy
      '*****THIRD APPROACH to diminish finite
        difference error from solver algorithm probably
      Vt1 = -Vx(blk, i, jj - 1, ttt, 0) * Math.Sin(alfa)
        'tangential component of Vx
      Vt2 = Vx(blk, i, jj - 1, ttt, 1) * Math.Cos(alfa)
        'tangential component of Vy
      Vt3 = Vt1 + Vt2 ' tangential velocity
      Vt4 = -Vt3 * Math.Sin(alfa) 'x component of
        tangential velocity (hor. initial point
        (arrow to center) VBref)
      Vt5 = Vt3 * Math.Cos(alfa) 'y component of tangential
        velocity (ver. initial point (arrow to
        center) VBref)

      'CFVijj3 = Vt3 * kV * dd / (ddr / 2) 'total viscosity
        force
      'CFVijj4 = Vt4 * kV * dd / (ddr / 2) 'x viscosity
        force (hor. initial point (arrow to
        center) VBref)
    
```

```

'CFVijj5 = Vt5 * kV * dd / (ddr / 2) 'y viscosity
force (ver. initial point (arrow to
center) VBref)

'*****FIRST taking the viscous force
between the first row and the boundary
'CFVijj3 = Vt3 * kV * dd / (ddrFiner(i) / 2) 'total
viscosity force
'CFVijj4 = Vt4 * kV * dd / (ddrFiner(i) / 2) 'x
viscosity force (hor. initial point
(arrow to center) VBref)
'CFVijj5 = Vt5 * kV * dd / (ddrFiner(i) / 2) 'y
viscosity force (ver. initial point
(arrow to center) VBref)
'*****FIRST a little bit more precise
CFVijj3 = Vt3 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) / 4) 'total viscosity force
CFVijj4 = Vt4 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) / 4) 'x viscosity force (hor.
initial point (arrow to center) VBref)
CFVijj5 = Vt5 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) / 4) 'y viscosity force (ver.
initial point (arrow to center) VBref)

'*****SECOND taking the viscous force
between the two first rows around the cylinder
'CFVijj3 = Vt3 * kV * dd / ddrFiner(i) 'total
viscosity force
'CFVijj4 = Vt4 * kV * dd / ddrFiner(i) 'x viscosity
force (hor. initial point (arrow to
center) VBref)
'CFVijj5 = Vt5 * kV * dd / ddrFiner(i) 'y viscosity
force (ver. initial point (arrow to
center) VBref)
'*****SECOND a little bit more precise
'CFVijj3 = Vt3 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) / 2) 'total viscosity force
'CFVijj4 = Vt4 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) / 2) 'x viscosity force (hor.
initial point (arrow to center) VBref)
'CFVijj5 = Vt5 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) / 2) 'y viscosity force (ver.
initial point (arrow to center) VBref)

'*****THIRD directly the more precise
CFVijj3 = Vt3 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) * 3 / 4) 'total viscosity force
CFVijj4 = Vt4 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) * 3 / 4) 'x viscosity force
(hor. initial point (arrow to center)
CFVijj5 = Vt5 * kV * dd / ((ddrFiner(i) + ddrFiner(i)
+ 1)) * 3 / 4) 'y viscosity force
(ver. initial point (arrow to center)
CFVM = CFVM + CFVijj3
CFVx = CFVx + CFVijj4
CFVy = CFVy + CFVijj5 ' (VB)

```

```

If CheckBox21.Checked Then
    xx30 = zm * (NQ / 2 + NQ / 4 * SclP *
                Math.Cos(alfa))
    yy30 = zm * (NQ / 2 - NQ / 4 * SclP *
                Math.Sin(alfa))

    If i = 0 Then
        CFPx2 = CFPx1 : CFPy2 = CFPy1
        xx40 = xx30 : yy40 = yy30
    End If
    xx10 = xx30 + zm * (CFPx1 * CFPscl * SclP)
                'Pressure line end
    yy10 = yy30 + zm * (CFPy1 * CFPscl * SclP)
    xx20 = xx40 + zm * (CFPx2 * CFPscl * SclP)
                'Pressure line begin
    yy20 = yy40 + zm * (CFPy2 * CFPscl * SclP)
    Dim lPen1Ref As New
        Pen(Drawing.Color.FromArgb(255, 255, 255,
                0), 2)
    e.Graphics.DrawLine(lPen1Ref, xx30, yy30, xx40,
                yy40) 'circular reference
    Dim lPen1c As New Pen(Drawing.Color.FromArgb(155,
                50, 250, 250), 1)
    e.Graphics.DrawLine(lPen1c, xx10, yy10, xx20,
                yy20)
    e.Graphics.DrawLine(lPen1c, xx10, yy10, xx30,
                yy30)
    'e.Graphics.DrawLine(lPen1c, xx20, yy20, xx40,
                yy40)

    CFPx2 = CFPx1 : CFPy2 = CFPy1
    xx40 = xx30 : yy40 = yy30
End If

If CheckBox22.Checked Then 'Q contribution from point
                            on cylinder graph
    xxQ30 = zm * (NQ / 2 + NQ / 4 * SclP *
                Math.Cos(alfa))
    yyQ30 = zm * (NQ / 2 - NQ / 4 * SclP *
                Math.Sin(alfa))

    If i = 0 Then
        CFPQx2 = CFPQx1 : CFPQy2 = CFPQy1
        xxQ40 = xxQ30 : yyQ40 = yyQ30
    End If
    xxQ10 = xxQ30 + zm * (CFPQx1 * CFPscl * SclP) 'Q
                contribution line end
    yyQ10 = yyQ30 + zm * (CFPQy1 * CFPscl * SclP)
    xxQ20 = xxQ40 + zm * (CFPQx2 * CFPscl * SclP) 'Q
                contribution line begin
    yyQ20 = yyQ40 + zm * (CFPQy2 * CFPscl * SclP)
    Dim lPen1Ref As New
        Pen(Drawing.Color.FromArgb(255, 255, 255, 0), 2)
    e.Graphics.DrawLine(lPen1Ref, xxQ30, yyQ30,
                xxQ40, yyQ40) 'circular reference
    Dim lPen1c As New Pen(Drawing.Color.FromArgb(155,
                250, 150, 0), 1)
    e.Graphics.DrawLine(lPen1c, xxQ10, yyQ10, xxQ20,
                yyQ20)
    e.Graphics.DrawLine(lPen1c, xxQ10, yyQ10, xxQ30,

```

yyQ30)

```
CFPQx2 = CFPQx1 : CFPQy2 = CFPQy1
xxQ40 = xxQ30 : yyQ40 = yyQ30
End If
Next
End If
Next
CFSx = CFx + CFVx
CFSy = CFy + CFVy
CF = Math.Sqrt(CFx ^ 2 + CFy ^ 2)
CFV = Math.Sqrt(CFVx ^ 2 + CFVy ^ 2)
CFS = Math.Sqrt(CFSx ^ 2 + CFSy ^ 2)
xx1 = zm * NQ / 2 ' cylinder center
yy1 = zm * (NR - NQ / 2)
xx2 = zm * (NQ / 2 - CFx * CFscl) 'Pressure force line end
yy2 = zm * (NR - NQ / 2 + CFy * CFscl)
xx3 = zm * (NQ / 2 - CFVx * CFscl) 'Viscous force line end
yy3 = zm * (NR - NQ / 2 + CFVy * CFscl)
xx4 = zm * (NQ / 2 - CFSx * CFscl) 'Sum force line end
yy4 = zm * (NR - NQ / 2 + CFSy * CFscl)
'before of this line all forces are in [m3/s2]
CFS = CFS / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless total
force
CF = CF / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless pressure
force
CFV = CFV / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless viscous
force
CFSx = CFSx / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless total
force
CFx = CFx / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless pressure
force
CFVx = CFVx / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless
viscous force
CFSy = CFSy / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless total
force
CFy = CFy / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless pressure
force
CFVy = CFVy / 0.5 / 2 / rr / Vel2 ^ 2 'get dimensionless
viscous force
If CFSyn Then 'total force
Dim lPen1c As New Pen(Drawing.Color.FromArgb(255, 50, 50,
50), 2)
Dim lPen1c2 As New Pen(Drawing.Color.FromArgb(255, 50, 50,
50), 1)
e.Graphics.DrawLine(lPen1c, xx1, yy1, xx4, yy4)
If Comp Then 'show components
e.Graphics.DrawLine(lPen1c2, xx1, yy1, xx4, yy1)
e.Graphics.DrawLine(lPen1c2, xx1, yy1, xx1, yy4)
e.Graphics.DrawLine(lPen1c2, xx4, yy1, xx4, yy4)
e.Graphics.DrawLine(lPen1c2, xx1, yy4, xx4, yy4)
End If
TextBox40.Text = Format(CFS, "#0.00")
TextBox56.Text = Format(CFSx, "#0.00")
TextBox59.Text = Format(CFSy, "#0.00")
End If
If CFyn Then 'pressure force
Dim lPen1a As New Pen(Drawing.Color.FromArgb(255, 255, 150,
```

```

100), 2)
Dim lPen1a2 As New Pen(Drawing.Color.FromArgb(255, 255, 150,
100), 1)
e.Graphics.DrawLine(lPen1a, xx1, yy1, xx2, yy2)
If Comp Then 'show components
    e.Graphics.DrawLine(lPen1a2, xx1, yy1, xx2, yy1)
    e.Graphics.DrawLine(lPen1a2, xx1, yy1, xx1, yy2)
    e.Graphics.DrawLine(lPen1a2, xx2, yy1, xx2, yy2)
    e.Graphics.DrawLine(lPen1a2, xx1, yy2, xx2, yy2)
End If
TextBox38.Text = Format(CF, "#0.00")
TextBox58.Text = Format(CFx, "#0.00")
TextBox61.Text = Format(CFy, "#0.00")
End If
If CFVyn Then 'viscous force
    Dim lPen1b As New Pen(Drawing.Color.FromArgb(255, 0, 0, 255),
2)
    Dim lPen1b2 As New Pen(Drawing.Color.FromArgb(255, 0, 0,
255), 1)
    e.Graphics.DrawLine(lPen1b, xx1, yy1, xx3, yy3)
    If Comp Then 'show components
        e.Graphics.DrawLine(lPen1b2, xx1, yy1, xx3, yy1)
        e.Graphics.DrawLine(lPen1b2, xx1, yy1, xx1, yy3)
        e.Graphics.DrawLine(lPen1b2, xx3, yy1, xx3, yy3)
        e.Graphics.DrawLine(lPen1b2, xx1, yy3, xx3, yy3)
    End If
    TextBox39.Text = Format(CFV, "#0.00")
    TextBox57.Text = Format(CFVx, "#0.00")
    TextBox60.Text = Format(CFVy, "#0.00")
End If
End If
fff(ttt, 0) = CFS : fff(ttt, 1) = CFSx : fff(ttt, 2) = CFSy :
fff(ttt, 3) = CF : fff(ttt, 4) = CFx
fff(ttt, 5) = CFy : fff(ttt, 6) = CFV : fff(ttt, 7) = CFVx : fff(ttt,
8) = CFVy
PictureBox2.Visible = False
PictureBox2.Visible = True
PictureBox2.Refresh()
End Sub

```

### ***Conversion to the unique reference system***

---

```

Sub blk0()
    xx1 = zm * x1(ii, jj) / dx
    yy1 = zm * (NR - y1(ii, jj) / dx)
    xx2 = zm * x1(ii + 1, jj) / dx
    yy2 = zm * (NR - y1(ii + 1, jj) / dx)
    xx3 = zm * x1(ii + 1, jj + 1) / dx
    yy3 = zm * (NR - y1(ii + 1, jj + 1) / dx)
    xx4 = zm * x1(ii, jj + 1) / dx
    yy4 = zm * (NR - y1(ii, jj + 1) / dx)
End Sub
Sub blk1()
    xx1 = zm * NQ - zm * y1(ii, jj) / dx

```

```

yy1 = zm * (NR - x1(ii, jj) / dx)
xx2 = zm * NQ - zm * y1(ii + 1, jj) / dx
yy2 = zm * (NR - x1(ii + 1, jj) / dx)
xx3 = zm * NQ - zm * y1(ii + 1, jj + 1) / dx
yy3 = zm * (NR - x1(ii + 1, jj + 1) / dx)
xx4 = zm * NQ - zm * y1(ii, jj + 1) / dx
yy4 = zm * (NR - x1(ii, jj + 1) / dx)
End Sub
Sub blk2()
xx1 = zm * NQ - zm * x1(ii, jj) / dx
yy1 = zm * (y1(ii, jj) / dx)
xx2 = zm * NQ - zm * x1(ii + 1, jj) / dx
yy2 = zm * (y1(ii + 1, jj) / dx)
xx3 = zm * NQ - zm * x1(ii + 1, jj + 1) / dx
yy3 = zm * (y1(ii + 1, jj + 1) / dx)
xx4 = zm * NQ - zm * x1(ii, jj + 1) / dx
yy4 = zm * (y1(ii, jj + 1) / dx)
End Sub
Sub blk3()
xx1 = zm * y1(ii, jj) / dx
yy1 = zm * (x1(ii, jj) / dx)
xx2 = zm * y1(ii + 1, jj) / dx
yy2 = zm * (x1(ii + 1, jj) / dx)
xx3 = zm * y1(ii + 1, jj + 1) / dx
yy3 = zm * (x1(ii + 1, jj + 1) / dx)
xx4 = zm * y1(ii, jj + 1) / dx
yy4 = zm * (x1(ii, jj + 1) / dx)
End Sub
Sub blk4()
xx1 = zm * NQ + zm * ii
yy1 = zm * (NQ - 1 - jj)
End Sub

```

### ***Drag and Lift pressure contributions from every point***

---

```

Sub ContFromPointTotal()
For itQ = 0 To nt - 1
Me.Text = "Calculating Contribution from point (own)" & " " & itQ
+ 1 & " of " & nt & " blk 0123"

For blkQ = 0 To 3
'For blkQ = 0 To 3
For jQ = 0 To NR - 2 'don't do in j=79
For iQ = 0 To NQ - 1
b1040 = blkQ : b1050 = iQ : b1090 = jQ
ii = iQ : jj = jQ : ttt = itQ
ContFromPoint0123()
ContFromPointAll()
DragLift()
'Store results
Vx(blkQ, iQ, jQ, itQ, 13) = CFxDL 'drag
Vy(blkQ, iQ, jQ, itQ, 14) = CFyDL 'lift
If itQ = 0 And blkQ = 0 And iQ = 0 And jQ = 0 Then
ulmin(13) = CFxDL : ulmax(13) = CFxDL
ulmin(14) = CFyDL : ulmax(14) = CFyDL

```

```

        End If

        If CFxDL > ulmax(13) Then ulmax(13) = CFxDL :
                                ulfiltmax(13) = CFxDL
        If CFxDL < ulmin(13) Then ulmin(13) = CFxDL :
                                ulfiltmin(13) = CFxDL
        If CFyDL > ulmax(14) Then ulmax(14) = CFyDL :
                                ulfiltmax(14) = CFyDL
        If CFyDL < ulmin(14) Then ulmin(14) = CFyDL :
                                ulfiltmin(14) = CFyDL

        Next
    Next
    Me.Text = "Calculating Contribution from point (own)" & " " & itQ
              + 1 & " of " & nt & " blk 4"

    'fifth block
    For i11 = 0 To NL - 1
        For i = 0 To NQ - 1
            b1450 = i11 : b1490 = i
            ii = i11 : jj = i
            ContFromPoint4()
            ContFromPointAll()
            DragLift()
            'Store results
            Vx4(i11, i, itQ, 13) = CFxDL 'drag
            Vx4(i11, i, itQ, 14) = CFyDL 'lift
            If CFxDL > ulmax(13) Then ulmax(13) = CFxDL :
                                    ulfiltmax(13) = CFxDL
            If CFxDL < ulmin(13) Then ulmin(13) = CFxDL :
                                    ulfiltmin(13) = CFxDL
            If CFyDL > ulmax(14) Then ulmax(14) = CFyDL :
                                    ulfiltmax(14) = CFyDL
            If CFyDL < ulmin(14) Then ulmin(14) = CFyDL :
                                    ulfiltmin(14) = CFyDL

        Next
    Next
    Me.Text = Ptxt6
End Sub

Sub ContFromPoint()
    'check the area and Q from the point
    If Qx02point <= 80 Then 'block 0, 1, 2 or 3 (Point data)
        ii = b1050 : jj = b1090
        ContFromPoint0123()
    Else 'block 4 (the point)
        b1450 = Int(Qx05) - NQ
        b1490 = NQ - 1 - Int(Qy05)
        ii = b1450 : jj = b1490
        ContFromPoint4()
    End If
    'Contribution from point
    ContFromPointAll()
End Sub

```

```

Sub ContFromPoint0123()
  blk040_0123()
  Qx01C = (xx1 + xx2 + xx3 + xx4) / 4
  Qy01C = (yy1 + yy2 + yy3 + yy4) / 4
  xa1 = (xx1 + xx2) / 2 : xa2 = (xx2 + xx3) / 2 : xa3 = (xx3 + xx4) / 2
                                     : xa4 = (xx4 + xx1) / 2
  ya1 = (yy1 + yy2) / 2 : ya2 = (yy2 + yy3) / 2 : ya3 = (yy3 + yy4) / 2
                                     : ya4 = (yy4 + yy1) / 2
  ha2 = ((xa4 - xa2) ^ 2 + (ya4 - ya2) ^ 2) ^ 0.5
  'perpendicular distance:
  ha1 = Math.Abs((xa4 - xa2) * (ya2 - ya3) - (xa2 - xa3) * (ya4 - ya2))
        / ((xa4 - xa2) ^ 2 + (ya4 - ya2) ^ 2) ^ 0.5
  ha1 = ha1 + Math.Abs((xa4 - xa2) * (ya2 - ya1) - (xa2 - xa1) * (ya4 -
        ya2)) / ((xa4 - xa2) ^ 2 + (ya4 - ya2) ^ 2) ^ 0.5
  Q01 = (dx / zm) ^ 2 * ha1 * ha2 'Area [m2]
  Q01 = Q01 / (2 * rr) ^ 2 ' dimensionless area
  Q03 = Vx(bl040, bl050, bl090, ttt, 5) 'Q
  Q03 = Q03 / Vel2 ^ 2 * (2 * rr) ^ 2 ' dimensionless Q
End Sub

```

```

Sub ContFromPoint4()
  blk4()
  Qx01C = xx1 + zm / 2
  Qy01C = yy1 + zm / 2
  Q01 = dx ^ 2 'Area [m2]
  Q01 = Q01 / (2 * rr) ^ 2 ' dimensionless area
  Q03 = Vx4(bl450, bl490, ttt, 5) 'Q from OpenFoam
  Q03 = Q03 / Vel2 ^ 2 * (2 * rr) ^ 2 ' dimensionless Q
End Sub

```

```

Sub ContFromPointAll()
  For blk = 0 To 3
    For i = 0 To NQ - 1
      ii = i : jj = 79
      blk000 = blk : blk0123()
      Qx01p = (xx1 + xx2 + xx3 + xx4) / 4
      Qy01p = (yy1 + yy2 + yy3 + yy4) / 4
      Qrr = dx * Math.Sqrt((Qx01p / zm - Qx01C / zm) ^ 2 + (Qy01p /
        zm - Qy01C / zm) ^ 2) 'distance to the point [m]
      Qrr = Qrr / (2 * rr) ' dimensionless radius
      QCP(blk, i, ttt) = -1 / pi * Q03 * Math.Log(1 / Qrr) * Q01
        '1/pi*Q*ln(1/Qrr)*dA non-D Q contribution from point
    Next
  Next
End Sub

```

```

Sub DragLift()
  CFxDL = 0 : CFyDL = 0
  dd = 2 * pi * rr / NQ / 4
  ddr = dx * f
  For blk = 0 To 3
    If blkbol01(blk) Then
      alfa = 5 * pi / 4 + blk * pi / 2 - dd / 2 / rr
      jj = NR - 1
      For i = 0 To NQ - 1
        ii = i
        alfa = alfa + dd / rr
      Next
    End If
  Next
End Sub

```

```

'Q contribution Pressure
CFPQijj = QCP(blk, i, ttt) 'Q contribution
CFPQx1 = CFPQijj * Math.Cos(alfa) 'Pressure in x (hor.
    initial point (arrow to center) VBref)
CFPQy1 = -CFPQijj * Math.Sin(alfa) 'Pressure in y (ver.
    initial point (arrow to center) VBref)
'Q DL force
CFijj = -CFPQijj * dd / 0.5 / 2 / rr 'F/rho/zleng [];
CFxDL = CFxDL + CFijj * Math.Cos(alfa) 'Q DL forces sum
CFyDL = CFyDL + CFijj * Math.Sin(alfa)
    Next
End If
Next
CFDL = Math.Sqrt(CFxDL ^ 2 + CFyDL ^ 2)
End Sub

```

## Appendix C. VerFlow-V.01: User's Manual

VerFlow-V.01 is a customized program developed to work quantitatively and *qualitatively* with specific OpenFoam simulation results of periodic laminar flow around a cylinder. The name VerFlow-V.01 includes the Spanish word “ver” which means “see” in English since the author is from Ecuador, where Spanish is his primary language. This is only considered an alpha release of where it is anticipated that future versions be published with enhanced capabilities. Comments are welcomed.

This User's Manual gives a complete guide through the customized characteristics of VerFlow-V.01. Although some of these characteristics will require entering into the code, most of them do not.

The program runs in Windows environment and was written in Visual Basic 2008.

### Window zones

---

When running the executable file of VerFlow-V.01, the user will see an interactively Graphical User Interface (GUI) where four zones are defined as shown in Figure C.1.

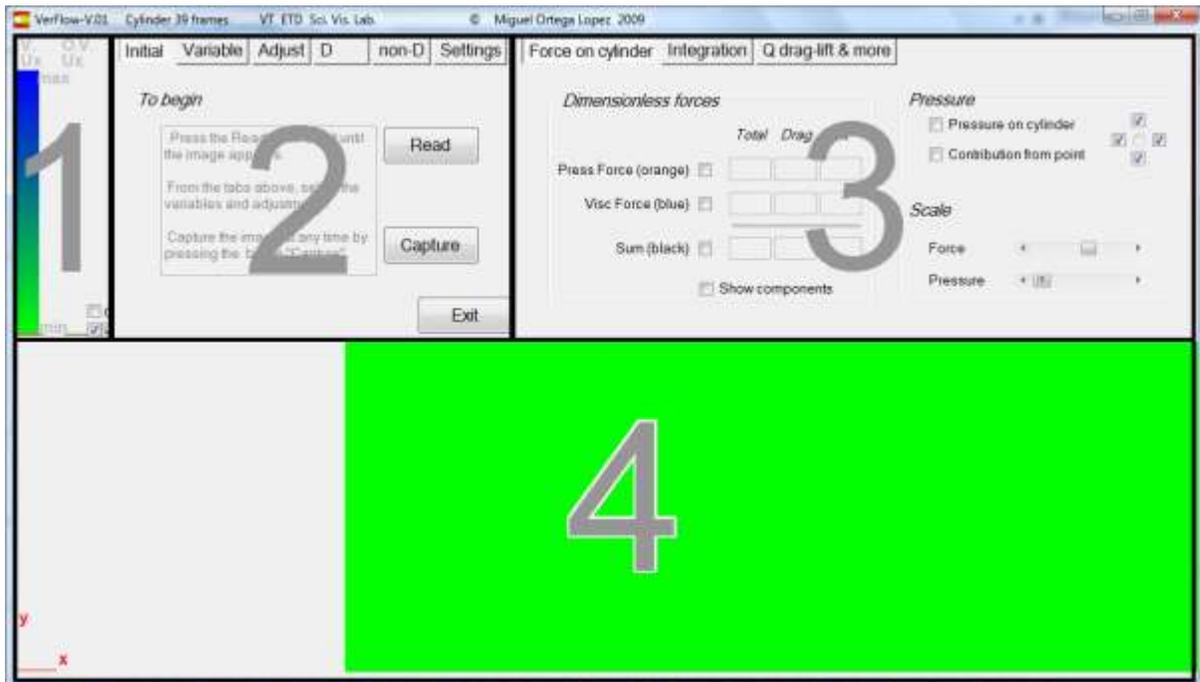


Figure C.1 VerFlow-V.01 window zones

A brief description of each zone is summarized here and the detail is shown later.

Zone 1: contains two color legend bars (sometimes only one color legend bar is visible as shown in Figure C.1) and some additional controls are also available: two check boxes to hide zones 2 or 3 and one check box to capture image sequence frames. This zone is always visible.

Zone 2: contains six tabs at the top. Each tab display particular information or different tools on this zone. This zone can be hidden by deselecting a box on zone 1 giving additional space to zone 4. When zone 2 is not visible, zone 3 is moved to the left over its space. This characteristic allows the user to work on screens with lower resolutions.

Zone 3: contains three tabs at the top which include quantitative tools. This zone can also be hidden from zone 1.

Zone 4: contains a customized *qualitative* instantaneous graph according with the settings established on the other zones. This zone can be extended when the window is maximized and also over zones 2 and 3 where they become invisible.

### **Zone 1**

---

This zone has two purposes: to show the color legend bars and to manage special additional controls, e.g. image capture, etc. A graphic explanation is given in Figure C.2.

The blue-green legend bar is displayed at the left of zone 1, while, not always visible, the red-yellow legend bar appears at the middle right of this zone. The blue-green legend bar corresponds to the *Variable V.* and the red-yellow legend bar corresponds to the *Overlapped Variable O.V.*

The user can select the option to display contour lines under the tab *Adjust* in zone 2. Contour lines can be incremented and displaced to coincide with specific values using the controls in the *Adjust* tab. Both color legend bars have also drawn a horizontal line, longer than contour lines, which coincides with the zero value of the variable.

A list of the variables is given under the *Variables tab in zone 2* section (see Table C.1). The *actual value* is the value of the *Variable V.* or *Variable O.V.* at the instantaneous mouse location

while the mouse cursor “point” is observed moving within zone 4. The actual value is simultaneously observed dynamically as a line on the color legend in zone 1 as a function of the point location on zone 4. The two color legend bars supports a “*filter*” characteristic, which is realized by selecting limits on the color legend in zone 1. Adjusting the limits on the color legend also limits the visible range of data observed in zone 4. This selection can also be accomplished using the *Adjust* tab in zone 2. Applicable to both variables, the selection of a range of visible values, below the maximum and above the minimum, can be accomplished by dragging the mouse pointer over the desired region in the color legend bar while pressing the left mouse button. When adjusting, a rectangle is observed with red lines which indicates the selection on the color legend bar and “*filters*” the data observed in zone 4.

The rectangle at the top right corner of this zone remains hidden until the user activates the option *Vel.Vectors* (Velocity vectors) in the *Settings* tab of zone 2. The vertical scroll bar controls the transition from a static to a moving reference frame.

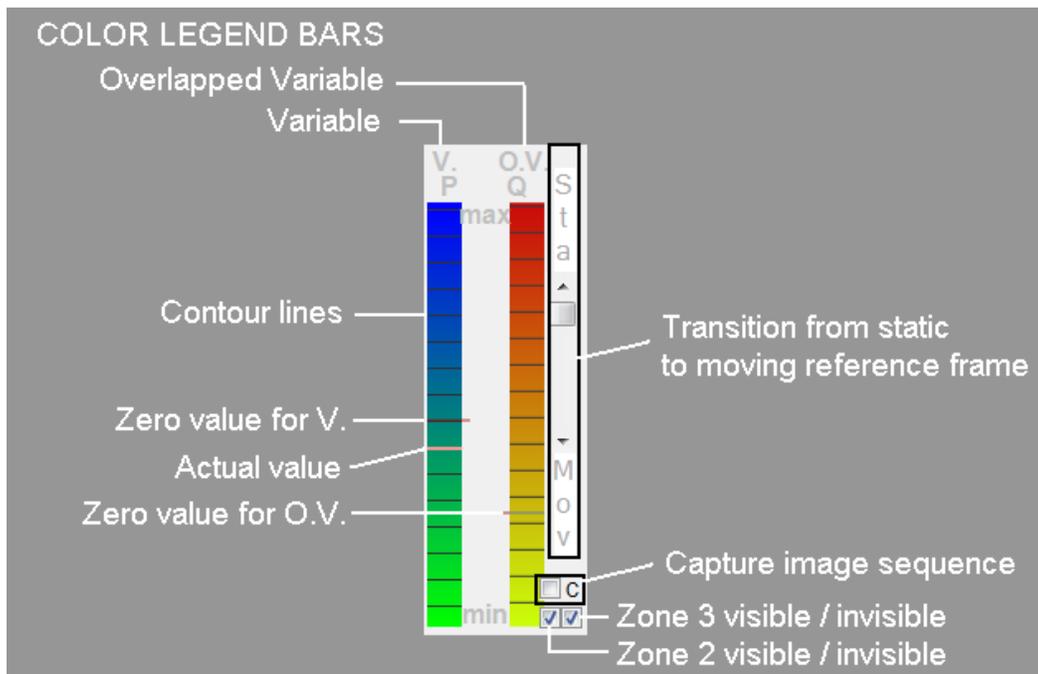


Figure C.2 VerFlow-V.01 color legend bar

The *capture image sequence* option shown in Figure C.2 is used to capture automatically an image sequence, which is stored in the *Sequence* folder, while the option *cycle* is also selected in

the *Settings* tab. This is similar to the *Capture* button in the *Initial* tab in zone 2 but for image sequence instead of individual pictures.

Two checked boxes, located at the bottom right corner of zone 1, define if zones 2 and 3 are visible or not. When zone 2 is invisible, zone 3 is moved to the left occupying its space. The advantage of using these two boxes is that there is more available space for zone 4, especially useful when looking for details in the flow in combination with other options customized in VerFlow-V.01.

### ***Initial tab in zone 2***

---

In Zone 2, under the tab *Initial*, basic initial instructions are given (see Figure C.3).

The information is initially stored in OpenFoam files and must be read by VerFlow-V.01 by pressing the button *Read*. It takes some time for the computer to read and store internally the numerical OpenFoam “data”. While reading, in the top of the windows it is displayed how the reading process advances by messages, i.e. *Reading velocity field*, *Reading pressure*, *Reading Q*, *Reading vorticity field* and *Calculating Qbb (own)*. Qbb (second invariant of the velocity gradient) is evaluated by VerFlow-V.01 in a region downstream for validation purposes when compared with Q given by OpenFoam.

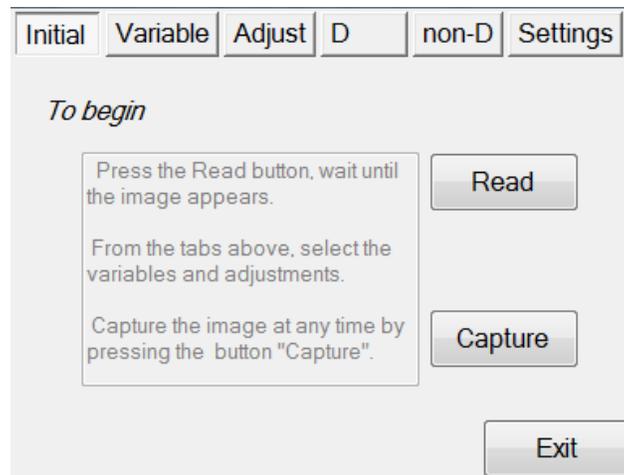


Figure C.3 *Initial* tab in Zone 2

Figure C.4 shows how the application appears when it finished reading the “data”. The variable displayed by default is the *Streamwise velocity  $U_x$*  (horizontal velocity).

The color legend bar in zone 1 shows the variable under the title,  $V$ , as  $U_x$ . The horizontal line in the color legend bar indicates the zero value for the variable  $V$ .

As the user moves the mouse pointer over the velocity field (or the visible variable) on zone 4, a simultaneous dynamic response appears as a line moving vertically in the color legend bar, which corresponds to the value for the variable  $V$  at the instantaneous mouse position in zone 4.

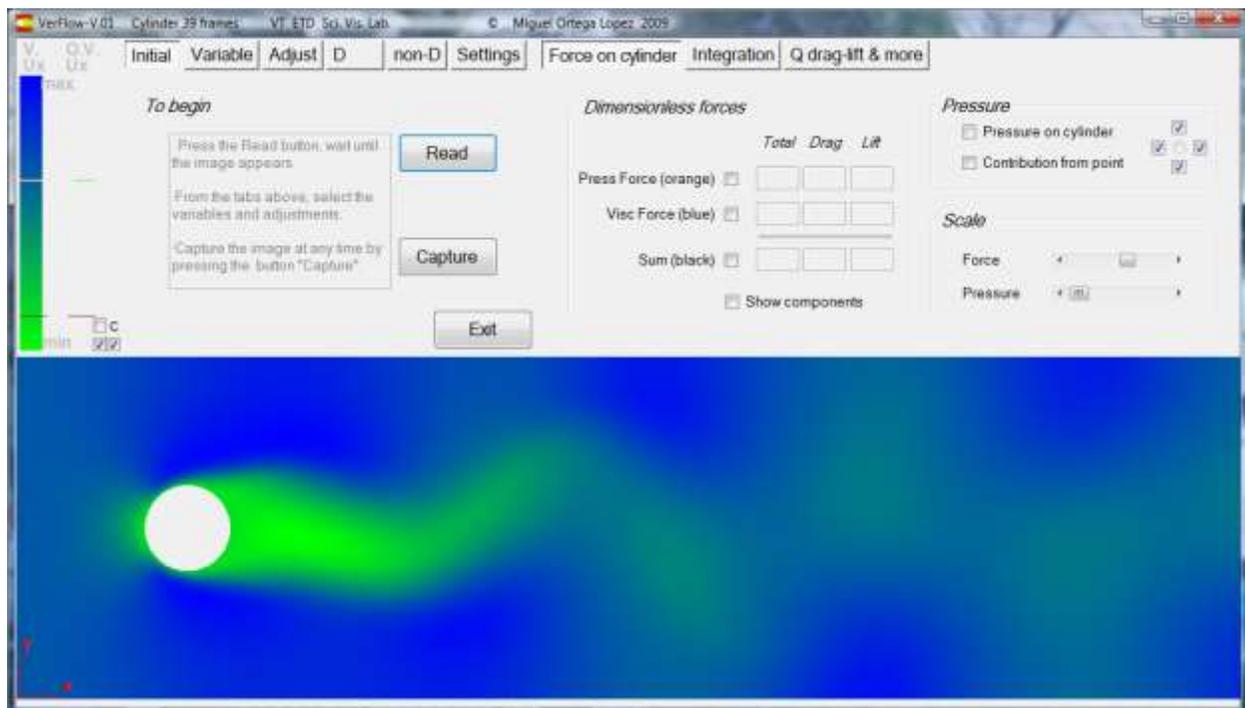


Figure C.4 Streamwise velocity field (default) displayed in zone 4 after reading

OpenFoam numerical simulated data includes velocities, vorticities, pressure and the second invariant of the velocity gradient,  $Q$ , in the entire domain. Although the problem is in two dimensions (2D), the velocity in the  $z$  direction and the vorticities in  $x$  and  $y$  are read by VerFlow-V.01, since a constant zero everywhere in these variables confirms part of the simulation results calculated by OpenFoam which always work in 3D (even 2D problems).

The button *Capture*, allows the user store the instantaneous displayed image as a jpg file in a folder named *Sequence*. The name of the file corresponds to actual frame number minus one, so

that if the first frame is displayed when *Capture* is pressed, *000.jpg* will be stored in the *Sequence* folder. Using this technique images for all frames follow the order given by the name of each file over one cycle and from this image sequence an animation can be generated.

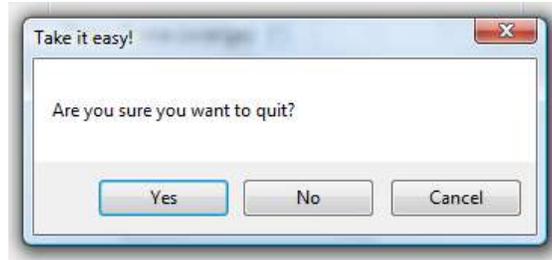


Figure C.5 Exit alert

The button *Exit* is used to finish the execution of the program. An alert message appears to ensure the user wants to close the application. This message is displayed in Figure C.5.

### ***Variable tab in zone 2***

---

When the *Variable* tab is selected in zone 2, the user can select the *Variable (V.)* and the *Overlapped Variable (O.V.)* from the corresponding option buttons. By default, both variables are initially set as *Ux*. Also, the overlapped variable is completely invisible and its transparency can be controlled under the *Adjust* tab in zone 2.

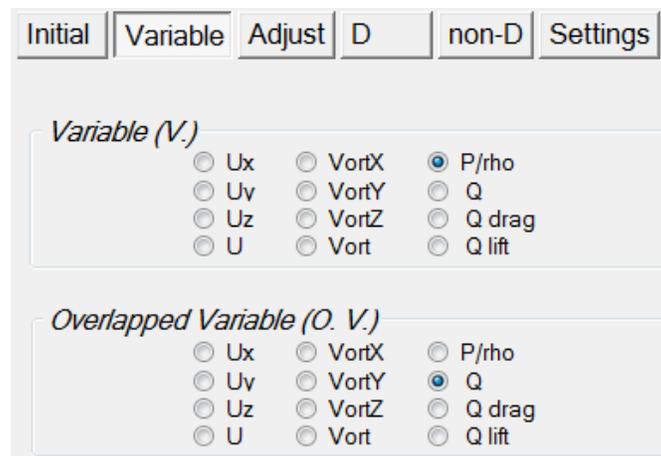


Figure C.6 Selection of variables

Selecting the pressure per unit density,  $P/\rho$ , as the *Variable (V.)* and the second invariant of the velocity gradient,  $Q$ , as the *Overlapped Variable (O.V.)*, zone 2 looks as shown in Figure C.6.

Table C.1 shows the list of variables, a general description and observations for each one.

<b>Variable</b>	<b>Description</b>	<b>Observations</b>
Ux	Streamwise Velocity (horizontal)	
Uy	Vertical Velocity	
Uz	z velocity	must be zero
U	Velocity magnitude	
VortX	Vorticity in x	must be zero
VortY	Vorticity in y	must be zero
VortZ	Vorticity in z	
Vort	Vorticity magnitude	
P/rho	Pressure per unit density	
Q	Second invariant of the velocity gradient	
Q drag	Drag contributions from every cell	calculation required
Q lift	Lift contributions from every cell	calculation required

Table C.1 Variables

The variables  $Q$  drag and  $Q$  lift are calculated using the  $Q$  drag-lift button in the  $Q$  drag-lift & more tab in zone 3, but the user must be aware that this operation takes about 1 minute per each frame and the operation should not be interrupted.

### ***Adjust tab in zone 2***

---

Figure C.7 shows zone 2 when the Adjust tab is selected.

The *Variable V.* and the *Overlapped Variable O.V.* are displayed in zone 4 when their values are between the limits adjusted by the user in the *Adjust* tab in zone 2. The selection of these limits is called the “*filter*” in VerFlow-V.01. *Filters* can also be selected from zone 1 by a directly drag the mouse pointer on the color legend bar while pushing the left mouse button.

In the *Adjust* tab in zone 2, the user can also choose a level of *transparency*. The horizontal scroll bar used for *transparency*, shows *Variable V*. in the default position which is at the left and the *Overlapped Variable O.V.* when its position is changed to the right. This allows the user to simultaneously observe two variables interactively as the animation is running.

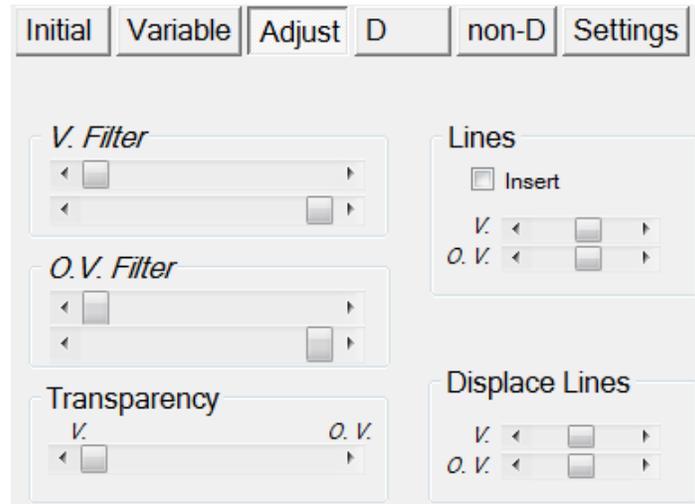


Figure C.7 *Adjust* tab in zone 2

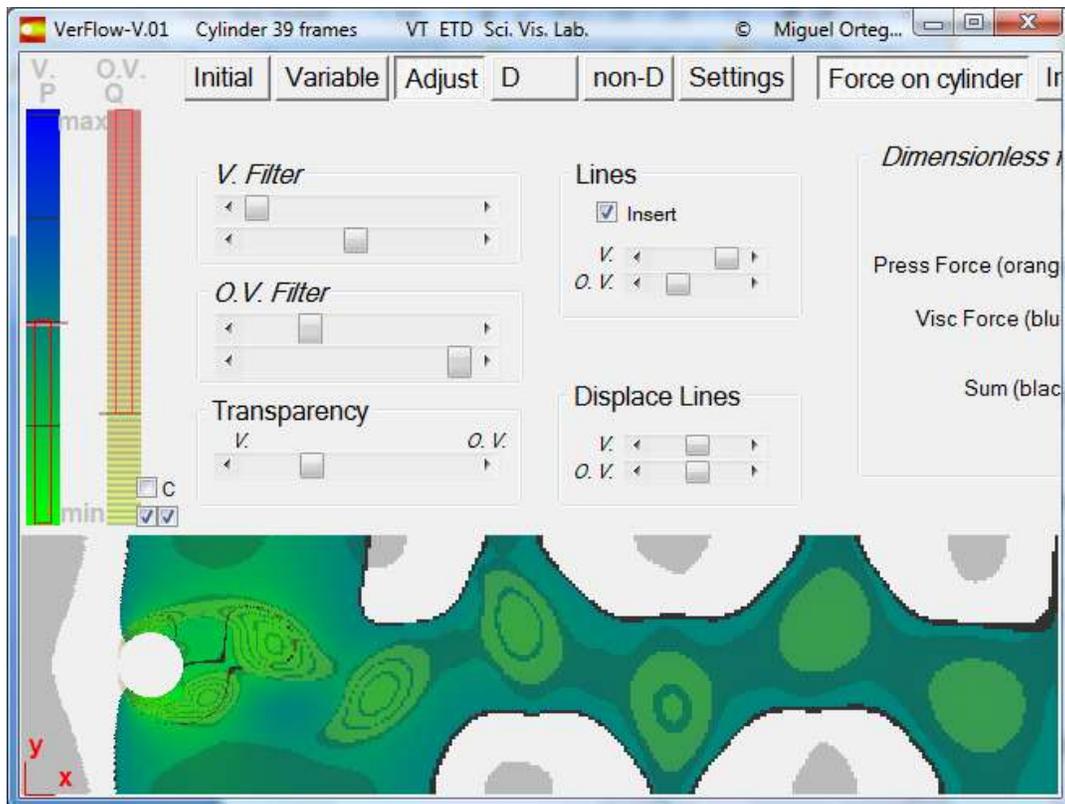


Figure C.8 Envisioning negative pressure and positive  $Q$  example

Contour lines can be displayed in zones 1 and 4 when the check box *Insert* is checked in the *Lines* label. The two horizontal scroll bars inside the *Lines* label allow the user to increase or decrease by a factor of two the number of contour lines uniformly distributed on the color legend bar. The two horizontal scroll bars, in the *Displace Lines* label, change the relative location of the contour lines up and down. This change is useful when the user requires a specific value to coincide with one contour line.

An example shows the negative pressure as the variable *V*. and positive *Q* as the Overlapped Variable O.V. (see Figure C.8). The transparency level is approximately 0.3, which means that pressure is more intense in color. The number of contour lines in the color legend bar is 4 for the pressure field but the rectangle with red boundaries indicates the filter on this variable which includes negative and zero values. The contour lines were interactively selected so that one contour line would coincide with zero pressure. The number of contour lines for *Q* is 64 and the corresponding rectangle marks limits on this region, e.g. to positive values including zero, which again is a “black” (it really looks gray for the transparency level) contour line.

### ***D and non-D tabs in zone 2***

---

Figure C.9 and C.10 shows the *D* and *non-D* tabs in zone 2 respectively. *D* refers to dimensional and non-D to dimensionless values of each variable.

Initial		Variable		Adjust		D		non-D		Settings	
Min	Max	Actual		Min	Max						
		Ux	0.0119	-0.0027	0.0195					[m/s]	
		Uy	-0.0001	-0.0098	0.0098					[m/s]	
		Uz	0.0000	0.0000	0.0000					[m/s]	
		U	0.0119	0.0000	0.0198					[m/s]	
		Vx	0.0000	0.0000	0.0000					[1/s]	
		Vy	0.0000	0.0000	0.0000					[1/s]	
		Vz	-0.0819	-55.5360	55.5356					[1/s]	
		V	0.0819	0.0000	55.5360					[1/s]	
		P	5.88E-05	-0.0001	0.0001					[m <sup>2</sup> /s <sup>2</sup> ]	
		Q	-0.042	-39.8953	109.0650					[1/s <sup>2</sup> ]	
		x	0.0013	0.0000	0.0800					[m]	
		y	0.0093	0.0000	0.0200					[m]	

Figure C.9 Dimensional values for an instantaneous mouse pointer location in zone 4

In the table located at the right of Figures C.9 and C.10, *Actual* refers to the instantaneous properties at the mouse pointer location in zone 4 while *Min* and *Max* refer to the minimum and maximum numerical values. In Figure C9, units are specified at the right for each variable.

These two tabs allow the user to interpret a dynamic quantitative and *qualitative* response in zone 2 of the values corresponding to the instantaneous location of the mouse pointer in zone 4. The quantitative response is the numerical data while the *qualitative* response is given by the lines in the gray zone at the left. Minimum values of each variable are located to the left of the gray zone while maximum to the right of the gray zone as indicated by *Min* and *Max*. Also drawn are dark lines (gray or brown) of zero values for each variable while the *actual* value is represented in purple.

Initial	Variable	Adjust	D	non-D	Settings		
	Ux	Actual	1.4531	Min	-0.2282	Max	1.6276
	Uy		-0.1482		-0.8163		0.8163
	Uz		0.0000		0.0000		0.0000
	U		1.4606		0.0000		1.6517
	Vx		0.0000		0.0000		0.0000
	Vy		0.0000		0.0000		0.0000
	Vz		-0.3218		-23.1485		23.1483
	V		0.3218		0.0000		23.1485
	P		-0.0555		-0.9397		0.9816
	Q		-0.062		-6.9313		18.9488
	Q drag		0.0000		0.0000		1.0000
	Q lift		0.0000		0.0000		1.0000
	x		7.8000		0.0000		16.0000
	y		3.4000		0.0000		4.0000

Figure C.10 Dimensionless values for an instantaneous mouse pointer location in zone 4

The actual *Q drag* and *Q lift* in Figure C.10 are zero because it is required first to run the *Q drag-lift* code to get these contributions to the drag and lift from the velocity field (see *Q drag-lift* tab in zone 3 and Sections 4.4.1 and 4.4.2 for more details).

The velocity in the *z* direction and vorticities in *x* and *y* are zero as indicated also in Table C.1, since only two dimensional “data” is evaluated and OpenFoam is in general, a three dimensional problem where one cell exists in the *z* direction, but data in the *z* direction is ignored.

## Settings tab in zone 2

---

In Figure C.11 the *Settings* tab is shown where four principal labels appear: *Time*, *Zoom*, *View* and *Move*.

The *Time* label contains a horizontal scroll bar and a *Cycle* check box. The horizontal scroll bar allows the user to select the frame in the simulation, a specific time which appear instead of *Time* below the horizontal scroll bar. The *Cycle* check box, when checked, activates an automatic sequential change of the time frames showing an interactive animation on zone 4 (this animated result generates dynamic quantitative (numerical) and qualitative (images) that relate to all other dependent tabs and zones).

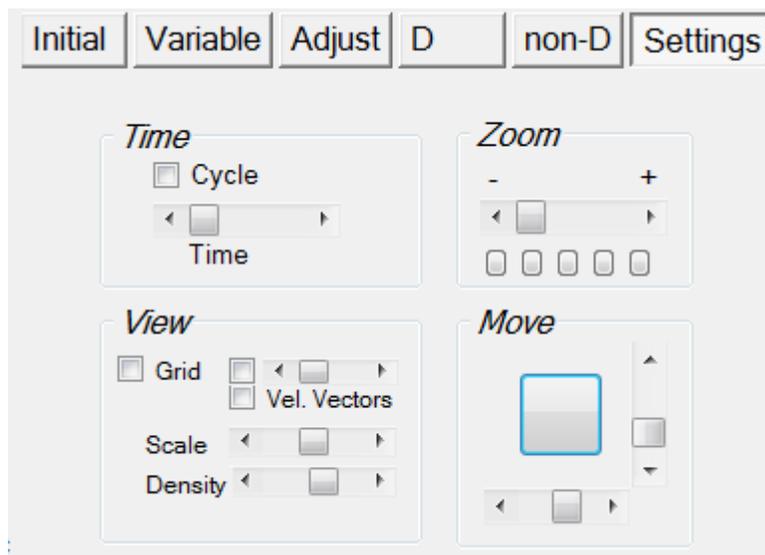


Figure C.11 Labels in the Settings tab in zone 4

The *View* label contains three check boxes. The first, marked as *Grid*, draws the grid on the flow region. The second, which has a horizontal scroll bar at the right, is useful to detect stagnation points (horizontal scroll bar set to the left) or zones (horizontal scroll bar set to the right), which are represented in yellow. The third, marked as *Vel. Vectors*, is used to select a representation of the velocity field as vectors over zone 4. The *Vel. Vectors* check box does not work alone, it interacts with the *Scale* scroll bar and the *Density* scroll bar below, and also with the vertical scroll bar that appears in zone 1, which changes from static to moving frame of reference. The *Scale* scroll bar changes the size of the vectors and “Scaled” *velocity vectors* are drawn in red

while “non-Scaled” velocity vectors are drawn in cyan. This distinction is completely necessary to envision the velocities in zones of interest, like the wake, where they are in the order of 0.01 (dimensionless). The *Density* scroll bar allows the selection of more or less cells in the grid for the represented velocity vectors. The vertical scroll bar with *Sta* and *Mov* labels in zone 1 can be used to change gradually from the static to the moving frames of reference and viceversa.

The *Zoom* label has a horizontal scroll bar used to magnify the image size displayed in zone 4 for a proper view. It is required also to work in the *Move* label to set an adequate image location. Five small buttons are located below the horizontal bar in the *Zoom* label. These buttons are used to set directly specific combinations of image magnification and location that the user should find useful.

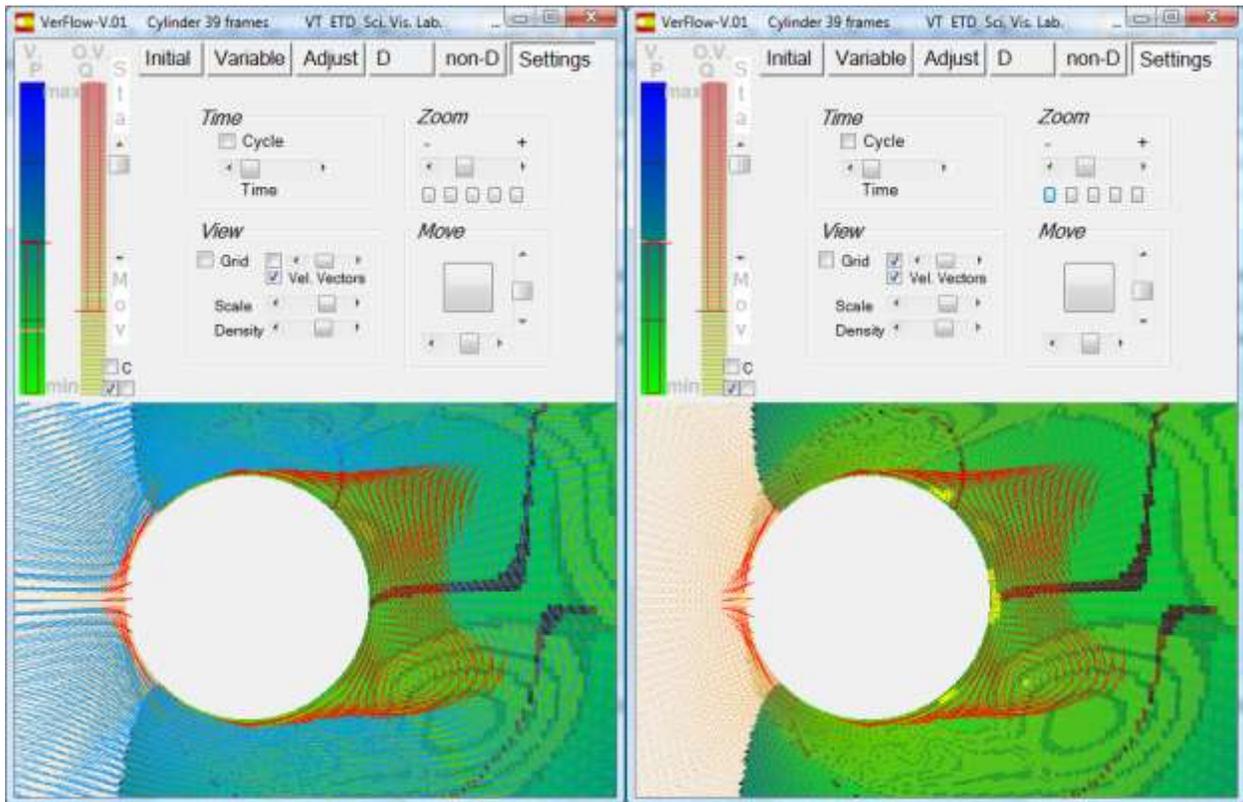


Figure C.12 *Settings* tab example. Note the effect of checking the box for stagnation points in the *View* label.

The *Move* label has a horizontal scroll bar, a vertical scroll bar and a button. The horizontal scroll bar moves the image in zone 4 horizontally while the vertical scroll bar moves the image

vertically. The button is used to set a position right below zone 1 regardless the magnification set in the *Zoom* label.

As a general recommendation, use the buttons in the *Zoom* label and after adjusting the image location, if required, use the scroll bars in the *Move* label.

An example of the use of the *Settings* tab is shown in Figure C.12. The difference between the left and right graphs in this figure was accomplished simply by checking the stagnation check box in the *View* label.

### ***Force on cylinder tab in zone 3***

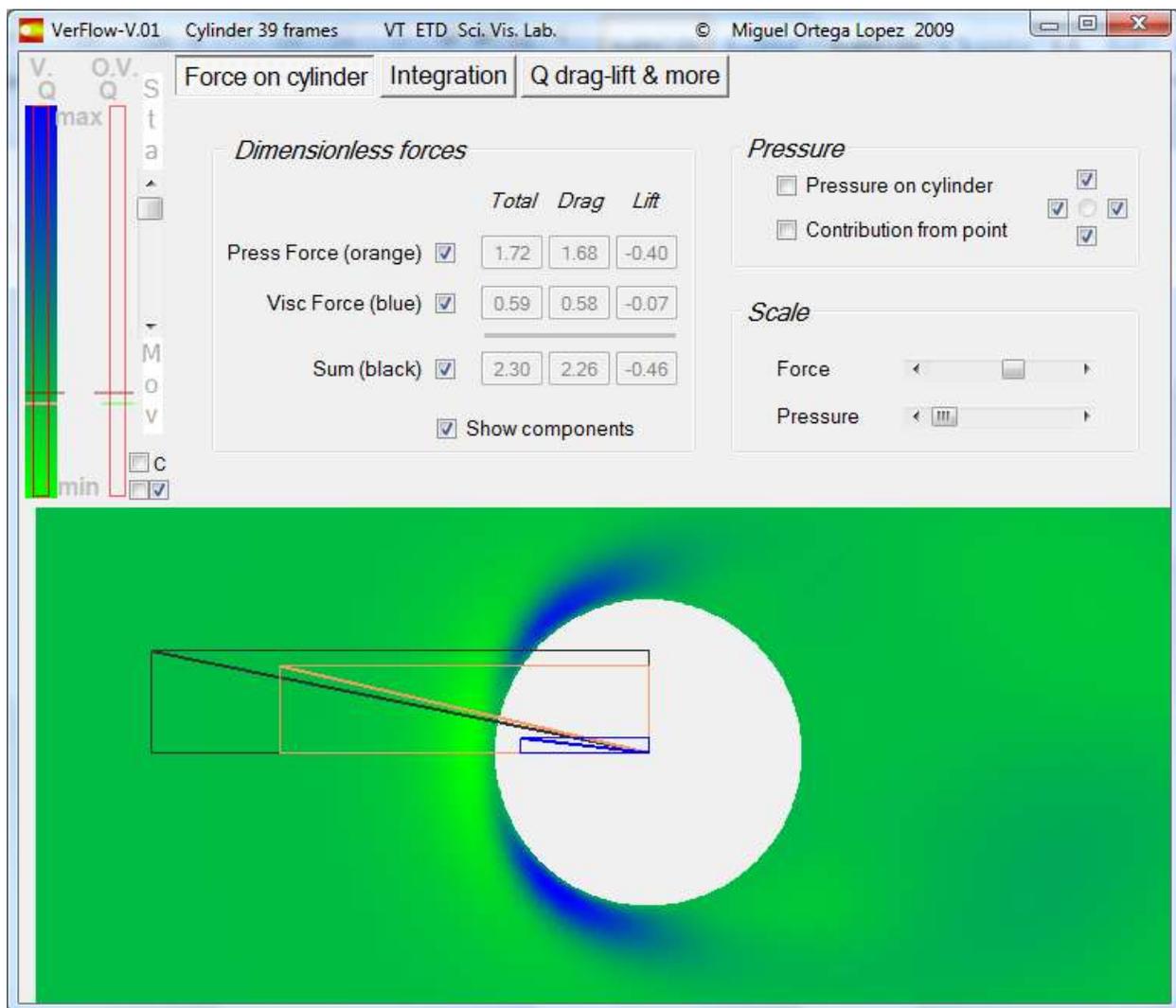


Figure C.13 Representation of pressure and viscous forces over  $Q$

Figure C.13 shows a *Dimensionless forces* label containing four check boxes which specify the calculations selected by the user in relation with the pressure and viscous forces on the cylinder. The equations implemented in VerFlow-V.01 for this purpose are described in Section 3.3 and the results are discussed in Section 4.1.

The quantitative data is shown in a table at the right of the check boxes and indicates the dimensionless coefficients as their drag and lift components and also the total for pressure forces, viscous forces and for the sum.

The color representation of each force is labeled at the left side of the corresponding check boxes: orange for pressure forces, blue for viscous forces and black for the sum. Forces are *qualitatively* represented as lines pointing to the cylinder center. The fourth check box, labeled as *Show components*, draws a rectangle which gives a visual representation of the drag and lift components.

Under the *Scale* label at the right in the *Force on cylinder* tab in zone 3, there is a horizontal scroll bar labeled as *Force*. Using this scroll bar, the user can change the size of the vectors for a convenient representation.

All forces are evaluated dynamically when the *Cycle* check box is selected in the *Settings* tab and the user can envision and explore the flow combining all the desired capabilities included in the program. When the program is processing a huge number of internal calculations over the entire domain and a proper view of each frame has been reached, the frames do not appear soon as desired. In this case it is better to use that information by automatically capturing an image sequence as discussed before in zone 1 and generating an animation using other software available such as Windows Movie Maker or Quick Time. This is applicable to all the customized capabilities introduced in VerFlow-V.01, since all variables can be superimposed giving a simultaneous multivariable representation. These superimposed multivariable animations are complex in the sense that they approach a “cognitive-limit” of the researcher. In this case *qualitative* use of images to analyze simulation results is insight, were as traditionally graphics has been used for presentation by implementing a less complex format.

In Figure C.14 the pressure distribution on the cylinder from the OpenFoam simulation “data” is shown when the check boxes in the *Dimensionless forces* label have been unchecked (this is

done to avoid distracting information not necessary in this discussion) and the *Pressure on cylinder* check box in the *Pressure* label has been checked. The positive pressure points outward the cylinder (from its boundary) and the negative pressure points towards the cylinder center (from its boundary). Results can be envisioned by quadrants which correspond to each block surrounding the cylinder according with Figure 2.2. These quadrants are selected in the four check boxes around the small circle in the *Pressure* label of the *Force on cylinder* tab at the right. The special distribution of these check boxes is directly related with the top, bottom, left and right quadrants. This is also applicable to the *Contribution from point* check box option below the *Pressure on cylinder* check box.

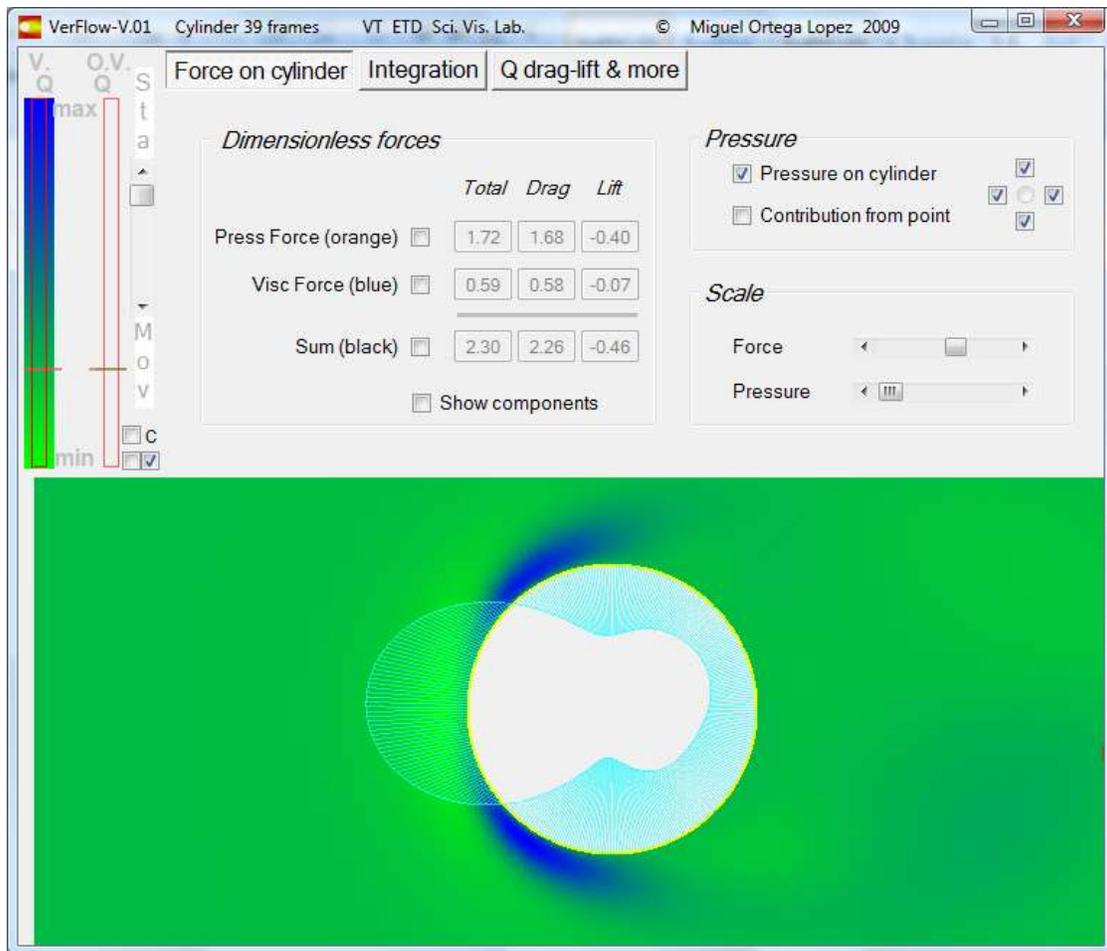


Figure C.14 Representation of pressure distribution from OpenFoam numerical “data”

The user can magnify the pressure distribution representation under the *Pressure* scroll bar in the *Scale* label, which affects the size of the circle (always centered at the cylinder center) and the pressure distribution for a proper view.

Figure C.15 shows how the *Point effect on the cylinder boundary* given in Section 4.3 is obtained in VerFlow-V.01. In this case the *Pressure on cylinder* check box was unchecked (to avoid unnecessary information in this discussion). The equation introduced in the code in VerFlow-V.01 for this calculation is 3.84a and is applied to each cell on the cylinder boundary. The user must select the *Contribution from point* check box in the *Pressure* label.

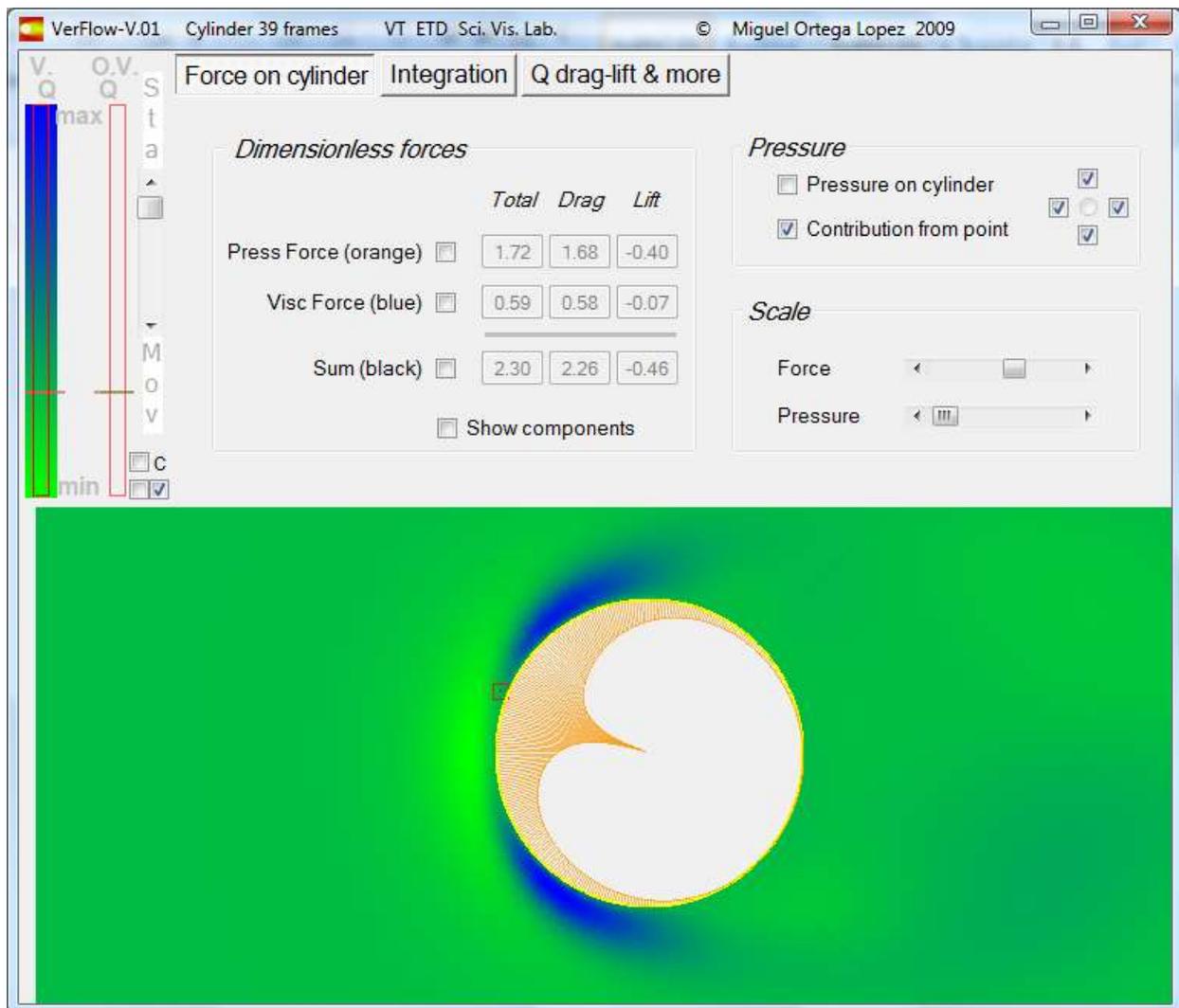


Figure C.15 Representation of pressure distribution originated from an arbitrary cell (enlighten in red)

The *Pressure* scroll bar in the *Scale* label and the four check boxes for the selection of the left, right, top and bottom quadrants in the *Pressure* label are still applicable as was explained for the *Pressure on cylinder* check box.

The selection of the point which generates such pressure distributions is realized by a left mouse button click applied directly on the represented flow in zone 4. Any point in the domain can be selected except in the first row on the cylinder boundary which generates an error because points P and Q are in the same location in one of the calculations along the cylinder boundary. For locations close to the cylinder boundary, the user can click directly inside the cylinder and the program automatically sets a cell in the third row from the cylinder boundary and in the direction specified by a line from the cylinder center to the point where the mouse click was initiated. The third row is selected to avoid the boundary problem and at the same time to be close enough to the cylinder boundary.

The positive pressure is again represented outward from the cylinder boundary and the negative pressure towards the center from the cylinder boundary.

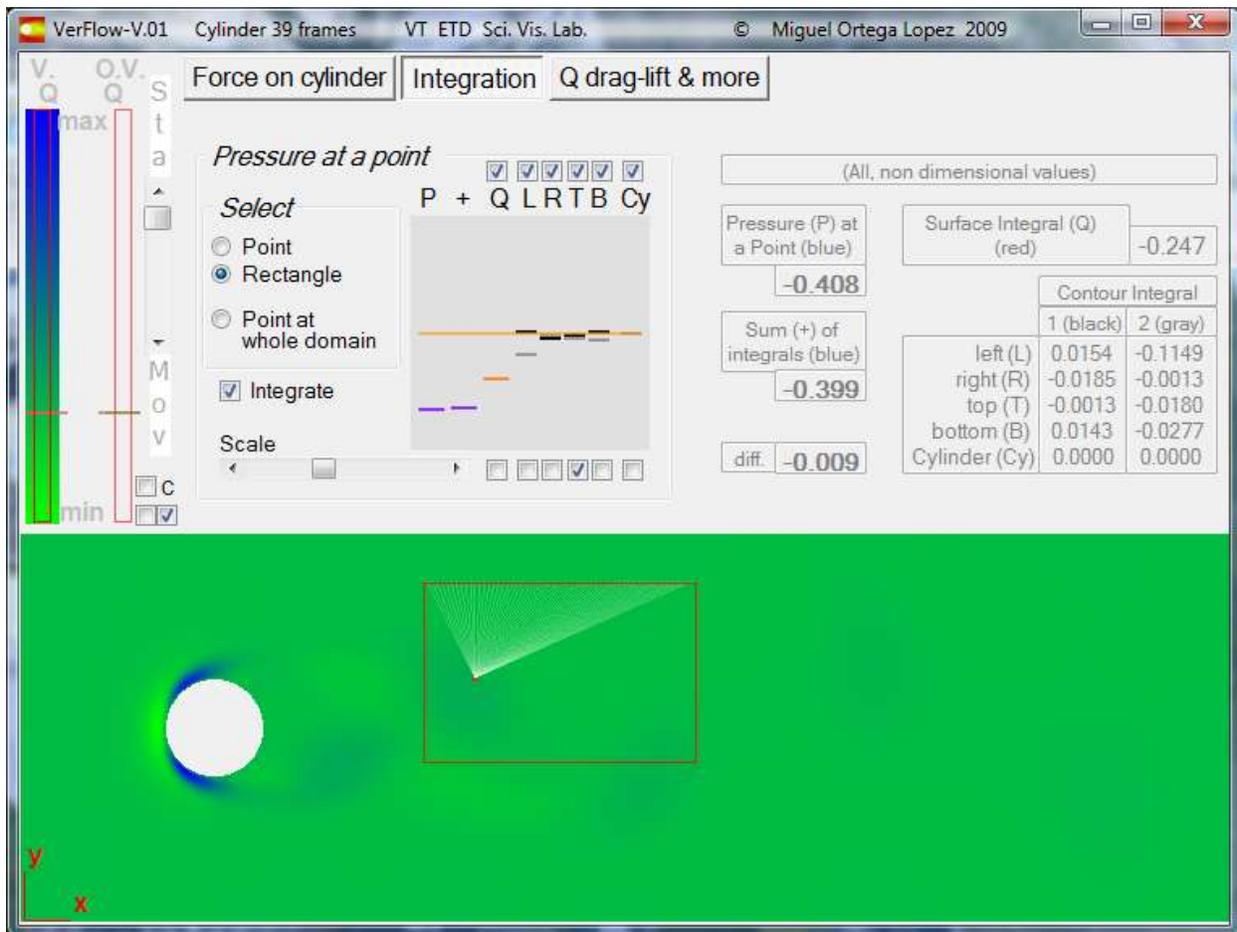


Figure C.16 Integration options for a rectangular subdomain

Figure C.16 shows how the integration for a rectangular domain, described in Section 4.2.2, is set in the *Integration* tab in zone 3. The user must select the *Integrate* check box in the *Pressure at a point* label in the *Integration* tab. The point location is defined by clicking (left mouse button) the point in the domain in zone 4 when the *Point* option button in the *Pressure at a point* label has been selected. The rectangle is defined by dragging (hold-down left mouse button) a region in the domain of zone 4 when the *Rectangle* option button in the *Pressure at a point* label has been selected.

Figure C.17 shows how the integration for the entire domain, described in Section 4.2.3, is realized in VerFlow-V.01. The *Integrate* check box and the *Point at entire domain* option button in the *Pressure at a point* label must be selected. The point location is directly selected by a left mouse button click directly over the represented flow in zone 4.

For both Figures C.16 and C.17, the individual integral components are represented *qualitatively* and quantitatively in the *Integration* tab in zone 3.

The *qualitative* representation is envisioned as a wide (left to right edge of image) orange horizontal line which is the zero reference pressure. The OpenFoam pressure at the specific selected point is envisioned as a shorter width blue line under the label “P”. The integration result is given at the right, again by a blue line, under the label “+”. The surface integral is represented by a red line under the label “Q”. Note that this integral is realized on the filtered visible region selected in zone 1. The contour integral components are represented by the gray and black lines at each boundary under the labels “L”, “R”, “T”, “B” and “Cy” corresponding to the left, right, top, bottom and cylinder boundaries. The Scale scroll bar, in the *Pressure at a point* label, allows the user to set a proper magnification of this representation. There are six check boxes at the bottom and six check boxes at the top of the graph in the *Integration* tab in zone 3. The six check boxes at the bottom are used to highlight the region of integration and also projected lines from the corresponding boundaries to the selected point in zone 4, this is only a visual effect. The six check boxes at the top of the graph filter the internal calculations so that the numerical results are filtered and the unselected integral components are not considered in the sum.

The quantitative dimensionless results are shown at the right in the *Integration* tab. The "*Pressure (P) at a Point (blue)*" is the result from OpenFoam for the pressure at the selected point. The "*Sum (+) of integrals (blue)*" is the result from the integration and sum of all selected contributions specified in the six check boxes at the top of the graph in the *Integration* tab. The difference between the two numerical results is marked as "*diff.*". All integral components are detailed in the "*Surface Integral (Q) (red)*" and in the "*Contour Integral*" table.

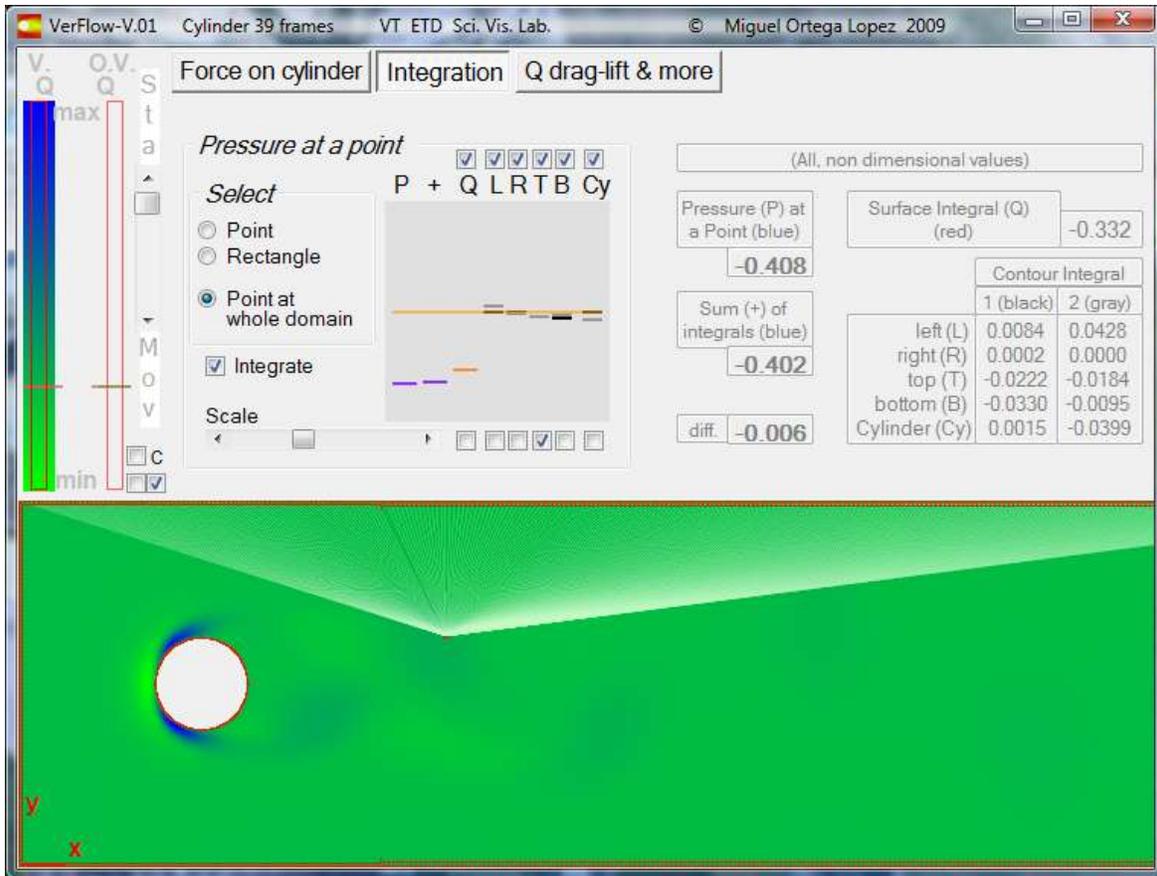


Figure C.17 Integration options for the entire domain

The available options for the *Q drag-lift & more* tab are given in Figure C.18.

The user can export the history "data" of horizontal (streamwise) velocity for a specific location, forces or pressure along the cylinder boundary as \*.txt files from the buttons *Export U*, *Export F* and *Export P* in the *Q drag-lift & more* tab in zone 3. These \*.txt files can be used to create other plots. The horizontal velocity is extracted from a location which can be changed from the code. The default location for the *Export U* button is the cell (50,20) in block 1, which is located in the

wake as shown in Figure 2.9. The forces for the *Export F* button are all nine components obtained in the *Force on cylinder* tab in zone 3. The pressure along the cylinder boundary (from OpenFoam result) is exported by VerFlow-V.01 in the counter clockwise order beginning in the cell (0,79) in block 0, which is located 45° counterclockwise from the horizontal radial line at the front of the cylinder. These buttons were used to extract information for the Fourier analysis in Sections 2.7.2 and 4.1.3, and for comparison with experimental data in Section 4.7.

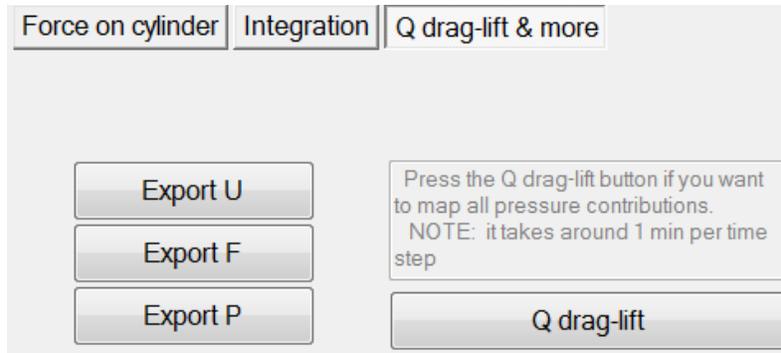


Figure C.18 *Q drag-lift & more* tab options

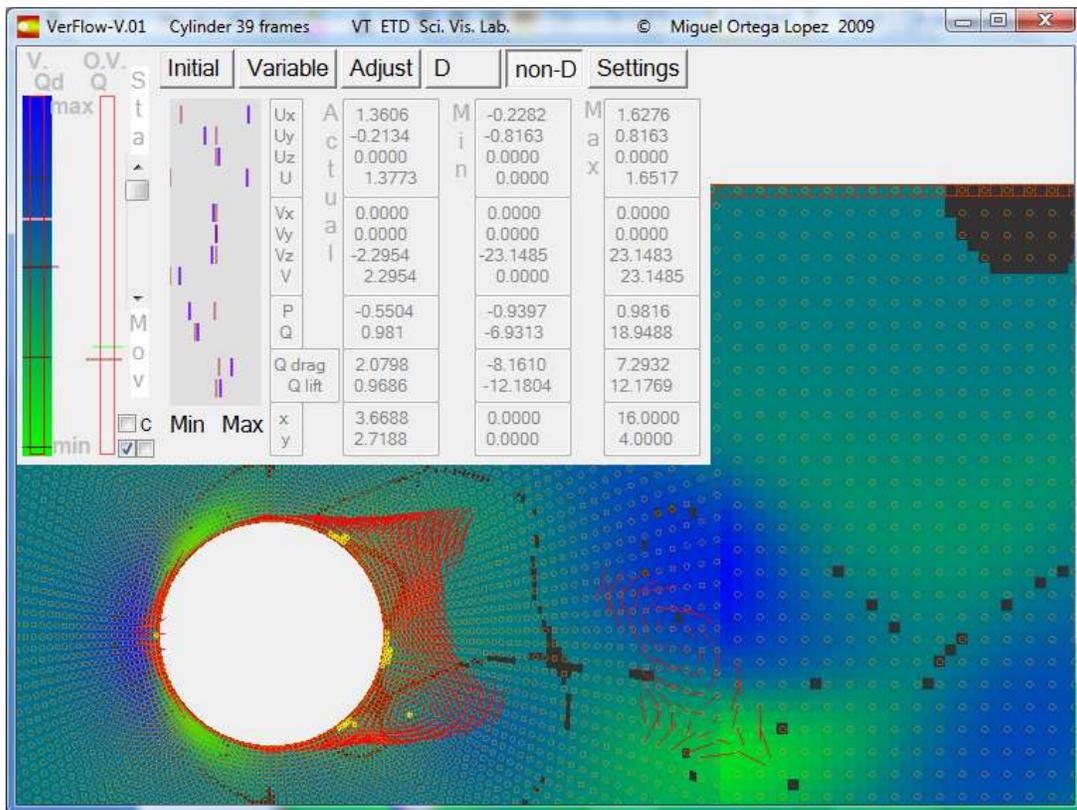


Figure C.19 Representation of *Q drag* after calculations realized on *Q drag-lift & more* tab

The *Q drag-lift* button in the *Q drag-lift & more* tab in zone 3 initiates the internal calculations described in Sections 4.4.1 and 4.4.2. A *Note* is included: *it takes around 1 min per time step* and the user must be aware that this operation should not be interrupted. So for a simulation containing 39 frames or time steps, this calculation will last about 40 minutes.

Once the calculation is finished, this variable information becomes available under the *Variable* tab in zone 2 and can be selected from there. Minimum and maximum limits will also appear in the *D* and *non-D* tabs in zone 2. All functions described in the previous tabs work with these two new variables: *Q drag* and *Q lift*. Figure C.19 shows an example of a customized representation of *Q drag* which includes the velocity vectors and stagnation points represented simultaneously. Note that the *non-D* tab includes the information of this variable.

## Appendix D. Solution to Poisson Equation in 2D

(Contributed by Clinton Dancey)

In 2D one of Green's identities is

$$\int_{\mathbb{S}} (u\nabla^2 v - v\nabla^2 u) dA = \int_{\mathcal{L}} \left( u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n} \right) dl \quad (\text{D.1})$$

where  $\mathbb{S}$  is the area enclosed by  $\mathcal{L}$ .  $u$  and  $v$  are assumed to be continuous through second derivatives.

Following Sokolnikoff & Redheffer (Sokolnikoff & Redheffer, 1966) consider the following limit:

$$\lim_{a \rightarrow 0} \int_{\mathcal{L}_1} \frac{f(Q)}{a^c} dl$$

where  $\mathcal{L}_1$  is a circle of radius  $a$ , centered at  $P$  and  $Q$  is a variable point on  $\mathcal{L}_1$ ,  $f$  is a continuous function and  $c$  is a constant.

$$\int_{\mathcal{L}_1} \frac{f(Q)}{a^c} dl = \int_{\mathcal{L}_1} \frac{f(P)}{a^c} dl + \int_{\mathcal{L}_1} \frac{f(Q) - f(P)}{a^c} dl = I_1 + I_2$$

$$I_1 = \int_{\mathcal{L}_1} \frac{f(P)}{a^c} dl = \frac{f(P)}{a^c} 2\pi a \quad \begin{cases} 2\pi f(P) & \text{for } c = 1 \\ 0 & \text{for } c < 1 \\ \infty & \text{for } c > 1 \end{cases}$$

$$|I_2| \leq \frac{2\pi a}{a^c} \max |f(Q) - f(P)|$$

If  $c = 1$   $|I_2| \rightarrow 0$  as  $a \rightarrow 0$  since  $f$  is continuous

If  $c < 1$   $|I_2| \rightarrow 0$  as  $a \rightarrow 0$

$$\therefore \int_{\mathcal{L}_1} \frac{f(Q)}{a^c} dl = \begin{cases} 2\pi f(P) & \text{for } c = 1 \\ 0 & \text{for } c < 1 \end{cases} \quad (\text{D.2})$$

Now in Equation (D.1) let  $v = w + \ln\left(\frac{1}{r}\right) = w + \ln(r^{-1})$

Let the region be inside  $\mathcal{L}$  and outside  $\mathcal{L}_1$ , with the enclosed region  $\mathbb{S}$  (outward normal  $n$  shown).

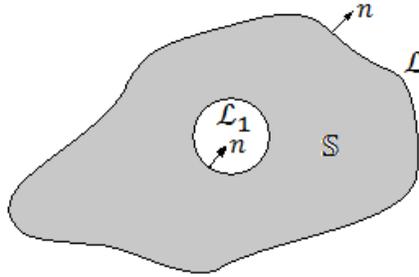


Figure D.1 Region definitions

So,

$$\begin{aligned}
 & \int_{\mathbb{S}} \{u \nabla^2 [w + \ln(r^{-1})] - [w + \ln(r^{-1})] \nabla^2 u\} dA \\
 &= \int_{\mathcal{L}} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl \\
 &+ \int_{\mathcal{L}_1} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl
 \end{aligned} \tag{D.1'}$$

We note:  $\nabla^2 v = \nabla^2 w + \nabla^2 [\ln(r^{-1})]$ , but it can be shown that  $\nabla^2 [\ln(r^{-1})] = 0$  in 2D.

So,  $\nabla^2 v = \nabla^2 w + 0$  in this case.

On  $\mathcal{L}_1$ :

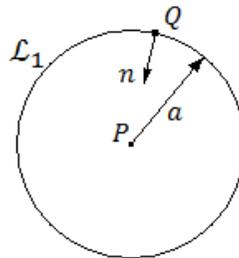


Figure D.2 Circular zone outside the region

$$\frac{\partial v}{\partial n} = \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n}$$

$$\frac{\partial}{\partial n} = -\frac{\partial}{\partial r}$$

$$\frac{\partial \ln(r^{-1})}{\partial n} = -\frac{\partial \ln(r^{-1})}{\partial r} = +\frac{\partial \ln(r)}{\partial r} = \frac{1}{r} = \frac{1}{a} \quad \text{on } \mathcal{L}_1$$

$$\therefore \left. \frac{\partial v}{\partial n} \right|_{\mathcal{L}_1} = \left. \frac{\partial w}{\partial n} \right|_{\mathcal{L}_1} + \frac{1}{a}$$

So we have on  $\mathcal{L}_1$ :

$$\int_{\mathcal{L}_1} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl = \int_{\mathcal{L}_1} u \left[ \frac{\partial w}{\partial n} + \frac{1}{a} \right] dl - \int_{\mathcal{L}_1} [w + \ln(r^{-1})] \frac{\partial u}{\partial n} dl$$

Assuming  $u$ , and  $w$ , and  $\frac{\partial w}{\partial n}$  are continuous:

$$\int_{\mathcal{L}_1} u \frac{\partial w}{\partial n} dl \rightarrow 0 \quad \text{as } a \rightarrow 0 \quad \text{and,}$$

$$\int_{\mathcal{L}_1} \frac{u}{a} dl \rightarrow 2\pi u(P) \quad \text{as } a \rightarrow 0 \quad \text{from Equation (D.2), so}$$

$$\lim_{a \rightarrow 0} \int_{\mathcal{L}_1} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl = 2\pi u(P) - \lim_{a \rightarrow 0} \int_{\mathcal{L}_1} [w + \ln(r^{-1})] \frac{\partial u}{\partial n} dl$$

where,

$$\lim_{a \rightarrow 0} \int_{\mathcal{L}_1} w \frac{\partial u}{\partial n} dl = 0$$

So,

$$\lim_{a \rightarrow 0} \int_{\mathcal{L}_1} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl = 2\pi u(P) - \lim_{a \rightarrow 0} \int_{\mathcal{L}_1} \ln(r^{-1}) \frac{\partial u}{\partial n} dl$$

Noting  $dl = a d\theta$ , the last term can be written:

$$\lim_{a \rightarrow 0} a \int_0^{2\pi} \ln(a^{-1}) \left. \frac{\partial u}{\partial n} \right|_{\mathcal{L}_1} d\theta = \lim_{a \rightarrow 0} \left[ a \ln a \int_0^{2\pi} \left. \frac{\partial u}{\partial n} \right|_{\mathcal{L}_1} d\theta \right]$$

But

$$a \ln a \rightarrow 0 \quad \text{as} \quad a \rightarrow 0$$

And,

$$\lim_{a \rightarrow 0} \int_{\mathcal{L}_1} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl = 2\pi u(P)$$

Then:

$$\int_{\mathcal{S}} \{ u \nabla^2 w - [w + \ln(r^{-1})] \nabla^2 u \} dA = \int_{\mathcal{L}} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl + 2\pi u(P)$$

Or,

$$2\pi u(P) = \int_{\mathcal{S}} \{ u \nabla^2 w - [w + \ln(r^{-1})] \nabla^2 u \} dA - \int_{\mathcal{L}} \left\{ u \left[ \frac{\partial w}{\partial n} + \frac{\partial \ln(r^{-1})}{\partial n} \right] - [w + \ln(r^{-1})] \frac{\partial u}{\partial n} \right\} dl$$

Now let  $w = 0$  to obtain:

$$2\pi u(P) = - \int_{\mathcal{S}} \ln(r^{-1}) \nabla^2 u dA - \int_{\mathcal{L}} \left\{ u \frac{\partial \ln(r^{-1})}{\partial n} - \ln(r^{-1}) \frac{\partial u}{\partial n} \right\} dl$$

Now let  $\nabla^2 u = -2\pi\Gamma$ , where  $\Gamma(x, y)$  is continuous, to obtain finally

$$2\pi u(P) = 2\pi \int_{\mathcal{S}} \Gamma \ln(r^{-1}) dA - \int_{\mathcal{L}} \left\{ u \frac{\partial \ln(r^{-1})}{\partial n} - \ln(r^{-1}) \frac{\partial u}{\partial n} \right\} dl$$

Which is the solution to Poisson's equation  $u(x, y)$ .

Applied to the equation for pressure, i.e.:

$$\nabla^2 p = -\rho u_{i,j} u_{j,i}$$

$$-2\pi\Gamma = -\rho u_{i,j} u_{j,i}$$

or,

$$\Gamma = \frac{\rho u_{i,j} u_{j,i}}{2\pi}$$

$$\therefore 2\pi p(P) = 2\pi \int_{\mathbb{S}} \frac{\rho u_{i,j} u_{j,i}}{2\pi} \ln(r^{-1}) dA - \int_{\mathcal{L}} \left\{ p \frac{\partial \ln(r^{-1})}{\partial n} - \ln(r^{-1}) \frac{\partial p}{\partial n} \right\} dl$$

$$\therefore \frac{p(P)}{\rho} = \frac{1}{2\pi} \int_{\mathbb{S}} u_{i,j} u_{j,i} \ln(r^{-1}) dA - \frac{1}{2\pi\rho} \int_{\mathcal{L}} \left\{ p \frac{\partial \ln(r^{-1})}{\partial n} - \ln(r^{-1}) \frac{\partial p}{\partial n} \right\} dl$$

where  $p(P)$  is evaluated at the point in question, all other quantities in  $\mathbb{S}$  and on  $\mathcal{L}$  are field points, with  $r = |P - Q|$ , the distance between  $P$  and the field point.