UNIVERSITÉ DE LORRAINE

École doctorale EMMA Lorraine

Extension de la méthode LS-STAG de type frontière immergée/cut-cell aux géométries 3D extrudées. Applications aux écoulements newtoniens et non newtoniens.

THÈSE

présentée et soutenue publiquement le 23 Mars 2018

pour l'obtention du

Doctorat de l'Université de Lorraine

(en Mécanique et Énergétique)

par

Farhad NIKFARJAM

Composition du jury

Rapporteur :	Kai SCHNEIDER	Professeur Aix-Marseille Université
Examinateurs :	Paola CINNELLA **(Présidente du jury)**	Professeur Arts et Métiers ParisTech
	Yoann CHENY	Maître de conférences Université de Lorraine
Directeur :	Olivier BOTELLA	Maître de conférences Université de Lorraine
Invités :	Salaheddine SKALI-LAMI	Maître de conférences Université de Lorraine
	Blaise NSOM	Professeur Université de Bretagne Occidentale
	Thibaut MENARD	Maître de conférences Université de Rouen

Laboratoire d'Energétique et de Mécanique Théorique et Appliquée — UMR 7536

Mis en page avec la classe thesul.

Acknowledgments

This thesis is the result of my studies at LEMTA labratory at université de Lorraine. I would like to begin by expressing my thanks to my advisors Salaheddine SKALI-LAMI and Olivier BOTELLA. My research has been also stimulated by discussions with Yoann CHENNY, and I am very grateful for his valuable advises. I gratefully acknowledge the other members of my thesis committee, Paola CINNELLA, Kai SCHNEIDER, Blaise NSOM and Thibaut MENARD.

I was fortunate to spend my time in LEMTA laboratory with nice and interesting people. In particular, my former office mates, Caroline HANOTIN, Sebastian CASTRILLON ESCOBAR, Miloud HADJ ACHOUR and Brice PORTELENELLE, who contributed to a comfortable working environment. I am also thankful to my good friends and colleagues, Waseem ALHADAD, Mondher BOUTERAA, Sofyane ABBOU, Thomas VARE, François KREMER, Rémy Christophe SELLIN, Saïd AIT HAMMOU TALEB, Zakariya BOUFAIDA, Lionel PERRIN, Thomas GAUMONT, Ahmad ADDOUM, Thomas LOUSSOUARN, Naima GAUDEL, Célien ZACHARIE, Jamal FANNIR, Morgan LAURENT-SANS and Omar RINGA, for all their support during my PhD studies. I would like to warmly thank all LEMTA staff, especially, Fatiha BOUMENDJEL, Irène LEONARD, Fabrice LEMOINE, Christian MOYNE, Ludovic BUHLER and Franck MANSUY. My sincerest thanks goes to the LEMTA electronics group (SEIM) members, Jamal OUHAJJOU, Simon BECKER, Mathieu WEBER, Hadrien CHAYNES, and also my friends in CRAN laboratory, Alain DELCONTE and Joël DAOUK. I thank you all for making this such an awesome place to work every day!

I truly appreciate my lovely Iranian friends, Ehsan JAMSHIDPOUR and Mohana AFSHARIAN, Amir BAHRANI and Fatemeh RAEISSI, Bamdad SALEMI, Zia ALBORZI and Mahta MAPAR, Meysam HASHEMI, Younes RAHMANI, Soheib MAGHSOODI and Mojdeh LAHOORI, Esmaeil MOUSSAVI, Mohsen BABAEIZADEH, Saeid AGHAEI, Alireza YEKTA and Mojdeh RASOULZADEH, Yaser HAFEZNIA, Mahta MOUSAVI, Abedin FERDOUSI and Masoumeh SADERI, and Shirin ENFERAD, I had the pleasure of spending time with these past five years.

My deepest gratitude and appreciation go to my family, Reza, Farangis and Farshad, for always encouraging and believing in me.

Farhad NIKFARJAM,

Nancy, April 2018

Je dédie cette thèse
à ma trés chère famille,
à mon Papa, Reza,
à ma Maman, Farangis,
et aussi mon Frère, Farshad.

Publications

My PhD publications

In the course of this thesis, the following articles has been published.

Journal Papers

2018 **F. Nikfarjam**, O. Botella, Y. Cheny, *The LS-STAG immersed boundary/cut-cell method for non-Newtonian flows in 3D extruded geometries*, **Computer Physics Communications (CPC)**, Volume 226, pages 67-802.

2016 O. Botella, Y. Cheny, **F. Nikfarjam**, M. Stoica, *Application of the LS-STAG immersed Boundary/cut-cell method to viscoelastic flow computations*, **Communications in Computational Physics (CiCP)**, volume 20, issue 4, pages 870-901.

Conference Papers

2016 O. Botella, Y. Cheny, **F. Nikfarjam**, J. Carlier, *The LS-STAG immersed boundary/cut-cell method for non-Newtonian flow and heat transfer in quasi-3D geometries*, **7$^{\text{ème}}$ édition du Colloque Francophone en Energie, Environnement, Economie et Thermodynamique - COFRET'16**, Bucharest, Romania.

2015 **F. Nikfarjam**, Y. Cheny, O. Botella, *The LS-STAG immersed boundary/cut-cell method for non-Newtonian flows in quasi-3D geometries*, **22$^{\text{ème}}$ Congrès Français de Mécanique**, Lyon, France.

2014 Y. Cheny, **F. Nikfarjam**, O. Botella, *Towards a fully 3D version of the LS-STAG immersed boundary/cut-cell method*, **8$^{\text{th}}$ International Conference on Computational Fluid Dynamics**, Chengdu, China.

2014 O. Botella, **F. Nikfarjam**, M. Stoica, Y. Cheny, *Entry flow computations of shear-thinning and viscoelastic liquids with the LS-STAG immersed boundary method*, **8$^{\text{th}}$ International Conference on Computational Fluid Dynamics**, Chengdu, China.

Contents

Introduction (Version Francaise)

Introduction (English Version)

Chapter 1
Physical modeling

1.1 Governing equations of fluid dynamics 13
1.2 Rheological behavior of fluids 14
1.2.1 Time-independent fluid behavior 15
1.3 Approximation of the shear-thinning behavior 17
1.3.1 Power-law (Ostwald-de Waele) model 18
1.3.2 Cross model 18
1.3.3 Hanotin model 19
1.4 Shear-thinning viscosity models employed in our numerical simulations 19
1.5 Conclusion 22

Chapter 2
Background of immersed boundary/cut-cell methods and 2D LS-STAG

2.1 Review of immersed boundary/cut-cell methods 23
2.1.1 Ghost-cell finite-difference approach 26
2.1.2 Cut-cell finite-volume approach 27
2.2 Background of LS-STAG method for 2D configurations 30
2.2.1 Principles and description of LS-STAG mesh 30
2.2.2 Global conservation laws for viscous incompressible flows 35
2.3 Advantage of LS-STAG over unstructured grid-based methods in terms of memory efficiency 40
2.4 Conclusion 42

Chapter 3
The LS-STAG discretization for 3D extruded geometries

3.1 Introduction . . . 43
3.2 Preliminaries and principles of LS-STAG method in 3D extruded meshes . . . 48
3.2.1 Governing equations for non-Newtonian flows . . . 48
3.2.2 LS-STAG mesh in 3D extruded geometries . . . 48
3.3 Discretization of the continuity equation . . . 52
3.4 Discretization of the momentum equation . . . 55
3.4.1 Discretization of the rate of change of the momentum . . . 58
3.4.2 Discretization of the pressure gradient . . . 58
3.4.3 Skew-symmetric discretization of the convective fluxes . . . 58
3.4.4 Discretization of velocity gradients on LS-STAG mesh . . . 62
3.4.5 Discretization of viscous fluxes . . . 71
3.5 Extension of LS-STAG method to non-Newtonian fluids . . . 76
3.5.1 Interpolation of the shear-rates at the centroids . . . 76
3.5.2 Interpolation of the shear-rates on edges . . . 78
3.6 LS-STAG time stepping method and solution of linear systems . . . 78
3.7 Conclusion . . . 80

Chapter 4
Numerical Validation of LS-STAG method

4.1 Parallelization of the LS-STAG solver . . . 83
4.1.1 Domain decomposition for Cartesian grids . . . 83
4.1.2 Scalability of LS-STAG solver . . . 87
4.2 Accuracy study of the LS-STAG method . . . 90
4.3 Computation of flows around a circular cylinder . . . 95
4.3.1 Benchmark computations of laminar flow past a circular cylinder in a confined square duct . . . 96
4.3.2 Weakly turbulent flow at $Re_D = 300$ . . . 108

Chapter 5
Validation of LS-STAG results with experimental data for non-Newtonian fluid flows

5.1 Flow of shear-thinning liquids in a 1:2 axisymmetric sudden expansion . . . 115
5.1.1 Grid convergence study in expansion flow . . . 119
5.1.2 Comparison of global parameters in expansion flow . . . 122

5.1.3 Description of velocity profiles in different sections upstream and downstream of the abrupt expansion 127
5.2 Numerical simulation of granular flows in a Taylor-Couette cell 133
5.2.1 Study of convergence criterion in granular flows 133
5.2.2 Comparison of LS-STAG results with experimental data of granular flows in Taylor-Couette cell 143

Conclusions and Perspective

Appendix A
Discretization of the convective fluxes in basic types of cut-cells

A.1 Discretization of convective fluxes in a triangular cut-cell 155
A.2 Discretization of convective fluxes in a trapezoidal cut-cell 157
A.3 Discretization of convective fluxes in a pentagonal cut-cell 157

Bibliography **159**

List of Figures

1 Maillage de type *body-fitted* (à gauche) et maillage cartésien pour méthode de frontières immergées (à droite) pour un domaine de calcul bidimensionnel Ω_f. . . 2

1 Illustration of an orthogonal structured body-fitted grid (left) and the schematic view of the immersed boundary method in a two-dimensional computational domain Ω_f (right). The cut-cells are highlighted in yellow. 8

1.1 Shear stress $\boldsymbol{\tau}$ as a function of shear-rate $\dot{\gamma}$. 16

1.2 Demonstration of zero-shear η_0 and infinite-shear η_∞ viscosities in the shear-thinning regime. 17

1.3 Flow curves of the 4 Xanthan solutions reprinted from [86]: viscosity as a function of the shear-rate. Comparison between rheometric measurements (symbols) and the Cross model (solid line). 20

1.4 Viscosity η as a function of the shear stress σ for different values of vibration stress σ_v for the glass beads of $100\,\mu m$ immersed in a Newtonian fluid with the viscosity of $\eta_f = 67.8\ mPa.s$ reprinted from Hanotin [46]. The rectangular zone display the estimation of frictional stress σ_f. 21

2.1 Sketch of velocity interpolation in the vicinity of immersed boundary. 25

2.2 Schematics of the points in the vicinity of an immersed boundary used in the ghost-cell approach. F_i represent the fluid points, G denotes the ghost point and B_i and P_i are the locations where the boundary condition can be enforced. . . . 26

2.3 Schematic of re-shaped cell next to immersed boundary with associated fluxes (left); 6-point interpolation stencil for determining the flux over the southwest cell face (right). Figures are based on [106]. 27

2.4 (a) A master and slave cell are shown for the u component of velocity. The slave cell velocity has only one pressure node associated with it. It is moved to the same position as the master cell node. The figures are reprinted from [58]. (b) Staggered grid with an oblique boundary in two dimensions. Cells are truncated by the boundary. Circles indicate pressure nodes. Horizontal arrows indicate the location of nodes for the u velocity component and vertical arrows the v velocity component. Examples of a u, v, and P cell at the boundary are highlighted. . . 29

2.5 Complex geometry Ω^{ib} embedded in a rectangular domain Ω. 31

2.6 Staggered arrangement of the variables near the trapezoidal cut-cell $\Omega_{i,j}$ on the LS-STAG mesh reprinted from [22]. 32

2.7 Location of stresses in fluid cells (Cartesian and cut-cells) reprinted from [17]. . . 33

2.8 Basic types of half $u_{i,j}$ control volumes, $\Omega^u_{i,j}$, inside the cut-cell $\Omega_{i,j}$ reprinted from [22]. The green diamonds (♦) denote the locations of the discretization of the velocity boundary conditions. Note that the re-entrant corner of a Cartesian mesh is a particular case of pentagonal cell (c) when $\phi_{i,j} = 0$, and the particular case of a salient corner (f) in which $\phi_{i-1,j-1} = \phi_{i-1,j} = \phi_{i,j} = 0$ and $\phi_{i,j-1} < 0$) only differs in definition of the cell volume with the limiting case of triangle (e) where $\phi_{i-1,j-1} = \phi_{i,j} = 0$, $\phi_{i-1,j} > 0$ and $\phi_{i,j-1} < 0$. 34

2.9 Example of unstructured mixed (triangular, quadrilateral) mesh in 2D. 41

3.1 16 types of cells existed in full 3D geometries. 44

3.2 Examples of flows in 3D geometries extruded in the z direction.. 45

3.3 3 basic types of cut-cells in 3D extruded geometries. 46

3.4 14 distinct cells (12 cut-cells, a Cartesian cell and a solid cell) present in 3D extruded geometries. Each cell is identified with its flag in red. 47

3.5 The elementary faces of a Cartesian cell denoted with the usual compass notations (w: West, e: East, s: South, n: North, b: Bottom and t: Top). 49

3.6 Cartesian cell and 3 basic types of cut-cells in 3D extruded configurations. The unknown velocities are discretized at faces centers, pressure and normal velocity gradients (*i.e.* $\partial u/\partial x$, $\partial v/\partial y$ and $\partial w/\partial z$) at cell centroid (●), tangential velocity gradients at edges centers: $\partial u/\partial y$, $\partial v/\partial x$ (■) and $\partial w/\partial x$, $\partial u/\partial z$ (■) and $\partial w/\partial y$, $\partial v/\partial z$ (■). 50

3.7 Geometric parameters of pentagonal cut-cell $\Omega_{i,j,k}$ in which the immersed boundary $\Gamma^{\text{ib}}_{i,j,k}$ is outlined with red dashed line (--). The extremities of $\Gamma^{\text{ib}}_{i,j,k}$, *i.e.* $(x_i, y^{\text{ib}}_{i,j,k}, z_k)$, $(x^{\text{ib}}_{i,j,k}, y_j, z_k)$, $(x^{\text{ib}}_{i,j,k}, y_j, z_{k-1})$ and $(x_i, y^{\text{ib}}_{i,j,k}, z_{k-1})$ are denoted with ①, ②, ③ and ④, respectively. The level-set function values ϕ (■) which stored at the vertices of cell, are used to compute the cell-edge fraction ratios of edges $\mathcal{E}^x$, $\mathcal{E}^y$ and $\mathcal{E}^z$ in x, y and z directions, respectively. The edge $\mathcal{E}^y_{i,j,k}$ endpoints are $(x_i, y^{\text{ib}}, z_k)$ and (x_i, y_{j-1}, z_k). For the solid edge of $\mathcal{E}^z_{i,j,k}$, the cell-edge fraction ratio is $\alpha^z_{i,j,k} = 0$. The eastern, northern and top elementary faces, namely $\Gamma^{\text{e}}_{i,j,k}$, $\Gamma^{\text{n}}_{i,j,k}$ and $\Gamma^{\text{t}}_{i,j,k}$, are highlighted in red, magenta and green, respectively. 51

3.8 Control volume $\Omega_{i,j,k}$ for a pentagonal cut-cell. 53

3.9 Representation of half control volumes for u in a Cartesian cell: (a) the isometric view, (b) top face at plane $z = z_k$, (c) western face at plane $x = x_i - \Delta x_i/2$ and (d) northern face at plane $y = y_j$. The blue and red halves are related to the control volumes $\Omega^u_{i-1,j,k}$ and $\Omega^u_{i,j,k}$, respectively. 56

3.10 Staggered arrangement of variables in control volume $\Omega^u_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i+1,j,k}$. This control volume (--) is constructed from the halves of two adjacent Cartesian cells far away the immersed boundary. 57

3.11 Staggered arrangement of the variables for two adjacent cut-cells in the LS-STAG mesh. The control volume $\Omega^u_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i+1,j,k}$ for $u_{i,j,k}$ is indicated in red dashed line (--). 57

3.12 Combination of the halves of 2 adjacent trapezoidal cut-cells which forms the control volume $\Omega^u_{i,j,k}$ (Red dashed line (--)). 61

3.13 The rectangular element of Saramito [88]. 64

3.14 Representation of the half control volumes (--) for $w_{i,j,k}$. The line (··) connecting two consecutive velocity vectors $w_{i,j,k}$ and $w_{i,j,k-1}$ is orthogonal to $\Gamma^{w,\text{b}}_{i,j,k}$ due to the extrusion in z direction. 65

3.15 Representation of the control volumes $\Omega^u_{i,j,k}$ and $\Omega^u_{i,j-1,k}$ in red dashed line (--). $\Omega^u_{i,j-1,k}$ is formed from 2 Cartesian cells and $\Omega^u_{i,j,k}$ is in the vicinity of immersed boundary. The velocity gradients $\partial u/\partial y|_{i,j,k}$, $\partial v/\partial x|_{i,j,k}$ and $\partial u/\partial y|_{i,j-1,k}$ ($\partial v/\partial x|_{i,j-1,k}$) are denoted with ①, ② and ③, respectively. 67

3.16 Two consecutive solid faces form the eastern face of v control volume, $\Omega^v_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i,j+1,k}$ (--). In this particular case, the velocity gradient $\partial u/\partial y|_{i,j,k}$ (■) is computed by employing the Schwarz theorem. The northern face $\Gamma^{\text{n}}_{i,j,k}$ of control volume $\Omega_{i,j,k}$ is shaded in red. 68

3.17 Velocity gradient $\partial u/\partial z|_{i,j,k}$ (■) is located at the center of edge $\mathcal{E}^y_{i,j,k}$. The face $\Gamma^{u,\text{t}}$ in the control volume $\Omega^u_{i,j,k,}$ is shaded in red. 69

3.18 Illustration of an arbitrary staggered control volume $\Omega^w_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i,j,k+1}$ (red dashed line (--)) whose eastern face $\Gamma^{w,\mathrm{e}}_{i,j,k}$ is shaded in red and its solid face $\Gamma^{w,\mathrm{ib}}_{i,j,k}$ is highlighted in green. Velocity gradient $\partial w/\partial x|_{i,j,k}$ (■) is located at the center of edge $\mathcal{E}^y_{i,j,k}$. The adjacent velocity vectors, $w_{i,j,k}$ and $w_{i+1,j,k}$, are connected by a dotted line (··) which is not perpendicular to the face $\Gamma^{w,\mathrm{e}}_{i,j,k}$. 70

3.19 The normal distance d_{ib} between the w velocity location (open cell face barycenter) and the immersed boundary in the case of the wall shear stress $\partial w/\partial \boldsymbol{n}|_{i,j,k}$ (■) for 3 types of cut-cells. 71

3.20 Illustration of the control volume $\Omega^u_{i,j,k}$ in the vicinity of two trapezoidal cells (red dashed line (--)). The western and $\Gamma^{u,\mathrm{w}}_{i,j,k}$ ($\Delta y^{u,\mathrm{w}}_{i,j,k} \times \Delta z_k$) and eastern $\Gamma^{u,\mathrm{e}}_{i,j,k}$ ($\Delta y^{u,\mathrm{e}}_{i,j,k} \times \Delta z_k$) faces are shaded in red. The shear stress $\dot{\gamma}_{xy}|_{i,j,k}$ is located at the midpoint of solid edge (■). 72

3.21 Staggered arrangement of the shear-rates on the LS-STAG mesh: the diagonal terms ($\dot{\gamma}_{xx}$, $\dot{\gamma}_{yy}$ and $\dot{\gamma}_{zz}$) at the cell centroids (●) and the off-diagonal ones at midpoints of the edges $\dot{\gamma}_{xy}$ (■), $\dot{\gamma}_{xz}$, (■) and $\dot{\gamma}_{yz}$ (■). 77

4.1 A 2D Cartesian decomposition for a non-periodic domain with the array indexing of sub-domains. The sub-domains are handled by the processors with distinct ranks starting from 0. 85

4.2 The communication between two successive processors. The velocity components in cells $\Omega_{i,j,k}$ and $\Omega_{i+1,j,k}$ related to two different processors (1 and 2), are respectively stored in overlapping cells $\Omega'_{i,j,k}$ and $\Omega'_{i+1,j,k}$ via MPI "point-to-point" communication routines. The data in blue cells are computed on current processor while those in red (overlapping) cells are retrieved from the neighbor processors. 86

4.3 A cubic domain with dimensions $2\,m \times 2\,m \times 2\,m$ in which the Taylor-Green vortex flow is simulated. The iso-surfaces of the z component of vorticity, $\omega_z = \pm 2$, are also illustrated at $t = 0$. 88

4.4 The efficiency of LS-STAG solver versus the number of processors (np) in the weak scaling measurement with the local grid $64 \times 64 \times 64$ cells per each processor. The dashed line (- -) shows the ideal behavior. 89

4.5 The speedup of LS-STAG solver versus the number of processors (np) in the strong scaling measurement. The dashed line (- -) shows the ideal speedup. . . . 90

4.6 Geometry of Hagen-Poiseuille flow. 91

4.7 Comparison of the 2D cross section of the coarsest mesh ($h = 1.65 \times 10^{-2}$) for (a) LS-STAG and (b) staircase methods for Hagen-Poiseuille flow. 92

4.8 Axial velocity profile for Newtonian and shear-thinning Hagen-Poiseuille flows at cross section $z = L_z/2$ for $h = 5.5 \times 10^{-3}$. 92

4.9 Maximal error on streamwise velocity versus grid size h for the Hagen-Poiseuille flow (a) on the whole fluid domain and (b) on 90% of the fluid domain. 94

4.10 Pointwise error profiles on cross section $z = L_z/2$ for a mesh with $h = 5.5 \times 10^{-3}$ in Hagen-Poiseuille flow. 95
4.11 Computational domain for the cylinder flows of Section 4.3. 96
4.12 2D view of the coarsest meshes for the circular cylinder benchmark: (a) L2 mesh of FLUENT© against and (b) M2 mesh LS-STAG method. 98
4.13 Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, in the center plane $z = 0.205\,m$ at $Re_D = 20$. 99
4.14 u and v velocity profiles for grid levels M2–M4 in y direction for 3 different locations $x = 0.35\,m$, 0.5 m and 0.65 m along the square duct. 101
4.15 Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, at the center plane $z = 0.205\,m$ for $t = 0$, 1, $2\,s$. 103
4.16 Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, at the center plane $z = 0.205\,m$ for $t = 3$, 4, $5\,s$. 104
4.17 Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, at the center plane $z = 0.205\,m$ for $t = 6$, 7, $8\,s$. 105
4.18 Time evolution of the force coefficients computed by LS-STAG, FLUENT© and the reference solution [9] at the cylinder. At left: on the total time interval; at right: close up around $t = 4\,s$. 106
4.19 Time evolution of the force coefficients for the cylinder flow at $Re_D = 300$ on the M1 mesh. 109
4.20 Vortex visualization of the cylinder flow at $Re_D = 300$ by the Q-criterion on mesh M2. The isosurfaces of $Q = 0.1$ (green) are displayed at an instant in time. The isosurfaces are colored with streamwise vorticity values: red is $\omega_x\, D/U_{\text{ref}} = 0.25$ and blue is $\omega_x\, D/U_{\text{ref}} = -0.25$. 110
4.21 Mean streamwise velocity profiles at different locations in the wake of circular cylinder at $Re_D = 300$. 112
4.22 Mean cross-flow velocity profiles at different locations in the wake of circular cylinder at $Re_D = 300$. 113
4.23 time-averaged streamwise velocity fluctuations at different locations in the wake of circular cylinder at $Re_D = 300$. 113
4.24 time-averaged cross-flow velocity fluctuations at different locations in the wake of circular cylinder at $Re_D = 300$. 114
4.25 time-averaged Reynolds shear stress at different locations in the wake of circular cylinder at $Re_D = 300$. 114

5.1 Sketch of a circular duct with sudden expansion: (a) isometric view and (b) 2D view in the vicinity of singularity. The sudden expansion plane is placed at $z = 0$. 116
5.2 25 distinct cells introduced in axisymmetric expansion flow. 118
5.3 25 distinct cells introduced in axisymmetric expansion flow (continued). 119

5.4 Grid convergence of axial velocity profiles at different cross-sections of the duct for Xanthan 0.1% flow at $Re_{D_1} = 56$. The circle (∘) denotes the experimental results of Rigal [86] while the dash-dotted (-·-), dotted (··), dashed (--) and solid (–) lines represent LS-STAG solutions on meshes M1 to M4 respectively. 121
5.5 Variation of recirculation zone length against generalized Reynolds number. . . . 123
5.6 Evolution of vortex intensity with the Reynolds number: LS-STAG (-□-) versus FLUENT© (-·∗-·) [86] for three different fluids. 125
5.7 Evolution of establishment length with the Reynolds number: LS-STAG (-□-) versus FLUENT© (-·∗-·) [86] for three different fluids. 126
5.8 Variation of Couette correction factor against Reynolds number for three different fluids. 126
5.9 Axial velocity profiles of water at $z = -9\ mm$ for different Reynolds numbers: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation. For all Reynolds number the ratio of w/U_{ref} is equal to 1.99. 127
5.10 Axial velocity profiles of Xanthan 0.1% at $z = -9\ mm$ for different Reynolds numbers: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation. 128
5.11 Axial velocity profiles of Xanthan 0.3% at $z = -9\ mm$ for different Reynolds numbers: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation. 128
5.12 Axial velocity profiles at different cross-sections after the singularity for water at $Re_{D_1} = 9$. 129
5.13 Axial velocity profiles at different cross-sections for Xanthan 0.1% flow at $Re_{D_1} = 9$: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation. 130
5.14 Axial velocity profiles at different cross-sections for Xanthan 0.3% flow at $Re_{D_1} = 9$: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation. 130
5.15 Axial velocity profiles in different cross-sections after the singularity for water at $Re_{D_1} = 100$. 131
5.16 Axial velocity profiles in different cross-sections after the singularity for Xanthan 0.1% flow at $Re_{D_1} = 100$: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation. 131
5.17 Axial velocity profiles in different cross-sections after the singularity for Xanthan 0.3% flow at $Re_{D_1} = 100$. 132
5.18 The axial velocity profiles at the centerline along the pipe at $Re_{D_1} = 9, 56$ and 100 for water, Xanthan 0.1% and 0.3%. The experimental data of Rigal [86] is also given for the shear-thinning fluids. 132
5.19 Schematics of a Taylor-Couette cell in which D_{i} and D_{e} are respectively the diameter of inner and outer cylinders and L_{z} is the length of cylinders. 134
5.20 Evolution of residuals on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations. 135

5.21 Evolution of the torque values exerted on the inner cylinder versus the time iterations in Run 1. 136
5.22 Evolution of the torque errors exerted on the inner cylinder versus the time iterations in Run 1. 136
5.23 A close-up view of azimuthal velocity profiles around the inner cylinder at different residuals ϵ_p for Run 1. 138
5.24 A close-up view of shear-rate profiles around the inner cylinder at different residuals ϵ_p for Run 1. 138
5.25 Evolution of residuals on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations. 140
5.26 Evolution of residues on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations. 140
5.27 Evolution of the torque values exerted on the inner cylinder versus the time iterations in Run 2 for the rescaled geometry. 141
5.28 Evolution of the torque errors exerted on the inner cylinder versus the time iterations for the rescaled geometry in Run 2. 141
5.29 A close-up view of azimuthal velocity profiles around the inner cylinder at different residuals ϵ_p for the rescaled geometry in Run 2. 142
5.30 A close-up view of shear-rate profiles around the inner cylinder at different residuals ϵ_p for the rescaled geometry in Run 2. 142
5.31 A close-up view of the azimuthal velocity profiles along the annular gap for different meshes in Case 1. 144
5.32 A close-up view of the shear-rate profiles along the annular gap for different meshes in Case 1. 145
5.33 A close-up view of the viscosity profiles along the annular gap for different meshes in Case 1. 145
5.34 The azimuthal velocity profile along the gap for Case 1 along with the experimental data reported by [38] and the velocity profile for corresponding Newtonian fluid. 147
5.35 The azimuthal velocity profile along the gap for Case 2 along with the experimental data reported by [38] and the velocity profile for corresponding Newtonian fluid. 147
5.36 The azimuthal velocity profile along the gap in Case 3 along with the experimental data reported by [38] and the velocity profile for corresponding Newtonian fluid. 148
5.37 The evolution of local shear-rate along the annular gap for different cases. 149
5.38 The evolution of local viscosity along the annular gap for different cases. 149
5.39 The shear stress profile along the annular gap ($r \in [7.5\,17.5]\,mm$) for different cases. 150

1 The xy view of two adjacent cut-cells. The velocities involved in diamond cell scheme are $w_{i,j,k}$, $w_{i+1,j,k}$, w_{n} and w_{s}. The vector joining two consecutive velocities $w_{i,j,k}$ and $w_{i+1,j,k}$ is indicated by $\boldsymbol{d}$. 153

A.1 Representation of the half control volumes for $u_{i,j,k}$ inside the generic types of cut-cells by red dashed line (--). 156

List of Tables

1.1 Parameters of the power-law and Cross models for 4 Xanthan solutions reported by Rigal [86]. 20

2.1 Analytical formulas of the volume for the basic types of cut-cells. 35

3.1 The cell-face fraction ratios for the cut-cells shown in Figure 3.6. 52
3.2 The interpolation weights $\beta_{i,j,k}$ for the generic types of cells in 3D extruded geometries. 78

4.1 Salient properties of the LS-STAG meshes used for the circular cylinder benchmark. Note that the time-steps are used for both LS-STAG and FLUENT© computations in unsteady cases. 98
4.2 Comparison of global parameters obtained by LS-STAG and staircase methods with the reference values reported in [19, 54] ($C_{\mathrm{D}} = 6.1853$, $C_{\mathrm{L}} = 0.009401$ and $\Delta P = 0.171$) on meshes M2–M4 in cylinder flow at $Re_D = 20$. The FeatFlow computations on meshes L2–L4 reported in [10] and the lower and upper bounds of results in [90] are also given. 102
4.3 Relative errors (in percent) of the force coefficients with respect to the reference solution [9] ($C_{D_{\max}} = 3.2978$, $C_{L_{\max}} = 0.00276$ and $C_{L_{\min}} = -0.0110$) along with L_2 and L_∞ norms of errors within time interval $t \in [0, 8]\, s$. 107
4.4 Clock time and memory usage of the LS-STAG and FLUENT© computations on the IJL-LEMTA parallel machine. Note that *np* refers to the number of processors. 108
4.5 Mean flow parameters for the cylinder flow at $Re_D = 300$. 111

5.1 Main properties of the grids for axisymmetric expansion flow. 119
5.2 Grid convergence study of the dimensionless vortex length ($L_{\mathrm{v}}/\mathrm{d}$), vortex intensity $\Delta\psi^* = \Delta\psi/(\pi \mathrm{d}^2\, U_{\mathrm{ref}})$ and establishment length ($L_{\mathrm{e}}/\mathrm{d}$) and Couette correction factor (C) at $Re_{D_1} = 56$. 122
5.3 Comparison of LS-STAG results and reported data of [86, 71] for water and Xanthan solutions with 0.1% and 0.3% mass concentration. 124
5.4 The main properties of 2 different cases studied for the flows of dry granular particles in a Taylor-Couette cell. 134

5.5 The values of torque exerted on the inner cylinder in Run 1 for different values of ϵ_p. 137
5.6 The main properties of 3 different cases studied for the flows of granular suspensions in a Taylor-Couette cell. ω_{i} represents the angular velocity inner cylinder. . 143
5.7 Main properties of the grids used for Case 1. 144
5.8 The torque values exerted on the inner cylinder for different meshes in Case 1. . 146
5.9 The torque values exerted on the inner cylinder for different cases. 146

Introduction (Version Francaise)

Les applications avancées en mécanique des fluides numérique (MFN) concernent des écoulements en géométries tridimensionnelles très complexes, dont les phases de création CAO et de maillage sont très gourmandes en homme-heures et nécessitent des utilisateurs très expérimentés. Ces maillages *body-fitted* ("ajustés au corps") sont de type non-structuré (cellules hexaédriques, tétraédriques, *etc.* $\cdots$), et utilisent donc un adressage indirect qui requiert de grandes ressources informatiques (taille mémoire, temps de calcul). En outre, il est très difficile d'assurer la qualité de ces maillages (cellules non distordues, denses en zone de proche paroi, *etc.* ...), ce qui impacte négativement la convergence et la précision de la solution numérique. Les méthodes de frontières immergées sont apparues ces dernières décennies comme une alternative aux méthodes *body-fitted*. Elle sont basées sur des discrétisations cartésiennes extrêmement robustes et optimisées, et ne nécessite donc qu'une fraction des ressources informatiques d'une méthode basée sur une grille non-structurée. Les méthodes de frontières immergées permettent d'effectuer des simulations en géométries complexes sur de simples grilles cartésiennes, voir par exemple la Figure 1, ce qui facilite considérablement les générations de maillage complexes et élimine le besoin de remailler le domaine pour les géométries mobiles, ce qui est une étape nécessaire pour les méthodes *body-fitted*.

Les méthodes des frontières immergées se référaient à l'origine à une approche développée par Peskin [78] en 1972 pour simuler les interactions fluide-structure pour des domaines à frontière élastique, où les effets des frontières immergées sont reproduits en imposant une force extérieure appropriée dans les équations. Par la suite, on a distingué deux classes majeures de méthodes de frontières immergées selon la manière dont les conditions aux limites sont imposées sur la frontière immergée [67]: l'approche de forçage continu dans laquelle un terme force est incorporé dans les équations continues avant la discrétisation, et l'approche de forçage discret où le forçage est introduit après discrétisation des équations. L'un des principaux problèmes de la première approche est l'impossibilité de représenter les interfaces de façon "nette" (*sharp*), ce qui entraîne une dégradation de la précision spatiale de la solution [79]. En outre, l'application du terme de forçage au niveau continu pose des problèmes pour des domaines à frontière rigide, car les lois constitutives pour parois élastiques ne sont pas nécessairement adaptées. Ce problème est en partie résolu par l'approche de forçage discret. Cependant, la plupart des méthodes de frontières immergées remplacent par de simples interpolations la discrétisation des équations dans les

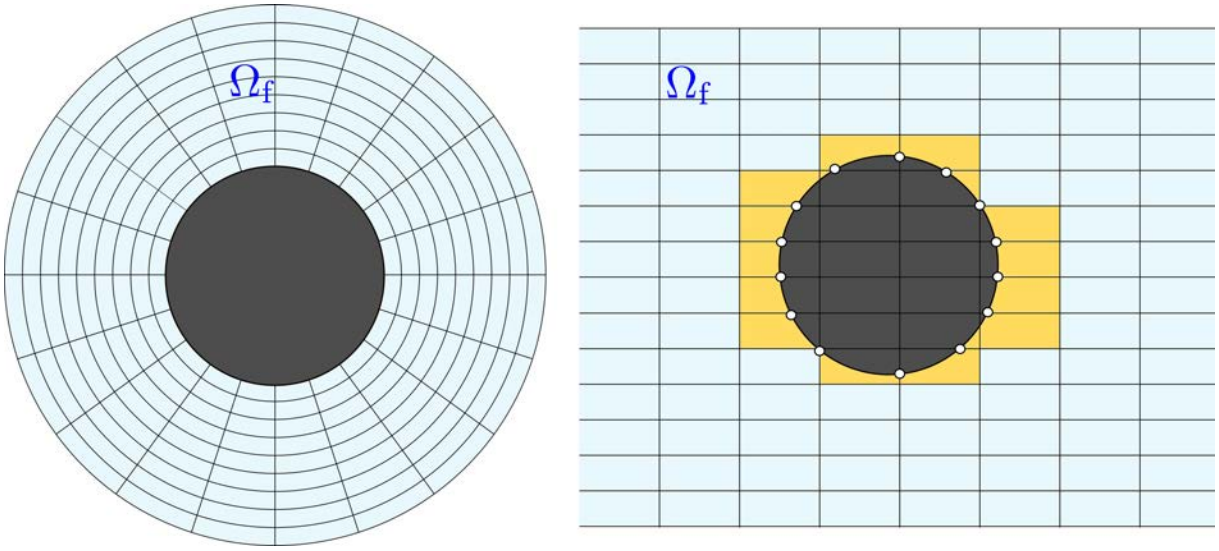


Figure 1: Maillage de type *body-fitted* (à gauche) et maillage cartésien pour méthode de frontières immergées (à droite) pour un domaine de calcul bidimensionnel Ω_f.

cut-cells (cellules de forme irrégulière qui sont formées par l'intersection du maillage cartésien avec la frontière immergée, *cf*. Figure 1 (à droite)). Ainsi, la conservation de la masse n'est pas observée au sens discret, ce qui entraine la présence de vitesses à divergence non nulle ou des oscillations de pression au voisinage de la frontière immergée [56]. Une représentation "nette" des frontières immergées est obtenue en utilisant l'approche dite *cut-cell*, initialement proposée pour les écoulements non visqueux dans [23]. Dans cette approche, les équations du mouvement sont discrétisées dans les *cut-cells* par des techniques *ad hoc*, par exemple en fusionnant les cellules, ce qui permet d'assurer la conservativité de la discrétisation. Néanmoins, l'inconvénient majeur de cette approche est la perte de la structure à 5 points (en 2D) des méthodes cartésiennes. De plus, l'extension tridimensionnelle des techniques basées sur la fusion des cellules est loin d'être triviale [67].

Une discrétisation innovante dans les *cut-cells* appelée "LS-STAG" (Level-Set, STAGgered) a été développée par Cheny et Botella [20, 22] pour les écoulements incompressibles 2D. La méthode LS-STAG est basée sur la discrétisation MAC usuelle sur grilles cartésiennes décalées où la frontière irrégulière est représentée par sa fonction *level-set*. La discrétisation des équations de Navier-Stokes dans trois types de *cut-cells* génériques en 2D (pentagonales, trapézoïdales et triangulaires) est construite de manière à ce que les invariants globaux de l'écoulement (masse totale, quantité de mouvement totale, énergie cinétique totale) soient strictement conservés au niveau discret. La discrétisation LS-STAG dans les *cut-cells* coïncide avec la discrétisation MAC dans les cellules fluides cartésiennes, et a la propriété de préserver la simplicité du stencil cartésien à 5 points (en 2D), résultant en une méthode numérique très efficace sur le plan du temps de calcul. La version 2D de la méthode LS-STAG a été appliquée avec succès à une grande variété d'écoulements, tels les écoulements newtoniens à nombre de Reynolds modéré

en géométries fixes et mobiles [22], les écoulements pseudoplastiques [15] et les écoulements viscoélastiques [16, 18, 17]. De plus, la discrétisation LS-STAG a également été implémentée dans le logiciel d'hydrodynamique navale *ComFLOW* [99] pour la représentation des géométries irrégulières. Elle a également été étendue au calcul de problèmes aéroélastiques couplés [80], ainsi qu'aux écoulements turbulents avec modèles *RANS*[1] [81], *LES*[2] et *DES*[3] [82].

Les objectifs

L'objectif global de cette étude est d'étendre la méthode LS-STAG aux géométries complexes 3D pour les fluides newtoniens et pseudo-plastiques, dans un code de simulation totalement optimisé pour le calcul haute performance. L'une des difficultés majeures de la discrétisation dans les *cut-cells* 3D est le grand nombre de cas à prendre en compte: nous avons dénombré 16 types différents de *cut-cells* en 3D alors qu'il n'y a que trois types en 2D. Pour aborder ce problème, nous avons considéré une étape intermédiaire de développement où le code LS-STAG sera appliqué à des géométries 3D possédant une symétrie translationnelle dans la $3^{\text{ème}}$ direction, appelées géométries 3D extrudées. Cette étape intermédiaire vers une discrétisation entièrement 3D sera considérée comme la clé de voûte du solveur 3D complet, car les problèmes de discrétisation et d'implémentation sur les machines à mémoire distribuée sont abordés dès cette étape de développement, et la méthode peut être appliquée à une grande variété d'écoulements canoniques (écoulement dans des cellules de type Taylor-Couette, autour de cylindres à section de forme arbitraire, dans des conduites avec singularités, *etc.* $\cdots$). Les objectifs de cette étude par rapport au projet global sont résumés comme suit:

(a) Etendre les principes de la discrétisation de la méthode LS-STAG 2D aux configurations 3D extrudées [22]. En particulier, comme les *cut-cells* 3D ne sont que des extensions extrudées de cellules pentagonales, trapézoïdales et triangulaires en 2D, nous sommes en mesure de concevoir une discrétisation qui vérifie strictement les lois de conservation discrètes dans tout le domaine, tout en conservant la simplicité de la structure à 7 points du schéma.

(b) Implémenter la discrétisation non-newtonienne dans la version 3D extrudée de la méthode LS-STAG.

(c) Paralléliser la version séquentielle du code et évaluer la scalabilité du solveur parallèle LS-STAG.

(d) Etudier la précision de la méthode LS-STAG pour les écoulements newtoniens et non-newtoniens dans les géométries 3D extrudées.

[1] Reynolds-averaged Navier-Stokes.
[2] Large Eddy Simulation.
[3] Detached Eddy Simulation.

(e) Évaluer la versatilité de la méthode en l'appliquant à divers régimes d'écoulements (fluides newtoniens et rhéofluidifiants, écoulements laminaires stationnaires et instationnaires, écoulements turbulents et écoulements granulaires).

(f) Comparer ses performances avec des méthodes numériques bien établies (méthodes non-structurées et de frontières immergées).

Contenu de la thèse

Le plan du manuscrit est le suivant:

Le chapitre 1 présente les équations gouvernant les écoulements de fluide incompressibles. De plus, pour le cas de fluides non-newtoniens (en particulier les fluides rhéofluidifiants), les trois modèles de viscosité non-newtonienne (Loi d'Ostwald–de Waele, de Cross et de Hanotin) utilisés dans les simulations numériques sont introduits.

Le chapitre 2 présente une revue de la littérature pour les méthodes de frontières immergées et de *cut-cell*. Nous énonçons également les lois de conservation globales pour la masse, la quantité de mouvement et l'énergie cinétique qui sont utilisées pour construire la version bidimensionnelle de la méthode LS-STAG. Enfin, le stockage mémoire des paramètres geometriques de grille de la méthode LS-STAG et des méthodes non-structurées sera comparé.

Le chapitre 3 présente la discrétisation des équations de continuité et de quantité de mouvement dans les configurations 3D extrudées. Comme dans la version 2D, la forme discrète de l'équation de continuité, du gradient de pression et des éléments diagonaux du gradient de vitesse (pour le calcul des flux visqueux) prennent des expressions uniques qui sont valables pour tous les types de *cut-cells* ainsi que pour les cellules cartésiennes. Les détails de la discrétisation dite "antisymétrique" (*skew-symmetric*) des flux convectifs seront présentés ensuite. Il est à noter que la discrétisation des composantes non-diagonales du gradient de vitesse (termes $\partial w/\partial x$ et $\partial w/\partial y$) représente la partie la plus complexe de la méthode LS-STAG et nécessite une attention particulière. Ces termes, qui étaient absents dans la méthode 2D, sont discrétisés en utilisant la formule *ad hoc* proposée dans [99]. À la fin du chapitre, le calcul du taux de cisaillement pour l'évaluation locale de la viscosité non-newtonienne sera présenté, et l'algorithme d'avancement en temps des équations de Navier-Stokes sera introduit.

Le chapitre 4 évalue d'abord la scalabilité du solveur LS-STAG parallèle en faisant des simulations de vortex de Taylor-Green dans une boîte 3D périodique. Ensuite, une étude de

convergence est effectuée sur l'écoulement de Hagen-Poiseuille pour les fluides newtoniens et non-newtoniens avec loi de puissance pour la viscosité. Une comparaison approfondie de la précision et des coûts de la méthode (taille mémoire, temps de calcul) par rapport à un logiciel de MFN sur grille non-structurée (ANSYS-FLUENT©) sera effectuée pour un cas-test portant sur l'écoulement autour d'un cylindre circulaire en géométrie confinée, pour lequel des résultats de référence ont été obtenus avec une méthode non-structurée [10]. Enfin, les simulations directes de l'écoulement turbulent autour d'un cylindre à $Re_D = 300$ [66, 59] démontrent que la méthode LS-STAG permet de simuler des écoulement à haut nombre de Reynolds. Pour cet écoulement, une comparaison avec les résultats d'une méthode de frontières immergées bien établie [7] est aussi effectuée.

Le chapitre 5 étudie en détail l'écoulement de fluides rhéofluidifiants dans une conduite cylindrique avec élargissement brusque pour laquelle la caractérisation rhéologique, les mesures expérimentales et les simulations FLUENT© de l'écoulement ont été réalisées dans [86]. Enfin, les calculs numériques de divers écoulements granulaires (secs et suspensions) dans une cellule de Taylor-Couette sont abordés et le résultat des simulations LS-STAG sera comparé aux données expérimentales rapportées par [38].

Introduction (English Version)

Advanced computational fluid dynamics (CFD) applications concern flows in highly irregular three-dimensional geometries whose process of grid generation, from the early stage of CAD design to final volume meshing, is very time-consuming and requires highly trained users. Typical unstructured body-conformal meshes are made of elements (cells of hexahedral, tetrahedral shape, $\cdots$) with irregular connectivity, which makes the solution algorithms extremely memory and CPU intensive. Moreover, it is difficult to assess the quality of the grid (smooth, undistorted cells properly clustered at the wall, $\cdots$), which may deteriorate the convergence and accuracy of the numerical solution. The immersed boundary method has emerged in recent decades as an alternative to the body-fitted mesh methods. They are based on Cartesian methods which are extremely robust and optimized, and thus requires only a fraction of the computer resources of an unstructured grid-based method. A Cartesian grid-based method deals with the bodies which do not necessarily have to conform a Cartesian grid, see for example Figure 1, which significantly facilitates complicated mesh generations and eliminates the need of re-meshing the domain for moving boundary which is required by body-conformal methods.

The immersed boundary methods originally referred to an approach developed by Peskin [78] in 1972 to simulate fluid-structure interactions with elastic boundaries. In this methods, the effects of immersed boundaries are reproduced through adopting an appropriate boundary force in fluid equations. Immersed boundary methods are classified into two major methods based on imposition of the boundary conditions on the immersed boundary [67]; continuous forcing approach in which a forcing term is incorporated into the continuous equations before discretization, and the discrete forcing approach where the forcing is introduced after the equations have been discretized. One of the major issues of the methods using the first approach, is that the sharp interfaces are smeared out which results in degrading of the spatial accuracy for non-smooth solutions [79]. In addition, for the flows with rigid boundaries, applying the continuous forcing poses some problems since the constitutive laws used for elastic boundaries are not necessarily suitable for these types of boundaries. This issue can be alleviated by employing discrete forcing approach. However, most of methods in this group discard the discretization of flow equations in the cut-cells (cells of irregular shape which are formed by the intersection of the Cartesian mesh with the immersed boundary, see Figure 1 (right)), and use special interpolations instead. Thus, the strict conservation of mass is not observed, which leads to non-divergence

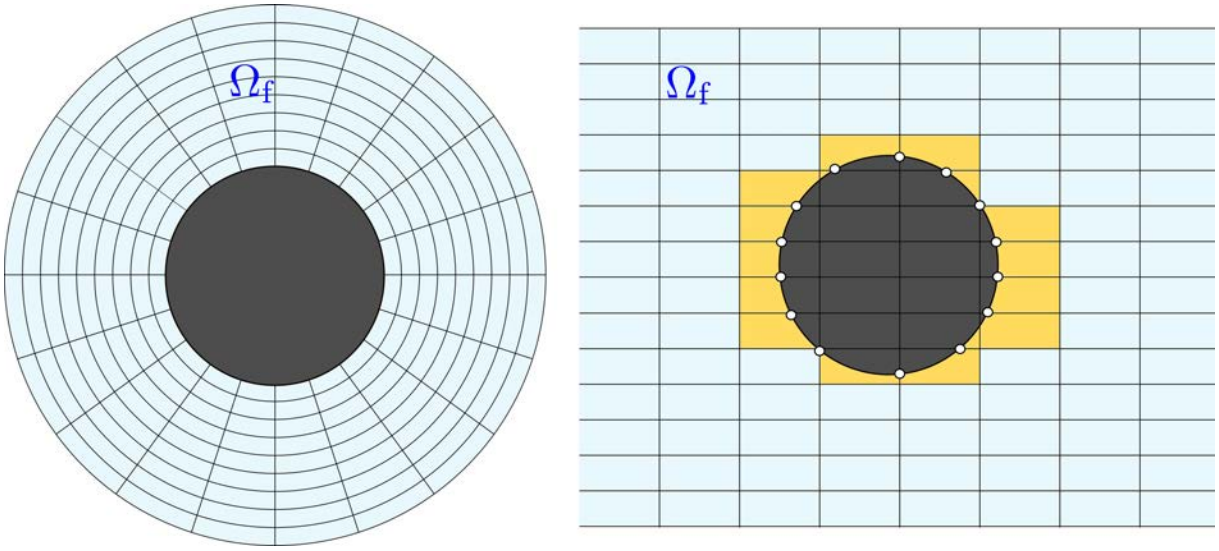


Figure 1: Illustration of an orthogonal structured body-fitted grid (left) and the schematic view of the immersed boundary method in a two-dimensional computational domain Ω_f (right). The cut-cells are highlighted in yellow.

free velocities or non-physical pressure oscillations in the vicinity of the immersed boundary [56]. The sharp representation of immersed boundaries is achieved by employing the cut-cell approach which was initially introduced in the context of Cartesian grid methods for inviscid flows [23]. In this method, the flow equations are discretized in the cut-cells by *ad hoc* treatments, *e.g.* cell-merging techniques, which can ensure that the strict conservation laws are strictly fulfilled. But, the major drawback is that the 5-points structure of Cartesian methods (in 2D) is lost. Moreover, the 3D extension of these methods which rely on cell-merging techniques is known to be non-trivial [67].

An innovative cut-cell discretization for 2D incompressible flows called "LS-STAG" (*Level-Set, STAGgered*) has been developed by Cheny and Botella [20, 22]. LS-STAG is based on the classical MAC method for staggered Cartesian grids where the irregular boundary is sharply represented by its level-set function. The discretization of governing equations in three basic types of 2D cut-cells (pentagonal, trapezoidal and triangular) is designed such that the global invariants of the flow (total mass, momentum and kinetic energy) are strictly preserved at the discrete level. The LS-STAG discretization in the cut-cells is consistent with the MAC discretization used in Cartesian fluid cells, and has the ability to preserve the simplicity of 5-point Cartesian stencil (in 2D), resulting in a highly computationally efficient method. The 2D version of LS-STAG has been successfully applied to a wide variety of flows such as Newtonian flows at moderate Reynolds number in fixed and moving geometries [22], pseudoplastic flows [15], and viscoelastic flows [16, 18, 17]. In addition, the LS-STAG discretization has also been implemented in the ship hydrodynamics software *ComFLOW* [99] which handles with irregular geometries. It has been also extended to the computation of coupled aero-elastic problems [80], turbulent

flows with $RANS$[1] [81] and LES[2]/DES[3] [82] equations modelings. However, the development of an accurate and cost-effective 3D solver based upon the immersed boundary/cut-cell approach is still a challenging task.

Objectives and aims

The overall aim for this investigation is to extend the LS-STAG towards a fully 3D code for Newtonian and shear-thinning liquids, totally optimized for high-performance computing. One of the major difficulty in cut-cell discretization for 3D geometries is the large number of cases to consider: we have enumerated 16 different type of cut-cells in 3D whereas there are only 3 in 2D. In order to mitigate this issue we first consider building the LS-STAG discretization for 3D configurations with translational symmetry in the z direction, that will be subsequently called 3D extruded geometries. This intermediate step towards the fully 3D implementation will be regarded as the milestone for the full 3D solver, since both discretization and implementation issues on distributed memory machines are tackled at this stage of development and can be applied to a wide variety of canonical flows (flow in Taylor-Couette reactors, flow past cylinder of arbitrary cross section shape, abrupt axisymmetric expansion/contraction flow, $\cdots$). The objectives of this project in relation to this overall aim is summarized as follows:

(a) To extend the principles of the energy-conserving discretization of 2D LS-STAG [22] to 3D extruded configurations. In these configurations, since the cut-cells are the extruded extension of pentagonal, trapezoidal and triangular cells appeared in 2D, thus we are able to design a discretization which strictly fulfills the conservation laws in the whole domain as well as retaining the simplicity of 7-point structure whereby the computational cost are significantly reduced.

(b) To implement the non-Newtonian discretization to the 3D extruded version of LS-STAG method.

(c) To parallelize the sequential version of the code and evaluate the scalability of LS-STAG parallel solver.

(d) To investigate the formal order of accuracy for Newtonian and non-Newtonian flows in 3D extruded geometries.

(e) To assess the versatility of this method for flow applications at various regimes (Newtonian and shear-thinning fluids, steady and unsteady laminar to turbulent flows and vibrated granular flows).

(f) To compare its performance with well-established numerical methods (unstructured and immersed boundary methods).

[1] Reynolds-averaged Navier-Stokes
[2] Large Eddy Simulation
[3] Detached Eddy Simulation

Overview of thesis

In this manuscript, the chapters are ordered so that:

Chapter 1 presents the governing equations of motion for the incompressible fluid flows. Additionally, for the non-Newtonian case (particularly shear-thinning fluids), three different viscosity models (*i.e.* power-law, Cross and Hanotin) are introduced to evaluate the local viscosity based on the the given fluid flows conditions in our numerical simulations.

Chapter 2 reviews the literature for the immersed boundary methods and most notably the background of the 2D LS-STAG method. We remind also the global conservation laws for total mass, momentum and kinetic energy which are used to derive the 2D version of LS-STAG. Finally, the LS-STAG method will be compared with unstructured grid-based methods in terms of required memory storage for the geometric parameters.

Chapter 3 presents the discretization of the continuity and momentum equations in 3D extruded configurations. The discrete form of continuity equation, pressure gradient and diagonal elements of the velocity gradient tensor in viscous fluxes are given with unique expressions valid for cut-cells and Cartesian cells in the same fashion as 2D formulas. Subsequently, the details of the skew-symmetric discretization of convective terms will be explained. The off-diagonal components of velocity gradient tensor $\partial w/\partial x$ and $\partial w/\partial y$ involved in viscous fluxes are the most intricate part of the discretization and needs further attention. These terms which were absent from the 2D cases, are discretized based on the formula given in [99]. At the end of this chapter, the shear-rates required in non-Newtonian fluid flows are calculated and the time-advancement algorithm in LS-STAG method will be explained.

Chapter 4 assesses first the parallel scalability of LS-STAG solver through weak and strong measurements of the parallel scaling in simulation of the 3D periodic and transitional Taylor-Green vortex flow. Next, a grid convergence study is conducted on the Hagen-Poiseuille flow for power-law liquids to investigate the order of accuracy. A thorough comparison on the accuracy and computational costs (memory usage, CPU time) with respect to a commercial CFD package on unstructured grid (ANSYS-FLUENT©) will be performed for the flows past a circular cylinder in a confined square duct in laminar steady and unsteady regimes to evaluate the performance of LS-STAG method with the unstructured grid-based methods [10]. The ability to perform computation of high Reynolds number flow is also addressed on the weakly turbulent cylinder flow at $Re_D = 300$ [66, 59], for which comparison with the results of a well-established immersed boundary method [7] is provided.

Chapter 5 studies in details the flow of shear-thinning fluids in an axisymmetric sudden expansion geometry for which the rheological characterization, experimental flow measurements

and FLUENT© simulations have been performed in [86]. Finally, the numerical computations of various granular flows (dry and suspensions) in a Taylor-Couette cell are performed and the acquired result from LS-STAG method will be compared with the experimental data reported by [38].

Chapter 1

Physical modeling

Contents

1.1 Governing equations of fluid dynamics 13
1.2 Rheological behavior of fluids 14
1.2.1 Time-independent fluid behavior 15
1.3 Approximation of the shear-thinning behavior 17
1.3.1 Power-law (Ostwald-de Waele) model 18
1.3.2 Cross model 18
1.3.3 Hanotin model 19
1.4 Shear-thinning viscosity models employed in our numerical simulations 19
1.5 Conclusion 22

In this chapter, we will describe the motion of any incompressible fluid flow by equations of conservation of mass, momentum. For non-Newtonian cases, we also require to introduce a constitutive law to complete the system of equations. There is no general law for all types of fluids and among the numerous of existing models (see for example Bird *et al.* [13]), we choose ones based upon the flow conditions to understand the behavior exhibited by a given material.

1.1 Governing equations of fluid dynamics

The local conservation equation of momentum is written:

$$\rho\left(\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v}\cdot\nabla)\boldsymbol{v}\right) = -\nabla p + \nabla\cdot\boldsymbol{\tau}, \tag{1.1}$$

where $\boldsymbol{v} = (u, v, w)$ is the velocity vector field, p is the pressure field, $\boldsymbol{\tau}$ is the stress tensor and ρ is the density of fluid. As the fluid is assumed incompressible, thus the velocity field $\boldsymbol{v}$ satisfies the continuity condition:

$$\nabla\cdot\boldsymbol{v} = 0. \tag{1.2}$$

Considering this assumption for the tensor identity:

$$(\boldsymbol{v}\cdot\nabla)\,\boldsymbol{v} = \nabla\cdot(\boldsymbol{v}\otimes\boldsymbol{v}) - \boldsymbol{v}\,\nabla\cdot\boldsymbol{v}, \tag{1.3}$$

allows us to acquire an alternative form of Equation (1.1) such that:

$$\rho\left(\frac{\partial\boldsymbol{v}}{\partial t} + \nabla\cdot(\boldsymbol{v}\otimes\boldsymbol{v})\right) = -\nabla p + \nabla\cdot\boldsymbol{\tau}. \tag{1.4}$$

This form of momentum equation (1.4) is more convenient to get the integral form of equations of motion which is the starting point of finite volume discretization. By applying the divergence theorem, ones can transform a volume integral of a scalar field of $\boldsymbol{\nabla}\cdot\boldsymbol{a}$ over the control volume Ω to a surface flux such that:

$$\int_\Omega \boldsymbol{\nabla}\cdot\boldsymbol{a}\ \mathrm{d}V = \int_\Gamma \boldsymbol{a}\cdot\boldsymbol{n}\ \mathrm{d}S, \tag{1.5}$$

where $\boldsymbol{n}$ is an outwardly directed normal unit vector of surface Γ of the control volume Ω. Hence, the integral form of continuity equation (the conservation law of mass) is determined as:

$$\int_\Gamma \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S = 0. \tag{1.6}$$

In addition, the following expression:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \rho\,\boldsymbol{v}\ \mathrm{d}V + \int_\Gamma \rho\,\boldsymbol{v}\,(\boldsymbol{v}\cdot\boldsymbol{n})\ \mathrm{d}S + \int_\Gamma p\,\boldsymbol{n}\ \mathrm{d}S - \int_\Gamma \boldsymbol{\tau}\cdot\boldsymbol{n}\ \mathrm{d}S = 0, \tag{1.7}$$

describes the conservation of momentum inside the control volume Ω. Accordingly, the integral form of momentum equation (1.7) consists of the terms:

- the rate of change of the momentum within the control volume, $\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \rho\,\boldsymbol{v}\ \mathrm{d}V$,
- the convective flux tensor $\int_\Gamma \rho\,\boldsymbol{v}\,(\boldsymbol{v}\cdot\boldsymbol{n})\ \mathrm{d}S$ over the surface Γ,
- the pressure flux $\int_\Gamma p\,\boldsymbol{n}\ \mathrm{d}S$,
- the diffusive flux $\int_\Gamma \boldsymbol{\tau}\cdot\boldsymbol{n}\ \mathrm{d}S$.

The aforementioned terminology will be employed later in the manuscript. According to Equations (1.6) and (1.7) in 3D, 13 unknowns are involved in these 4 sets of equations (1 continuity and 3 momentum); 3 components of velocity vector $\boldsymbol{v}$, the pressure term and 9 components of stress tensor $\boldsymbol{\tau}$. In order to complete system of equations, one must define an extra equation, hereafter the constitutive law. The constitutive law can be obtained for specific fluid types.

1.2 Rheological behavior of fluids

In order to have the "ready for use" form of the equations of motion, the stress tensor must be determined regarding the viscous behavior of fluid. A Newtonian behavior is related to a fluid

in which the shear stress is directly proportional to the shear-rate such that the constant of proportionality is the familiar dynamic viscosity η [28]. That is, the flow behavior of Newtonian fluids is described with a simple linear relation between shear stress and shear-rate. Thus, the viscosity of fluid, which is the slope of shear stress against shear-rate in Figure 1.1, will remain constant for a Newtonian fluid. Water, air, low molecular weight liquids, and solutions of low molecular weight substances in liquids exhibit the Newtonian behavior. For a Newtonian fluid, the momentum equations are significantly simplified to Navier-Stokes equations taking into account the assumption of constant viscosity. In a Newtonian fluid, the shear stress tensor $\boldsymbol{\tau}$ is defined such that:

$$\boldsymbol{\tau} = \eta\,\dot{\boldsymbol{\gamma}}, \tag{1.8}$$

where $\dot{\boldsymbol{\gamma}} = (\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^{\mathrm{T}})$ is the rate-of-strain tensor [13]. However, in reality most of the fluids especially those with complicated microscopic structure, are non-Newtonian and may present a dependence of viscosity against shear-rate (shear-thinning or shear-thickening) or the deformation history (*e.g.* thixotropic). A non-Newtonian behavior which is also known *non-linear, complex* or *rheologically complex* behavior, is referred to a fluid with non-linear relationship between the shear stress and the shear-rate. Thus, the viscosity is no longer constant and it is a function of the shear stress or the shear-rate. Furthermore, the viscosity of certain materials is not only a function of flow conditions (*e.g.* shear-rate), but also depends on the kinematic history of the fluid element under consideration.

Essentially, the non-Newtonian behaviors are classified into three major groups of "*time-independent*", "*time-dependent*" and "*viscoelastic*" fluids based on the nature of fluid viscosity [28]. A time-independent fluid behaves such that the viscosity depends only on shear stress or the shear-rate and not on the duration of shear. A time-dependent fluid displays a behavior such like its viscosity depends not only on the shear stress or the shear-rate at any instant, but also on the previous kinematic history of the fluids. The time-dependent fluid behavior such as waxy crude oil, hand lotions and creams, are also subdivided into two types of *thixotropy* and *rheopexy* (or negative thixotropy) concerning to the response of the material to shear over a period of time, for further details refer to [28]. A viscoelastic material exhibits a response which resembles that of an elastic solid under some circumstances or the response of a viscous liquid under other circumstances. Typically, the high-molecular-weight fluids such as polymeric fluids (melts and solutions) display viscoelastic behavior. In this chapter, we will only investigate the time-independent fluids.

1.2.1 Time-independent fluid behavior

The time-independent behavior known as *GNF*[1], refers to the fluids in which the current value of the shear-rate at a point in the fluid is determined only by the corresponding current value of the shear stress and vice versa. In other words, such fluids have no memory of their past history. The time-independent fluids can be also subdivided into three types as a function of shear-rate:

[1] generalized Newtonian fluids

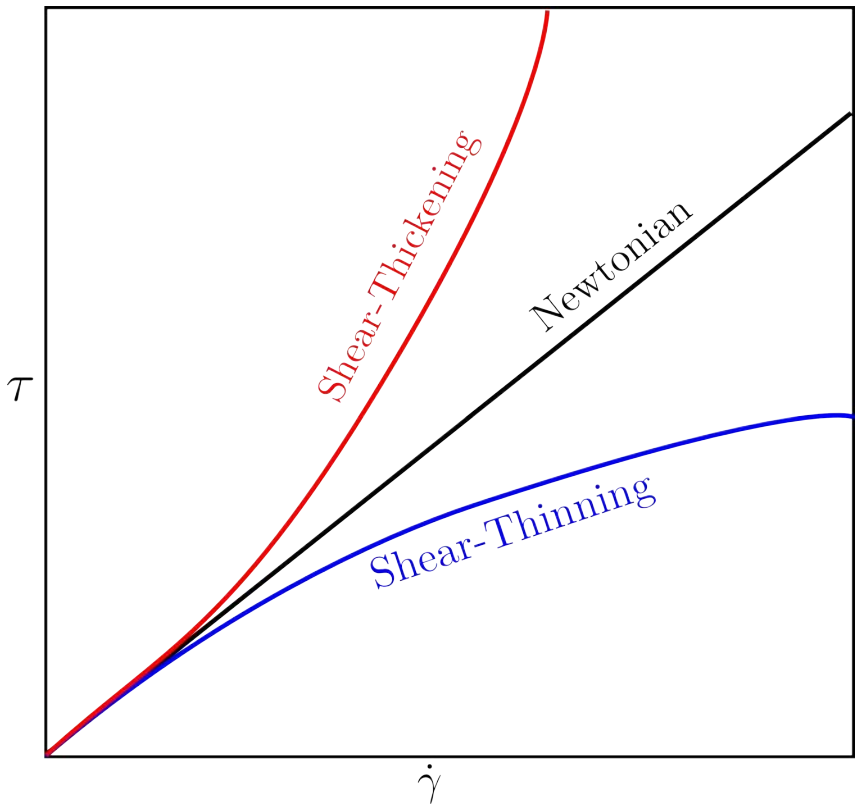


Figure 1.1: Shear stress $\boldsymbol{\tau}$ as a function of shear-rate $\dot{\gamma}$.

Shear-thinning : is the non-Newtonian behavior of a fluid whose viscosity decreases as the shear-rate increases, see Figure 1.1. Shear-thinning fluids, also known as pseudo-plastics, are the most common type of time-independent non-Newtonian fluid behavior and mainly employed in industrial and biological processes. Many polymer solutions, ketchup, paints and blood displays shear-thinning behavior. At extreme low and high shear-rates, the shear-thinning fluids exhibit Newtonian behavior and their viscosities are independent of the shear-rate. Figure 1.2 demonstrates the behavior of fluids in the shear-thinning regime of viscosity from the zero-shear to infinite-shear viscosity. The viscosity approaches a Newtonian plateau at infinite slow shear which is called zero-shear viscosity, η_0. At the other extreme, very high shear-rates, the viscosity also tends asymptotically to a similar plateau which is called infinite-shear viscosity, η_∞. In most cases, η_∞ is only slightly greater than the solvent viscosity η_s [28]. The behavior between these to regions can usually be approximated by a straight line which is known as the power-law region.

Viscoplastic : is a non-Newtonian behavior of a fluid in which the fluid is not flow when the imposed stress is below a certain critical threshold (yield stress σ_0), and the material behaves like a rigid solid (an elastic solid). If the stress in the fluid exceeds the yield stress, then it deforms and flows like a viscous fluid. In this case, the fluid may exhibit Newtonian (constant η) or shear-thinning behaviors ($\eta(\dot{\gamma})$). For the stresses higher than σ_0, if a fluid

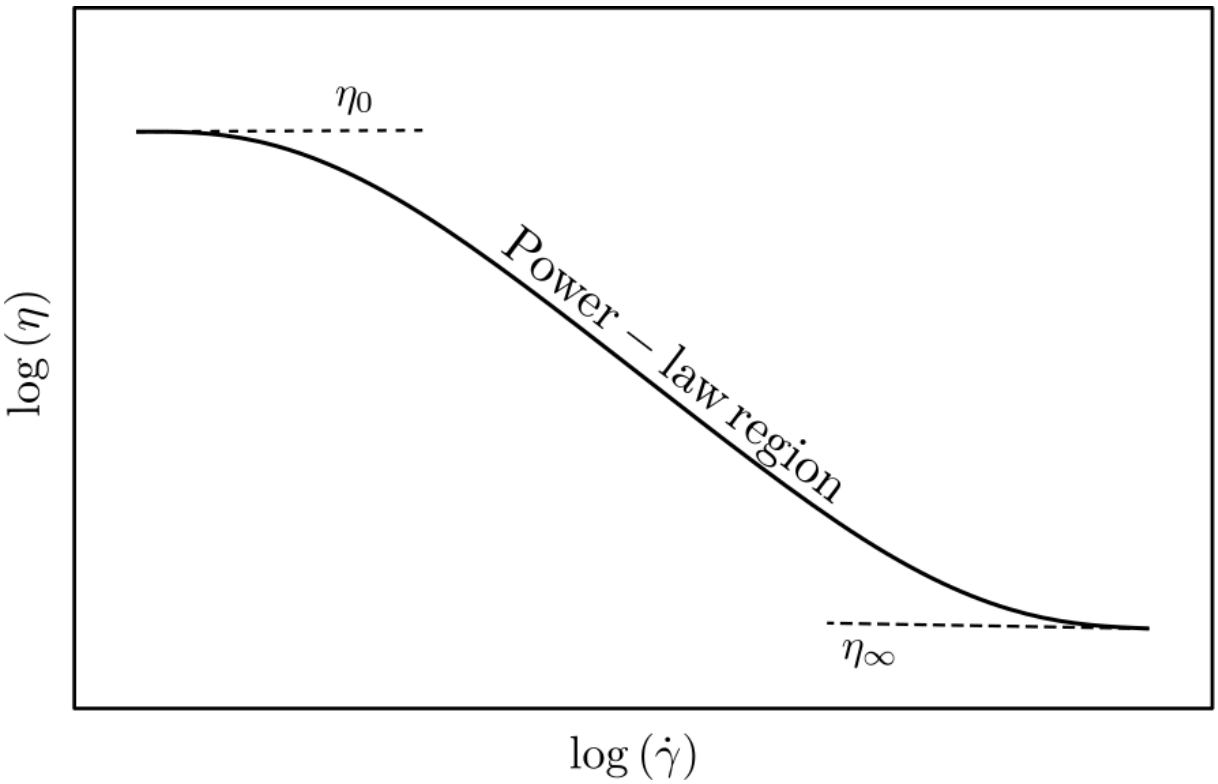


Figure 1.2: Demonstration of zero-shear η_0 and infinite-shear η_∞ viscosities in the shear-thinning regime.

indicates Newtonian behavior with a linear flow curve, this fluid is called a Bingham plastic fluid, otherwise it shows a shear-thinning behavior and usually approximated by Herschel-Bulkley fluid model, refer to [28] for further information. Practically, such fluid behavior appears in slurries and suspensions, muds and clays, heavy oils, cosmetic creams, hair gel, liquid chocolate, and *etc.*.

Shear-thickening : is a fluid behavior in which the viscosity of the fluid increases as the shear-rate increases, see Figure 1.1. This behavior was originally observed in concentrated suspensions. The mixture of cornstarch and water (Oobleck) or Silica nano-particles dispersed in a solution of polyethylene glycol display the shear-thickening behavior. Shear-thickening (dilatant) materials are much less widespread in the chemical and processing industries. They have also specific uses in military applications such as reinforcing the body armors.

1.3 Approximation of the shear-thinning behavior

In this manuscript, we will only focus on the Newtonian and shear-thinning fluids. Fluid flow is highly dependent on the viscosity. In non-Newtonian behavior, the viscosity is no more a constant scalar and is a function of the magnitude of shear-rate, $\eta(\dot{\gamma})$. According to Bird [13],

the shear-rate depends on the second invariant of the rate-of-strain tensor $\dot{\boldsymbol{\gamma}}$ such that:

$$\dot{\gamma} = \sqrt{\tfrac{1}{2}\dot{\boldsymbol{\gamma}} : \dot{\boldsymbol{\gamma}}} = \sqrt{[\dot{\gamma}_{xx}]^2 + [\dot{\gamma}_{yy}]^2 + [\dot{\gamma}_{zz}]^2 + [\dot{\gamma}_{xy}]^2 + [\dot{\gamma}_{xz}]^2 + [\dot{\gamma}_{yz}]^2}, \tag{1.9}$$

in which $\dot{\gamma}_{ij}$ is determined such that $\dot{\gamma}_{ij} = \partial v_i/\partial x_j + \partial v_j/\partial x_i$.

Let us now turn back to the shear stress equation, $\boldsymbol{\tau} = \eta(\dot{\gamma})\,\dot{\boldsymbol{\gamma}}$, where viscosity η depends on shear-rate $\dot{\gamma}$ for non-Newtonian fluids. For shear-thinning behavior, we can resort too many mathematical expressions proposed in the literature to calculate $\eta(\dot{\gamma})$; some of these are straightforward attempts at curve fitting and give empirical relationships for the shear stress such as power-law, Cross and Carreau-Yasuda models while others have some theoretical basis in statistical mechanics. A more complete description of such models are available in literature *i.e.* [13, 28, 84] and we cite only power-law, Cross and Hanotin [46] models.

1.3.1 Power-law (Ostwald-de Waele) model

The power-law model offers the simplest representation of shear-thinning behavior. The viscosity in the so-called power-law model takes the form of:

$$\eta(\dot{\gamma}) = k\,\dot{\gamma}^{n-1}, \tag{1.10}$$

where k ($Pa.s^n$) and n (dimensionless) are two empirical curve-fitting parameters and are known as the fluid consistency coefficient and the flow behavior index, respectively. The consistency of a substance is measured by the consistency coefficient k. Based on the value of power-law index, power-law fluids can be subdivided into three different class of fluids. Shear-thinning behavior is characterized by a value of n smaller than unity. For $n = 1$, the fluid exhibits Newtonian behavior. For the shear-thickening fluids, the power-law index n is higher than unity. For a shear-thinning fluid, the power-law index may have any value between $0 < n < 1$ such that the fluids with higher degree of shear-thinning behavior possess smaller value of power-law index n. The power-law model applies over only a limited range of shear-rates. It is incapable to predict neither the upper (zero-shear (η_0) viscosity) nor the lower (infinite-shear (η_∞) viscosity) Newtonian plateau in Figure 1.2. Despite these restrictions, the power-law model is perhaps the most widely used in the literature dealing with engineering applications. Its simplicity allows to derive analytical solutions of the Navier-Stokes equations for canonical flows [13].

1.3.2 Cross model

At very high and very low shear-rates, we have the significant deviations from the power-law model. Hence, we should employ another models which take into account the values of viscosities η_0 and η_∞. The Cross model is a four parameter model originally proposed by Cross [26] in which the viscosity of the solution can be correlated with shear-rate using:

$$\eta(\dot{\gamma}) = \eta_\infty + \frac{(\eta_0 - \eta_\infty)}{1 + (K\,\dot{\gamma})^m}, \tag{1.11}$$

where K (s) is the Cross time constant and m which is known as the Cross rate constant is a dimensionless parameter for measuring the degree of dependence of viscosity on shear-rate in the shear-thinning region. Cross [26] proposed an admissible fraction of 2/3 for m that was a satisfactory estimation for numerous substances. Nowadays, a significant improvement in terms of the degree of fit is achieved by treating dimensionless constant m as an adjustable parameter [8]. One can recover the Newtonian fluid behavior at low shear-rates ($K \to 0$). Similarly, the Cross model simplifies to the familiar power-law model at high shear-rates [39]. In this case, the Cross parameters can be correlated to the power-law parameters (1.10) by observing that $k = \eta_0 K^{-m}$ and $n = 1 - m$ in the limiting case where $\eta_0 \gg \eta_\infty$, $K\dot{\gamma} \gg 1$ and η_∞ is small.

1.3.3 Hanotin model

In this section, we introduce another model, hereinafter called Hanotin model. This model is proposed by Hanotin [46] for predicting the rheological response of both vibrated and sheared granular suspensions in stationary and non-stationary conditions. According to this model which is an extension of the Cross model, the viscosity is defined such that:

$$\eta(\dot{\gamma}) = \frac{\eta_0 + \eta_{\mathrm{H}}\,(\dot{\gamma}/\dot{\gamma}_{\mathrm{c}})}{1 + \dot{\gamma}/\dot{\gamma}_{\mathrm{c}}}, \tag{1.12}$$

where η_{H} $(Pa.s)$ represents the viscosity of suspensions at high shear and $\dot{\gamma}_{\mathrm{c}}$ (s^{-1}) is the critical shear-rate. In this expression, η_0, η_{H} and $\dot{\gamma}_{\mathrm{c}}$ are the fitting parameters.

1.4 Shear-thinning viscosity models employed in our numerical simulations

In this part, we will explain why we will employ the models discussed in Section 1.3, in our computations of non-Newtonian flows. In Chapter 4, we will study the grid convergence on the Hagen-Poiseuille flow for shear-thinning fluids in order to evaluate the order of accuracy of our method. In this simulation, the viscosity of fluid will be defined based on the power-law model which allows us to derive the analytical solution for this flow.

In Chapter 5, two different types of non-Newtonian flows will be investigated. First, we will study the flow of shear-thinning fluids in an axisymmetric sudden expansion geometry whose rheological characterization and experimental measurements are reported in Rigal [86]. In Rigal's work [86], the non-Newtonian fluids are aqueous solutions of Xanthan (SATIAXANE CX 930, Cargill France) at different mass concentrations (0.1%,0.2%,0.3%,0.4%). The flow curves extracted from [86] are displayed in Figure 1.3 for different Xanthan solutions. All these flow curves describe that the shear-thinning behavior increases with the concentration. In addition, Rigal [86] stated that the consistency of Xanthan solution increases strongly with the concentration. According to this figure, we can clearly see that the shear-thinning characters of these solutions are fitted very well to the Cross model (1.11). Thus, in our numerical computations

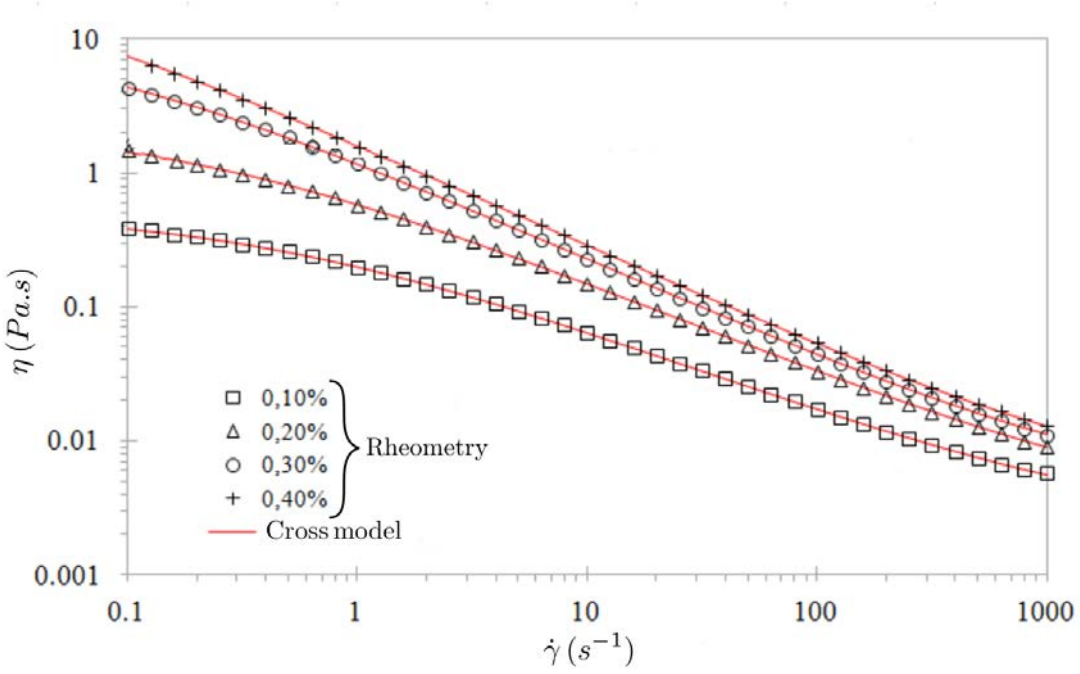


Figure 1.3: Flow curves of the 4 Xanthan solutions reprinted from [86]: viscosity as a function of the shear-rate. Comparison between rheometric measurements (symbols) and the Cross model (solid line).

for the flow through the expansion, detailed in Chapter 5, the Cross model will be employed to predict the shear-thinning viscosity. Note that in this manuscript, we will take into account only two mass concentrations of Xanthan solution *i.e.* 0.1% and 0.3%, whose rheological parameters are given in Table 1.1. Note that the parameters of power-law model is used in the inlet velocity profile required for expansion flow.

Table 1.1: Parameters of the power-law and Cross models for 4 Xanthan solutions reported by Rigal [86].

Fluid type	n	k (Pa.s^n)	η_0 (Pa.s)	η_∞ (Pa.s)	m	K (s)
Xanthane 0.1%	0.543	0.128	0.376	0.00193	0.631	1.64
Xanthane 0.2%	0.360	0.596	4.07	0.00335	0.741	9.09
Xanthane 0.3%	0.322	1.11	9.84	0.00480	0.764	13.1
Xanthane 0.4%	0.281	1.58	47.3	0.00540	0.769	76.4

The second flow which will be studied in Chapter 5, is the steady flow of granular particles in a Taylor-Couette cell. The granular matter shows both solid and fluid behaviors. In the dry case without interstitial fluid, the rheology is solely governed by momentum transfer and energy dissipation occurring in direct contacts between grains and with the walls. The granular

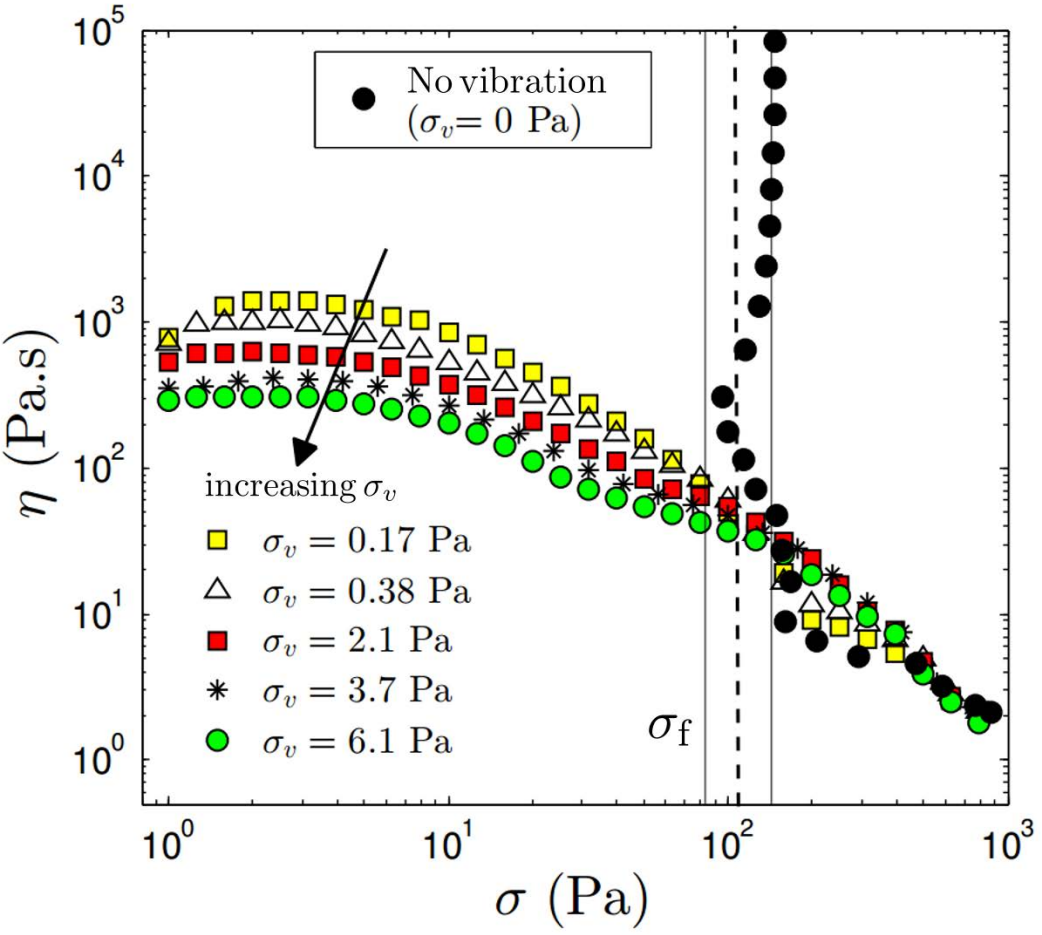


Figure 1.4: Viscosity η as a function of the shear stress σ for different values of vibration stress σ_v for the glass beads of 100 μm immersed in a Newtonian fluid with the viscosity of $\eta_f = 67.8\ mPa.s$ reprinted from Hanotin [46]. The rectangular zone display the estimation of frictional stress σ_f.

suspensions commonly involves mixture of particles with different sizes higher than a few hundred micrometers that are completely or partially filled with an interstitial fluid. These materials are widely found in industrial processes such as food manufacturing (cereals), pharmaceutical manufactures (powders, medicines) and so on. They exhibit complex rheological properties due to the existence of a variety of interactions (frictional, lubricated, collisional) that add up to the physical properties of particles (volume fraction, friction coefficient). The granular suspensions exhibit several non-Newtonian behaviors such as shear-thinning, shear-thickening or yield stress in the steady state regime according to the flow conditions considered. Hanotin [46] stated that in the absence of vibration, the granular suspensions does not flow as long as the imposed stress does not exceed the frictional stress σ_f. For the stress $\sigma > \sigma_f$, the suspension behaves such like a yield stress fluid when no vibration is applied. In the presence of vibration, the yield stress

behavior disappears and the suspensions have a shear-thinning behavior. Figure 1.4 displays the behavior of viscosity against the shear-stress σ for a granular suspension of the glass beads of 100 μm immersed in a Newtonian fluid with the viscosity of $\eta_{\mathrm{f}} = 67.8\ mPa.s$ reported by Hanotin [46]. The yield stress behavior of the suspension in the absence of vibration can be clearly observe in this figure. In addition, we can see that it behaves such a shear-thinning fluid if the system is subjected to the vibration. In this case, by increasing the vibration stress (σ_{v}), the viscosity of suspension decreases. Hanotin also reported that in the presence of vibration, the suspension behavior matches very well with model (1.12), so we will use this model in the numerical simulation of the granular flows in Chapter 5.

It is noteworthy to mention that all computations performed in this manuscript, are characterized by the Reynold number. The Reynolds number is a dimensionless parameter which helps us to predict similar flow patterns in different fluid flow situations and represents the ratio of inertial forces to viscous forces. In all our computations, the Reynolds number is defined such that:

$$Re_D = \frac{\rho\, U_{\mathrm{ref}}^{2-n}\, D_{\mathrm{ref}}^{n}}{k}, \tag{1.13}$$

where D_{ref} and U_{ref} are the reference length and velocity, respectively. Note that for the flow through the expansion, the power-law index and the consistency coefficient involved in Reynolds number definition, are obtained by adjusting the power-law model to the flow curves of Xanthan solutions reported by Rigal [86]. These values are listed in Table 1.1. For the granular flows investigated in Section 5.2, the Reynolds number is defined such like $n = 1$ and:

$$k = \frac{\eta_0 + \eta_{\mathrm{H}}\left((U_{\mathrm{ref}}/D_{\mathrm{ref}})/\dot{\gamma}_{\mathrm{c}}\right)}{1 + (U_{\mathrm{ref}}/D_{\mathrm{ref}})/\dot{\gamma}_{\mathrm{c}}},$$

where $\dot{\gamma} = U_{\mathrm{ref}}/D_{\mathrm{ref}}$ represents the reference shear-rate in Equation (1.12).

1.5 Conclusion

The governing equations of motion for the incompressible fluid flow which will be solved numerically have been described in this chapter. Subsequently, we have discussed the behavior of non-Newtonian fluids most notably the shear-thinning fluids and three different models, namely power-law, Cross and Hanotin, have been introduced in order to calculate the local viscosity in non-Newtonian flows.

Chapter 2

Background of immersed boundary/cut-cell methods and 2D LS-STAG

Contents

2.1 Review of immersed boundary/cut-cell methods 23
2.1.1 Ghost-cell finite-difference approach 26
2.1.2 Cut-cell finite-volume approach 27
2.2 Background of LS-STAG method for 2D configurations 30
2.2.1 Principles and description of LS-STAG mesh 30
2.2.2 Global conservation laws for viscous incompressible flows 35
2.3 Advantage of LS-STAG over unstructured grid-based methods in terms of memory efficiency 40
2.4 Conclusion 42

2.1 Review of immersed boundary/cut-cell methods

One of the most important steps in the numerical solutions of the fluid dynamics equations, is the decomposition of the domain into a discrete grid. The discretization method for the domain depends on the numerical technique applied to solve the equations. Nowadays the most used grids for domain discretization are *body-fitted*, both structured and unstructured. In the body-fitted grids, the boundaries are aligned with a grid system which can simplify the accurate application of boundary conditions. However, generating a high quality boundary fitted mesh is cumbersome especially for the complex flow structures over intricate boundary geometries. In addition, the transformation of the governing equations to curvilinear coordinates results in complex systems of equations which may affect the stability, convergence, and number of operations for the solution. An alternative of the body-fitted methods is proposed with the *immersed*

boundary methods which use a regular underlying Cartesian non-body-fitted grid. Thus, the shape of body does not necessarily conform with the grid system. Although imposing boundary conditions are not as straightforward as in body-fitted methods, the easy grid generation process makes the immersed boundary methods an attractive choice to handle complex geometries. Another advantage is that the need to regenerate grids in time is eliminated for the moving boundaries due to the stationary non-deforming Cartesian grid. Also, the memory and CPU costs in an immersed boundary method is much less than a body fitted grid since the Cartesian grids reduce the per-grid-point operation counts due to absence of additional terms associated with grid transformations. Furthermore, the interface between different materials in the multi-phase and multi-material problems can be regarded as immersed boundary problems. All these properties promote extensive interests in using the immersed boundary methods over the last decades.

The term immersed boundary method was first used in reference to an approach developed by Peskin [78] in 1972 to study flow patterns around the heart valves which has evolved into a generally useful method for problems of fluid-structure interaction. In fact, the immersed boundary method is a mixture of Eulerian and Lagrangian variables so that the Eulerian variables (fluid) are defined on a fixed Cartesian mesh and the Lagrangian ones (structure) are represented on a curvilinear mesh which moves freely through the fixed Cartesian mesh without any constraint [79]. In this method, a boundary force is often introduced to mimic the presence of solid boundary which requires modifying the governing equations in the vicinity of the boundary. Including a forcing function in the equations are carried out in two different ways: the"*continuous*" and "*discrete*" forcing approaches.

In continuous forcing approach, an external body force is directly included into the momentum equation which results in a force formulation independent of the numerical scheme. In this approach, the "elastic" and "rigid" boundaries requires different treatments in an immersed boundary formulation. For the flows with elastic boundaries such that of Peskin [78], the immersed boundary is represented by elastic fibers whose locations are tracked by the massless points that move with the local fluid velocity. The effect of the immersed boundary on the surrounding fluid is essentially captured via a distributed force function which is imposed on the momentum equation. Several distribution functions can be found in the literature *e.g.* [12, 60, 78, 87]. For the flows that involve rigid boundaries, applying the continuous forcing poses some problems as the constitutive laws used for elastic boundaries are not necessarily suitable for the rigid boundaries. This issue can be tackled by employing models such as the virtual boundary method [43] or the penalization techniques [57], however the parameters introduced in these methods affect the numerical accuracy and stability due to the stiffness of the system. Satisfactory results have only been attained for low Reynolds number flows with moderate unsteadiness *e.g.* [57, 43]. The volume penalization method [2] is a particular example of the penalization techniques. This method, which is inspired by the physical intuition that a solid wall is similar to a vanishingly porous medium, is more attractive for computing fluid

flows for small viscosity values, *i.e.* flows at high Reynolds numbers, see for example [92, 91].

Nevertheless, the main problem associated with the continuous forcing approach is that smoothing of the forcing functions smears out the sharp representation of the immersed boundary which is often undesirable at high Reynolds numbers. For high Reynolds numbers, however, a substantial amount of grid points can be located inside the body which leads to a burdensome overhead [67]. In discrete forcing approach, the boundary condition is applied directly into the discrete equations. This approach is very well suited for flows around rigid bodies. The key elements of this approach are the absence of stiffness or user defined parameters that can impact the stability of the method, the ability to represent sharp immersed boundaries by imposing the boundary conditions directly on the numerical scheme and the fact that computing flow variables inside a rigid body becomes unnecessary. The main advantage of the sharp interface is that high-order local accuracy can be achieved around the immersed boundary. However, considering the boundary motion is difficult and unlike continuous forcing, a pressure boundary condition should be imposed on the immersed boundary, see Udaykumar *et al.* [96]. Another drawback is that the implementation of the forcing function is intimately connected to the discretization of the governing equations.

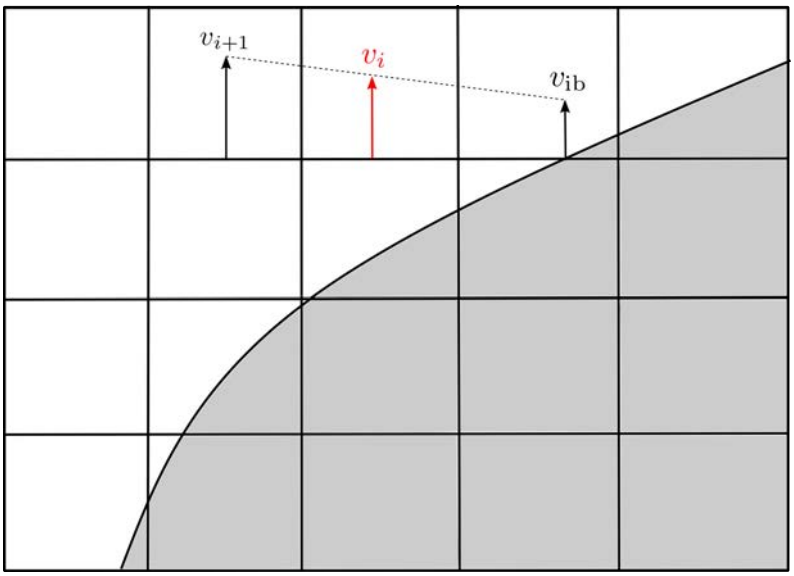


Figure 2.1: Sketch of velocity interpolation in the vicinity of immersed boundary.

The methods that fall under the discrete forcing approach are categorized into methods that enforce the boundary conditions indirectly on the immersed boundary and those that directly impose the boundary conditions. In indirect imposition of the boundary conditions, a forcing term is added to the discrete system of governing equations in the cells near the immersed boundary. In general, the forcing term is derived through estimation of the velocity field and correcting it at the boundary to fit the boundary conditions. In this direction, Fadlun *et al.* [32] implemented a discrete-time forcing approach in which the velocity at the first grid point

external to the body is obtained by linear interpolation of the velocity at the second grid point (computed by directly solving the Navier-Stokes equations) and the velocity at the body surface, see for instance v_i in Figure 2.1 which is calculated from interpolation of v_{i+1} and v_{ib}. Since the velocity boundary condition is enforced with implicit forcing, there is no severe limit on the time-step. However, coupling between the solutions across the immersed boundary via discretized operators which disobey the pressure decoupling constraint results in mis-estimation of the flow fields. Some improvements have been reported in [7, 42, 107], but none of them verify the conservation laws. In addition, all these methods violate the wall condition in the discrete equation system during time advancement.

In order to alleviate these issues, the computational stencil near the immersed boundary can be modified to directly implement the boundary condition on the immersed boundary which results in improving the local accuracy near the boundaries. Note that the immersed boundary is also retained as a sharp boundary. In the following, we will introduce two methods which benefit from the direct boundary condition imposition.

2.1.1 Ghost-cell finite-difference approach

The idea is to achieve a higher-order representation of the boundary using a ghost zone inside the body. In this method, the "ghost cells" are the cells in the solid that have at least one neighbor in the fluid, see for example point G in Figure 2.2. Then, by employing an interpolation scheme (*e.g.* in [62, 40, 53]) for each ghost-cell, the boundary condition is enforced on the immersed boundary. By employing a generic interpolation formula, the value of flow variable at the ghost

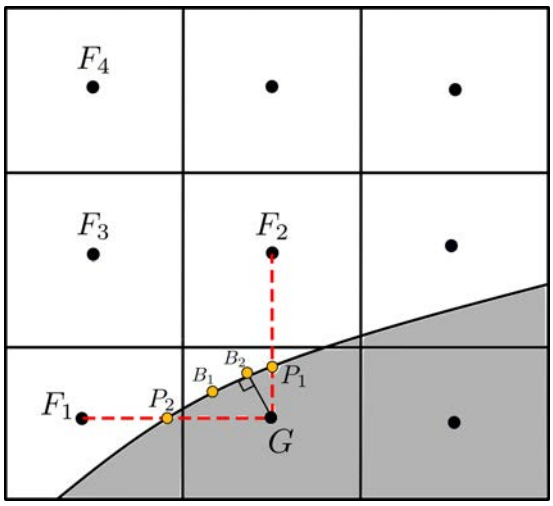


Figure 2.2: Schematics of the points in the vicinity of an immersed boundary used in the ghost-cell approach. F_i represent the fluid points, G denotes the ghost point and B_i and P_i are the locations where the boundary condition can be enforced.

node ϕ_{G} is expressed as:

$$\sum \omega_i \, \phi_i = \phi_{\mathrm{G}},$$

where ω_i are the known geometry dependent coefficients and ϕ_i expresses the points in the stencil including one or more boundary points. The advantage of the ghost cell approach is that the fluid equations are no longer solved inside the solid, thus it significantly saves the CPU time especially at high Reynolds number. This method has been employed in a wide variety of flows such as aquatic propulsion [68], flow through a rib-roughened serpentine passage [51] and turbulent flow past a road vehicle [52].

2.1.2 Cut-cell finite-volume approach

None of the aforementioned immersed boundary methods ensures to satisfy the conservation laws for the cells next to the immersed boundary. The primary motivation of the cut-cell methodology is to provide a method that guarantees the strict global and local conservation of mass and momentum which is accomplished by employing a finite volume approach. The cut-cell method focuses on the discretization of the equations in the mesh cells cut by the immersed boundary.

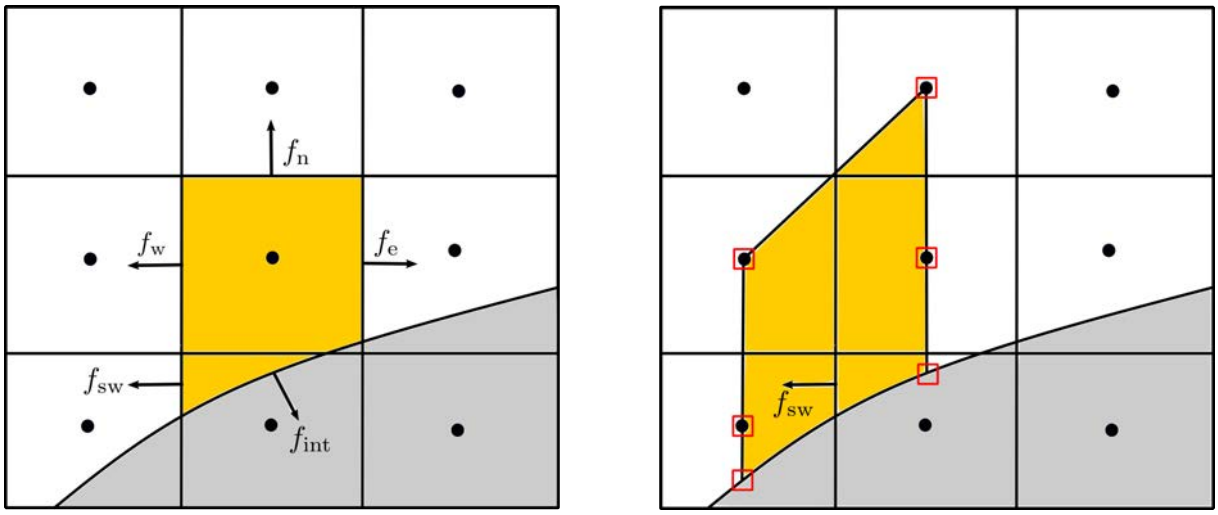


Figure 2.3: Schematic of re-shaped cell next to immersed boundary with associated fluxes (left); 6-point interpolation stencil for determining the flux over the southwest cell face (right). Figures are based on [106].

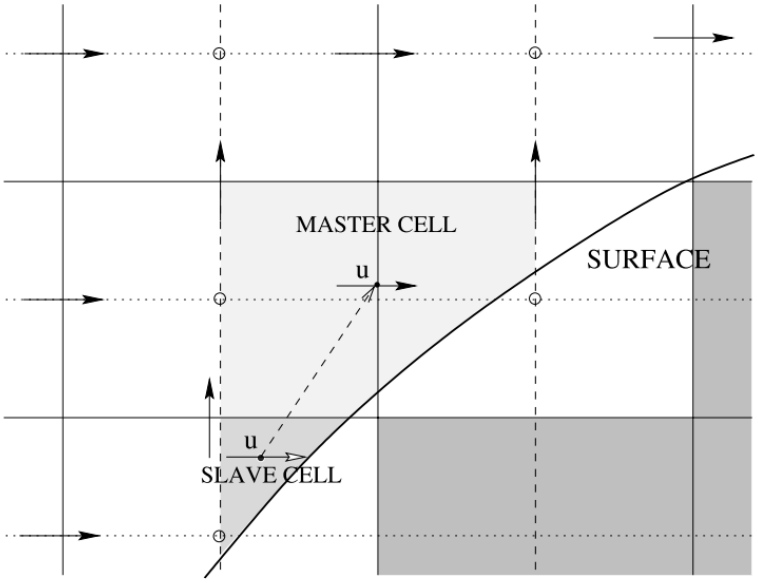
Ye *et al.* [106] developed a cut-cell method to simulate the convection-dominated flows on a Cartesian collocated grid in which a central difference interpolation scheme with the second-order spatial accuracy is used near the immersed boundary. In this method, the cells cut by the immersed boundary are identified and the intersection of the boundary with the sides of these cut-cells is determined. Then, if the center of the original cell lies in the fluid, the reshaped cell becomes an independent new control volume. Otherwise the cut-cell may be merged with a neighbor to avoid the creation of new cells that are significantly smaller than the surrounding ones which leads to problems with stiffness of the equations and numerical

stability, refer to [95, 97] for more details. Such a newly formed cell is indicated in Figure 2.3. The next step is to approximate the flux integrals of mass, convection and diffusion as well as pressure gradients on the faces of each cell. For this purpose, Ye *et al.* [106] proposed to formulate a two-dimensional polynomial interpolating function that expresses a flow variable ϕ and to determine the coefficients by evaluating ϕ in a number of fluid nodes and points on the immersed boundary. As an example, let us consider the flux f_{sw} in Figure 2.3. This flux is expressed based on a 6-point interpolation stencil that is linear in x and quadratic in y:

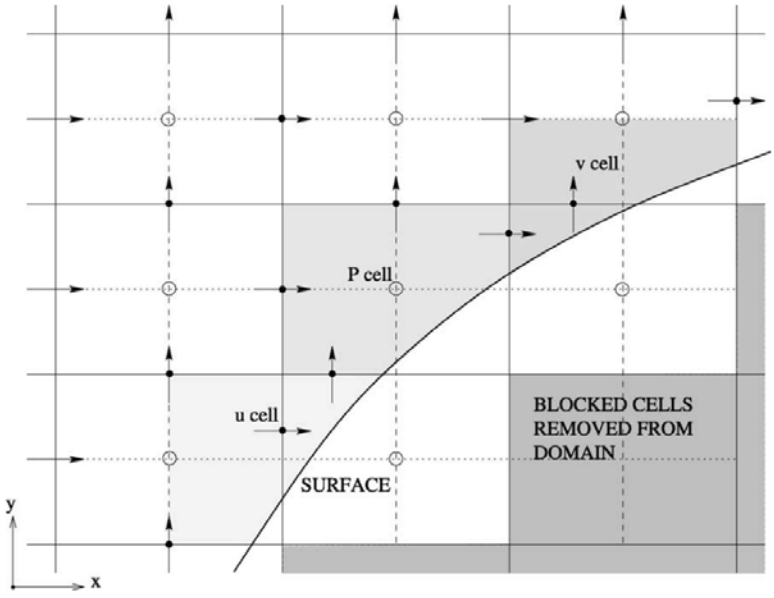
$$\phi = C_1\, x\, y^2 + C_2\, y^2 + C_3\, x\, y + C_4\, x + C_5\, y + C_6,$$

where the unknown coefficients C_1 to C_6 can be determined by evaluating ϕ at the six points shown in Figure 2.3 (□). This stencil guarantees local second-order accuracy and strict conservation of mass and momentum, irrespective of the grid resolution. However, the downside of this cell-merging method is that the discretization stencil in this newly formed cell loses its 5-point structure of Cartesian methods in 2D. The cell-merging technique significantly increases the complexity of coding logistics as fluxes between diagonally adjacent cells need to be also calculated and the computational molecule for merged boundary cells becomes different than the standard cells [58]. Moreover, the formulation of a systematic merging algorithm in three-dimensions is problematic because of the presence of complex polyhedral cells [58].

Kirkpatrik *et al.* [58] proposed a novel "cell-linking" method which mitigates the issues related to the so-called cell-merging method. In this method which is used for representing curved boundaries for the solution of the Navier-Stokes equations on a nonuniform, staggered, three-dimensional Cartesian grid, the two cells are linked as a "master/slave" pair in which the two nodes are coincident while each cell remains a distinct entity, see Figure 2.4(a). Since the slave node and cell exist as entities separate from the the master node and cell, thus the fluxes, wall shear stress, volumetric and surface parameters of master and slave cells are computed exactly in the same manner as those in the standard boundary cells. According to [58], cell-linking is performed based on three criteria. The first criterion is that each velocity cell with only one associated pressure cell becomes a slave cell and is linked with a master cell, see Figure 2.4. The second criterion is that any pressure cell in which the area of its elementary faces are too small (less than 1% of the original cell face area), are merged by discarding the node from the calculation and treating the associated velocity cells as slave cells. The third one expresses that linking the master and slave velocity nodes is achieved when the slave node is placed a small distance ($10^{-8}\times$ cell size) from the master node. For more information about the cell-linking procedure refer to [58]. Linking the cells based on these criteria leads to elimination of the stiffness problem, while both momentum and mass are still conserved throughout the entire computational domain. However, it can not be easily applied to the moving objects due to the presence of extremely small-size cells [94]. Another drawback is that since the distance between the slave and master nodes is too small, the diffusion flux between the two nodes becomes extremely high. Thus, the two velocities take the same value which is undesirable for high Reynolds number flows.



(a)



(b)

Figure 2.4: (a) A master and slave cell are shown for the u component of velocity. The slave cell velocity has only one pressure node associated with it. It is moved to the same position as the master cell node. The figures are reprinted from [58]. (b) Staggered grid with an oblique boundary in two dimensions. Cells are truncated by the boundary. Circles indicate pressure nodes. Horizontal arrows indicate the location of nodes for the u velocity component and vertical arrows the v velocity component. Examples of a u, v, and P cell at the boundary are highlighted.

In summary, it is observed that in these methods, the strict conservation of mass and momentum is fulfilled even in the vicinity of the immersed boundary as they are based on the finite volume method. Furthermore, the application of adaptively refined grids does not complicate the implementation of these techniques and the boundary conditions can be accurately imposed on the body in the cut-cell methods. But implementing the boundary conditions in irregular cells requires special treatments which could result in complex coding process. The occurrence of too small cut-cells could not only lead to stability problems, but could also result in slow convergence of the Poisson solver. Finally, an iterative solution procedure may be required due to the irregular stencil near the immersed boundary. In the following, we will present a Cartesian grid/immersed boundary method, hereinafter called LS-STAG, "Level-Set, STAGgered", developed by Cheny and Botella [22] which combines the best aspects of both classes of immersed boundary methods. The LS-STAG method strictly preserves the conservation of global invariants such as total mass, momentum and kinetic energy in the whole fluid domain. These properties have been successfully devised up to the cut-cells even on the non-uniform Cartesian grids through a precise attention on the terms acting on the immersed boundary in the global conservation equations, both at continuous and discrete levels. Another distinctive feature of LS-STAG is its capability in sharp representation of the immersed boundary by using the level-set function. In addition, unlike the classical immersed boundary methods, the flow variables are actually computed in the cut-cells with no interpolation. Furthermore, the five-point structure of the stencil in 2D is also retained in LS-STAG method which significantly reduces the computational cost in flow simulations. A brief study of this highly computationally efficient method in 2D will be carried out in Section 2.2.

2.2 Background of LS-STAG method for 2D configurations

2.2.1 Principles and description of LS-STAG mesh

In this section, we will overview the 2D version of LS-STAG method. LS-STAG is an immersed boundary method for viscous incompressible flows where the irregular boundary is sharply represented by its level-set function. The level-set function enables us to compute efficiently all relevant geometric parameters of the so-called "cut-cells", *i.e.* the cells that are cut by the immersed boundary, reducing thus the bookkeeping associated to the handling of complex geometries. Let us consider an irregular solid domain Ω^{ib} that is embedded in the computational domain Ω indicated in Figure 2.5. The Navier-Stokes equations are discretized in the fluid domain $\Omega^{\text{f}} = \Omega \backslash \Omega^{\text{ib}}$. The finite-volume discretization of 2D incompressible Navier-Stokes equations in their integral forms are written for the continuity equation as:

$$\int_{\Gamma^{\text{ib}}} \boldsymbol{v} \cdot \boldsymbol{n} \ \mathrm{d}S = 0, \tag{2.1}$$

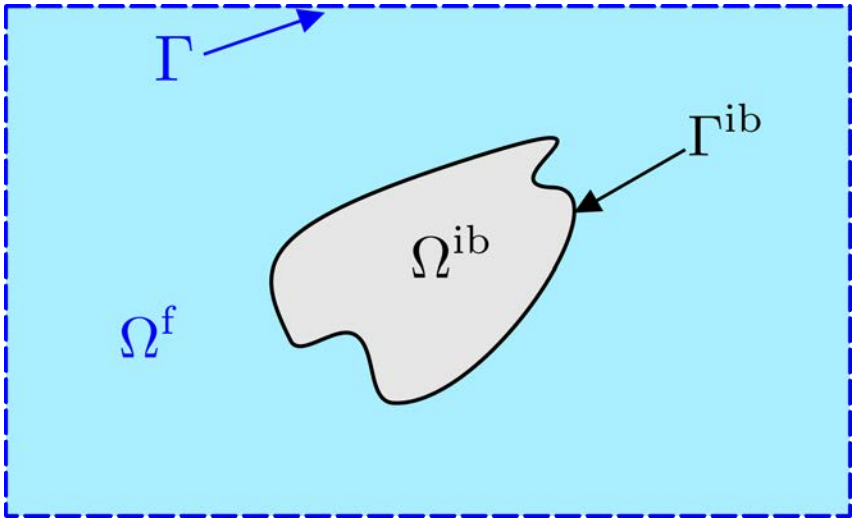


Figure 2.5: Complex geometry Ω^{ib} embedded in a rectangular domain Ω.

and for the momentum equations in x and y directions as:

$$\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega^{\mathrm{f}}} u \ \mathrm{d}V + \rho \int_{\Gamma^{\mathrm{ib}}} (\boldsymbol{v} \cdot \boldsymbol{n})\, u \ \mathrm{d}S + \int_{\Gamma^{\mathrm{ib}}} p\, \boldsymbol{e}_x \cdot \boldsymbol{n} \ \mathrm{d}S - \eta \int_{\Gamma^{\mathrm{ib}}} \boldsymbol{\nabla} u \cdot \boldsymbol{n} \ \mathrm{d}S = 0, \tag{2.2a}$$

$$\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega^{\mathrm{f}}} v \ \mathrm{d}V + \rho \int_{\Gamma^{\mathrm{ib}}} (\boldsymbol{v} \cdot \boldsymbol{n})\, v \ \mathrm{d}S + \int_{\Gamma^{\mathrm{ib}}} p\, \boldsymbol{e}_y \cdot \boldsymbol{n} \ \mathrm{d}S - \eta \int_{\Gamma^{\mathrm{ib}}} \boldsymbol{\nabla} v \cdot \boldsymbol{n} \ \mathrm{d}S = 0, \tag{2.2b}$$

where $\boldsymbol{n}$ is the outwardly directed normal unit vector of surface Γ^{ib}, $\boldsymbol{v} = (u, v)$ is the velocity vector, p is the pressure and η is the dynamic viscosity.

A signed distance function $\phi(\boldsymbol{x})$ (*i.e.* the level-set function [75, 35]) is employed to keep track of the irregular boundary Γ^{ib} as the negative value of $\phi(\boldsymbol{x})$ represents the fluid region Ω^{f}, the solid region Ω^{ib} describes with the positive value of $\phi(\boldsymbol{x})$ and such that the boundary Γ^{ib} corresponds to the zero level-set of this function (refer to [22]), *i.e.* :

$$\phi(\boldsymbol{x}) \equiv \begin{cases} -\Delta, & \boldsymbol{x} \in \Omega^{\mathrm{f}}, \\ 0, & \boldsymbol{x} \in \Gamma^{\mathrm{ib}}, \\ +\Delta, & \boldsymbol{x} \in \Omega^{\mathrm{ib}}, \end{cases} \tag{2.3}$$

where Δ represents the distance between $\boldsymbol{x}$ and the nearest point on the immersed boundary. Employing the aforementioned definition entails some modifications on the MAC mesh and thereby introducing the LS-STAG mesh as shown in Figure 2.6. Three basic types of cut-cells, namely, *trapezoidal* cells such as $\Omega_{i,j}$, *triangular* cells *e.g.* $\Omega_{i-1,j+1}$ and *pentagonal* such as $\Omega_{i-1,j}$, present in 2D version of LS-STAG mesh. In each cut-cell $\Omega_{i,j}$, the immersed boundary is represented by a line segment whose extremities are defined by linear interpolation of the variable $\phi_{i,j}$, which takes the value of level-set function $\phi(x_i, y_j)$ at the upper right corner of the cell. The surface $\Gamma_{i,j}$ of cut-cell $\Omega_{i,j}$ is decomposed to 4 elementary faces by using the customary conventional compass notations (w : West, e : East, s : South, n : North) used in finite volume

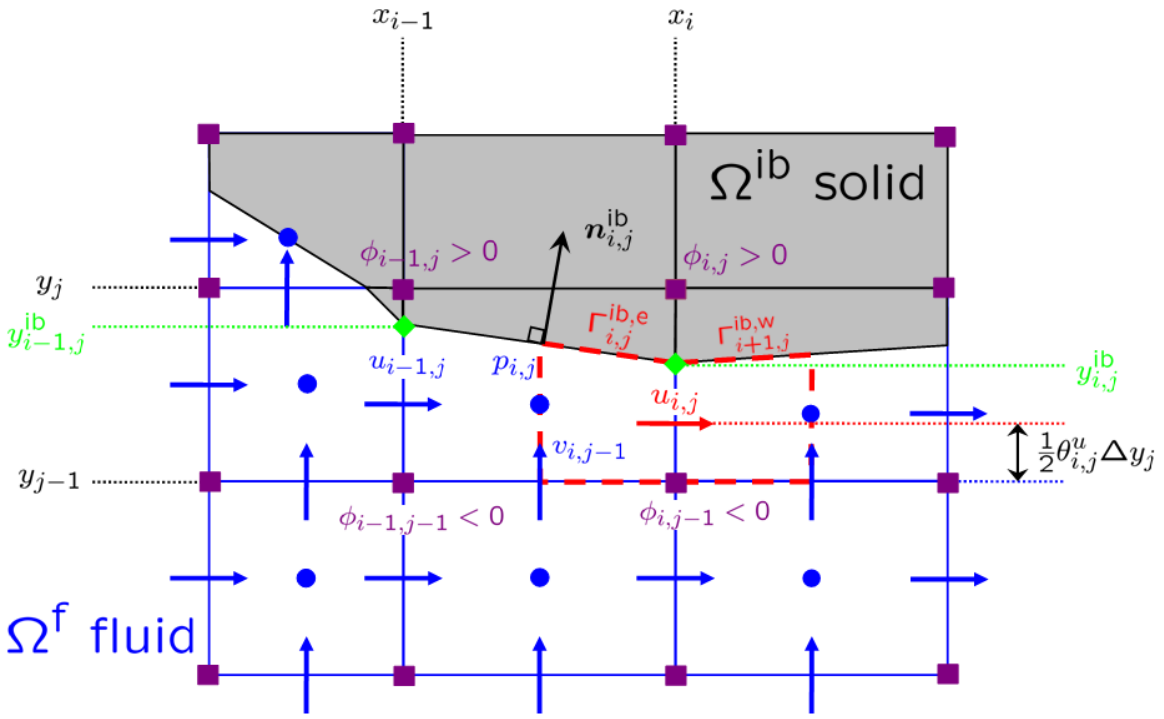


Figure 2.6: Staggered arrangement of the variables near the trapezoidal cut-cell $\Omega_{i,j}$ on the LS-STAG mesh reprinted from [22].

books (*e.g.* [36, 77]). For instance, the faces of trapezoidal cut-cell $\Omega_{i,j}$ in Figure 2.6 are denoted:

$$\Gamma_{i,j} = \Gamma^{w}_{i,j} \cup \Gamma^{e}_{i,j} \cup \Gamma^{s}_{i,j} \cup \Gamma^{ib}_{i,j}, \tag{2.4}$$

where $\Gamma^{ib}_{i,j}$ represents the solid north face of the cut-cell. The velocity unknowns are exactly located in the middle of fluid part of faces. The discrete pressure $p_{i,j}$ is a piecewise constant in each cut-cell and thus there is no need to be precisely located in the cut-cells. Discretization of the pressure gradient is obtained through the duality with divergence operator and not by the using a finite differences formula based on the geometric characteristics of the cells. Note that in 2D LS-STAG mesh, the shear stresses are located at the vertices of the fluid cells (Cartesian or cut-cell), see Figure 2.7, whereas the normal stresses are considered to have a physical origin (diagonal part of the Cauchy stress tensor) and a mathematical regularity as the pressure, thus are assumed to take constant values in fluid cell and do not need to be precisely located. In other words, the normal stresses are discretized with piecewise constant polynomials with a degree of freedom at the cell centroid [88].

The 2D discretization of the momentum equations are performed in the staggered control volumes $\Omega^{u}_{i,j}$ and $\Omega^{v}_{i,j}$, whose shapes must be adapted to each type of cut-cells. For example in Figure 2.6, the control volume $\Omega^{u}_{i,j}$ with the surface of $\Gamma^{u}_{i,j}$, is used as the control volume for the momentum equation in x direction (3.2a). This control volume is centered around $u_{i,j}$ and is formed by halves of neighboring cut-cells $\Omega_{i,j}$ and $\Omega^{u}_{i+1,j}$ such that its volume is $\frac{1}{2}V_{i,j} + \frac{1}{2}V_{i+1,j}$.

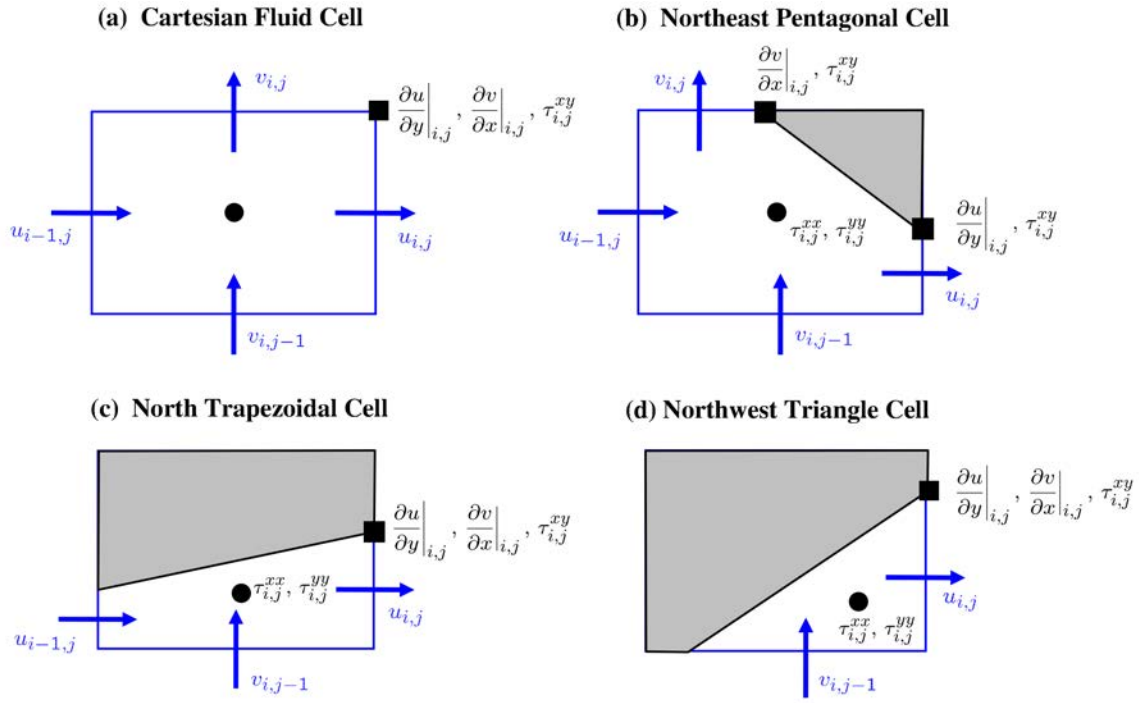


Figure 2.7: Location of stresses in fluid cells (Cartesian and cut-cells) reprinted from [17].

The surface of control volume $\Omega^u_{i,j}$ is decomposed such that:

$$\Gamma^u_{i,j} = \Gamma^{u,\mathrm{w}}_{i,j} \cup \Gamma^{u,\mathrm{e}}_{i,j} \cup \left(\Gamma^{\mathrm{s,e}}_{i,j} \cup \Gamma^{\mathrm{s,w}}_{i+1,j}\right) \cup \left(\Gamma^{\mathrm{ib,e}}_{i,j} \cup \Gamma^{\mathrm{ib,w}}_{i+1,j}\right), \tag{2.5}$$

where the face $\Gamma^{\mathrm{ib,e}}_{i,j} \cup \Gamma^{\mathrm{ib,w}}_{i+1,j}$ is constructed from the halves of the solid faces of two adjacent trapezoidal cut-cells $\Gamma^{\mathrm{ib,e}}_{i,j} \subset \Gamma^{\mathrm{ib}}_{i,j}$ and $\Gamma^{\mathrm{ib,w}}_{i+1,j} \subset \Gamma^{\mathrm{ib}}_{i+1,j}$. For other type of cut-cells, this control volume can be constructed from the combination of six halves of generic control volumes represented in Figure 2.8. Analogous decomposition can be also applied for the control volume $\Omega^v_{i,j}$ which is used in discretization of y momentum equation (2.2b). Note that the irregular shape of staggered control volumes are indicated for illustration purpose only, and their geometric parameters such as their actual volume or shape of the vertical faces $\Gamma^{u,\mathrm{w}}_{i,j}$ and $\Gamma^{u,\mathrm{e}}_{i,j}$ have been never used in the LS-STAG discretization. Instead, the discretization in the cut-cells has been built for the half control volumes, indicated in Figure 2.8, such that any combination yields a consistent discretization of the momentum equations. Consequently, the strict conservation of global invariants of flow [22, 20, 21] is verified. The LS-STAG method relies on a sharp representation of the geometry of cut-cells $\Omega_{i,j}$ based on level-set function, thereby the geometric parameters of a cut-cell, such as its volume or the projected areas of its faces can be efficiently computed. In 2D meshes, an important quantity that used extensively to determine the geometric parameters, is the fluid portion of the faces of cell $\Omega_{i,j}$.

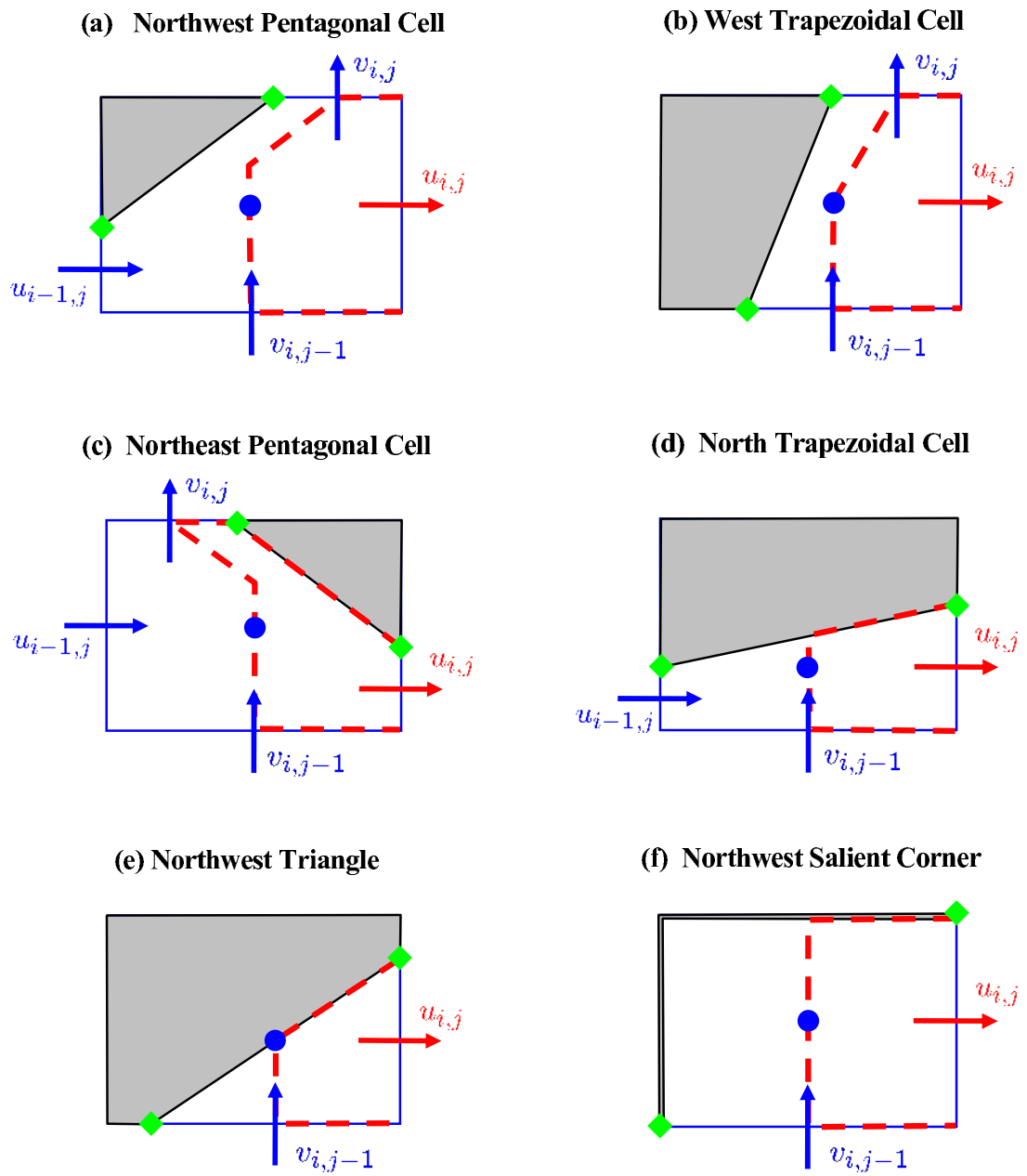


Figure 2.8: Basic types of half $u_{i,j}$ control volumes, $\Omega^u_{i,j}$, inside the cut-cell $\Omega_{i,j}$ reprinted from [22]. The green diamonds (♦) denote the locations of the discretization of the velocity boundary conditions. Note that the re-entrant corner of a Cartesian mesh is a particular case of pentagonal cell (c) when $\phi_{i,j} = 0$, and the particular case of a salient corner (f) in which $\phi_{i-1,j-1} = \phi_{i-1,j} = \phi_{i,j} = 0$ and $\phi_{i,j-1} < 0$) only differs in definition of the cell volume with the limiting case of triangle (e) where $\phi_{i-1,j-1} = \phi_{i,j} = 0$, $\phi_{i-1,j} > 0$ and $\phi_{i,j-1} < 0$.

For instance, the length of fluid portion of face $\Gamma^{\mathrm{e}}_{i,j}$ $(y^{\mathrm{ib}}_{i,j} - y_{j-1})$ in Figure 2.6 is defined by using one-dimensional linear interpolation of $\phi(x_i, y)$ in $[y_{j-1},\, y_j]$ as:

$$y^{\mathrm{ib}}_{i,j} - y_{j-1} = \theta^{u}_{i,j}\, \Delta y_j, \quad \text{with} \quad \theta^{u}_{i,j} = \frac{\phi_{i,j-1}}{\phi_{i,j-1} - \phi_{i,j}},$$

since $\phi(x_i, y^{\mathrm{ib}}_{i,j}) = 0$. The scalar quantities $\theta^{u}_{i,j}$ and $\theta^{v}_{i,j}$, hereinafter *cell-face fraction* ratios, represent the fluid portion of the eastern $\Gamma^{\mathrm{e}}_{i,j}$ and northern $\Gamma^{\mathrm{n}}_{i,j}$ faces respectively. Note that the cell-face fraction ratio takes values in $[0, 1]$. The analytic expression of the volume $V_{i,j}$ of $\Omega_{i,j}$ can be defined for the three basic types of cut-cells through employing the cell-face fraction ratios, see Table 2.1. In this table, $V_{i,j}$ corresponds to the VOF[1] function used for the simulation of multiphasic flows, *e.g.* [89].

Table 2.1: Analytical formulas of the volume for the basic types of cut-cells.

Type of cut-cell	Volume
(c) Northeast Pentagonal Cell	$V_{i,j} = [\theta^{v}_{i,j} + \frac{1}{2}(1 + \theta^{u}_{i,j})(1 - \theta^{v}_{i,j})]\, \Delta x_i\, \Delta y_j$
(d) North Trapezoidal Cell	$V_{i,j} = \frac{1}{2}\left(\theta^{u}_{i,j} + \theta^{u}_{i-1,j}\right) \Delta x_i\, \Delta y_j$
(e) Northwest Triangle cell	$V_{i,j} = \frac{1}{2}\, \theta^{u}_{i,j}\, \theta^{v}_{i,j-1}\, \Delta x_i\, \Delta y_j$
(f) Northwest Salient Corner cell	$V_{i,j} = \Delta x_i\, \Delta y_j$

2.2.2 Global conservation laws for viscous incompressible flows

The interest of numerical methods in physically realistic integration of dynamical systems was recognized from the late 50s. Focusing on the vortex dynamics equation, Arakawa [3] showed that such methods allow us to perform an integration of the system longer in time than traditional methods. Although it is not designed for this purpose, the MAC method of Harlow and Welch [47] is known as the first method for Navier-Stokes equations ((2.1), (2.2)) that preserves the mass, momentum and kinetic energy for incompressible viscous fluid. Since the late 70s, the research team of CTR[2] Stanford [63] have focused on the development of such second and higher-order methods in simulation of turbulence flows by using LES[3] et DNS[4] method [69, 101, 72] for non-uniform meshes.

In the fluid domain Ω^{f}, the flow invariants are the total mass $\int_{\Omega^{\mathrm{f}}} \nabla{\cdot}\boldsymbol{v}\ \mathrm{d}V$ in the whole domain, total momentum $\mathbf{P}(t) = \int_{\Omega^{\mathrm{f}}} \rho \boldsymbol{v}\ \mathrm{d}V$, and in the limiting case of non-viscous fluid (vanishing viscosity), total kinetic energy $\mathrm{E_c}(t) = \frac{1}{2}\int_{\Omega^{\mathrm{f}}} \rho \boldsymbol{v}^2\ \mathrm{d}V$. At discrete level, the conservation of linear invariants of flow is ensured as soon as the applied numerical method satisfies the property of *local conservation of the fluxes*, see [22, 20]. On the other side, by applying the approach of

[1] Volume-Of-Fluid
[2] Center for Turbulence Research
[3] Large Eddy Simulation
[4] Direct Numerical Simulation

Verstappen and Veldman [102], we are able to build a numerical method preserving the quadratic invariants such as kinetic energy. According to [4], the total mass and momentum are the linear invariants of flow whose conservation equations are obtained by a straightforward integration of the Navier-Stokes equations (2.1, 2.2) in the whole fluid domain Ω^{f} such that:

$$\dot{m} = \int_{\Omega^{\mathrm{f}}} \nabla \cdot \boldsymbol{v} \ \mathrm{d}V = \int_{\Gamma^{\mathrm{ib}}} \boldsymbol{v} \cdot \boldsymbol{n} \ \mathrm{d}S = 0, \tag{2.6a}$$

$$\frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} = -\int_{\Gamma^{\mathrm{ib}}} \boldsymbol{v}\, \boldsymbol{v} \cdot \boldsymbol{n} \ \mathrm{d}S - \mathbf{F}, \tag{2.6b}$$

where $\mathbf{F} = (F_x, F_y)$ represents the hydrodynamic force acting on the immersed boundary, such that its components are given by:

$$F_x = -\int_{\Gamma^{\mathrm{ib}}} (-p + \eta \nabla \boldsymbol{v} \cdot \boldsymbol{e}_x) \cdot \boldsymbol{n} \ \mathrm{d}S = \int_{\Gamma^{\mathrm{ib}}} \big[\, p - \eta \frac{\partial u}{\partial x} \,\big]\, \boldsymbol{e}_x \cdot \boldsymbol{n} \ \mathrm{d}S - \int_{\Gamma^{\mathrm{ib}}} \eta \frac{\partial u}{\partial y}\, \boldsymbol{e}_y \cdot \boldsymbol{n} \ \mathrm{d}S, \tag{2.7a}$$

$$F_y = -\int_{\Gamma^{\mathrm{ib}}} (-p + \eta \nabla \boldsymbol{v} \cdot \boldsymbol{e}_y) \cdot \boldsymbol{n} \ \mathrm{d}S = \int_{\Gamma^{\mathrm{ib}}} \big[\, p - \eta \frac{\partial v}{\partial y} \,\big]\, \boldsymbol{e}_y \cdot \boldsymbol{n} \ \mathrm{d}S - \int_{\Gamma^{\mathrm{ib}}} \eta \frac{\partial v}{\partial x}\, \boldsymbol{e}_x \cdot \boldsymbol{n} \ \mathrm{d}S. \tag{2.7b}$$

It is observed that only the surface terms on Γ_{ib} are involved in conservation equations of total mass (2.6a) and momentum (2.6b) as they are obtained from the conservative form of Navier-Stokes equations. Thus, in order that the discrete counterparts of these equations consist only the surface terms, the numerical method must verify the property of local conservation of the fluxes. We will subsequently review the global conservation laws employed to derive the 2D version of LS-STAG method.

Conservation of total mass

In LS-STAG method, the continuity equation describes as the net mass flux through the faces of each fluid cell (Cartesian or cut-cell) such that:

$$\dot{m}_{i,j,k} \equiv -\overline{u}_{i-1,j} + \overline{u}_{i,j} - \overline{v}_{i,j-1} + \overline{v}_{i,j} + \overline{U}^{\mathrm{ib}}_{i,j} = 0, \tag{2.8}$$

where $\overline{U}^{\mathrm{ib}} \equiv \int_{\Gamma^{\mathrm{ib}}_{i,j}} \boldsymbol{v}^{\mathrm{ib}} \cdot \boldsymbol{n}^{\mathrm{ib}} \ \mathrm{d}S$ denotes the mass flux through the solid part of cut-cell. By summing this discrete continuity equation in all fluid cells, we reach to the discrete counterpart of the conservation of total mass (2.6a) whose matrix form reads:

$$\mathbb{1}^{\mathrm{T}} \mathcal{D} U + \mathbb{1}^{\mathrm{T}} \overline{U}^{\mathrm{ib}} = 0, \tag{2.9}$$

where $\mathbb{1} = \begin{pmatrix} \mathbb{1}_x & 0 \\ 0 & \mathbb{1}_y, \end{pmatrix}$ is the constant vector and $\mathcal{D}$ is the divergence operator. In this equation, the vectors U and $\overline{U}^{\mathrm{ib}}$ contain the velocity unknowns $(u_{i,j}, v_{i,j})$ and the discretization of boundary terms, respectively.

In a general case when the immersed boundary has a time-dependent motion of a rigid body, the immersed boundary velocity vector is defined such that:

$$\boldsymbol{v}^{\mathrm{ib}}(\boldsymbol{x}, t) = \boldsymbol{V}^{\mathrm{ib}}(t) + \boldsymbol{\Omega}^{\mathrm{ib}}(t) \times \mathbf{O}\boldsymbol{x}, \tag{2.10}$$

in which $\mathbf{O}$ is a reference point of the immersed solid Ω^{ib}, vectors $\boldsymbol{V}^{\text{ib}}(t)$ and $\boldsymbol{\Omega}^{\text{ib}}(t)$ are the given translation and angular velocities. According to [22], it is assumed that the immersed boundary Γ_{ib} is a closed surface:

$$\int_{\Gamma^{\text{ib}}} \boldsymbol{n}^{\text{ib}} \ \mathrm{d}S = \sum_{\text{Cut-cells}\,\Omega_{i,j}} [\boldsymbol{n}\boldsymbol{\Delta}\boldsymbol{S}]^{\text{ib}}_{i,j} = \mathbf{0}, \tag{2.11}$$

and the fluid domain is not dilatable:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega^{\text{f}}} \ \mathrm{d}V = \sum_{\text{Cut-cells}\,\Omega_{i,j}} \boldsymbol{\Omega}^{\text{ib}}(t) \times \frac{\mathbf{O}\boldsymbol{a}^{\text{ib}}_{i,j} + \mathbf{O}\boldsymbol{b}^{\text{ib}}_{i,j}}{2} \cdot [\boldsymbol{n}\boldsymbol{\Delta}\boldsymbol{S}]^{\text{ib}}_{i,j} = 0, \tag{2.12}$$

where $\boldsymbol{a}^{\text{ib}}_{i,j}$ and $\boldsymbol{b}^{\text{ib}}_{i,j}$ denote the extremities of $\Gamma^{\text{ib}}_{i,j}$.

Considering the local conservativity of the mass fluxes at fluid faces, the LHS[1] of Equation (2.9) is simplified to $\mathbb{1}^{\text{T}}\overline{U}^{\text{ib}}$. Now, by integrating the general velocity datum (Equation (2.10)) with the trapezoidal rule, one can get:

$$\mathbb{1}^{\text{T}}\mathcal{D}U + \mathbb{1}^{\text{T}}\overline{U}^{\text{ib}} = \mathbb{1}^{\text{T}}\overline{U}^{\text{ib}} = \sum_{\text{Cut-cells}\,\Omega_{i,j}} \boldsymbol{V}^{\text{ib}}(t) \cdot [\boldsymbol{n}\boldsymbol{\Delta}\boldsymbol{S}]^{\text{ib}}_{i,j} + \boldsymbol{\Omega}^{\text{ib}}(t) \times \frac{\mathbf{O}\boldsymbol{a}^{\text{ib}}_{i,j} + \mathbf{O}\boldsymbol{b}^{\text{ib}}_{i,j}}{2} \cdot [\boldsymbol{n}\boldsymbol{\Delta}\boldsymbol{S}]^{\text{ib}}_{i,j} = 0, \tag{2.13}$$

thanks to the aforementioned assumptions ((2.11) and (2.12)). Thus, it is clearly observed that the LS-STAG method inherits the global conservation property of mass from the original MAC method [22].

Conservation of total momentum

In LS-STAG method, the semi-discrete matrix representation of Navier-Stokes equations is given by:

$$\rho\,\frac{\mathrm{d}}{\mathrm{d}t}(\mathcal{M}U) + \rho\,\mathcal{C}[\overline{U}]U + \mathcal{G}P - \eta\,\mathcal{K}U + \rho\,S^{\text{ib,c}} - \eta\,S^{\text{ib,v}} = 0, \tag{2.14}$$

where the diagonal mass matrix $\mathcal{M}$ is built from the volume of fluid cells, matrix $\mathcal{C}[\overline{U}]$ represents discretization of the convective fluxes, $\mathcal{G}$ is the discrete pressure gradient, P indicates the pressure, $\mathcal{K}$ stands for the diffusive term, $S^{\text{ib,c}}$ and $S^{\text{ib,v}}$ are source terms arising from the boundary conditions of the convective and viscous terms, respectively. These different terms will now be constructed such that the total momentum and kinetic energy are discretely conserved. The total momentum $\mathbf{P}(t) = \int_{\Omega^{\text{f}}} \boldsymbol{v} \ \mathrm{d}V$ is discretized by using the trapezoidal rule as:

$$\mathbf{P}(t) \cong \mathbf{P}^h(t) = \mathbb{1}^{\text{T}}\mathcal{M}U + \mathbb{1}^{\text{T}}\mathcal{M}^{\text{ib}}U^{\text{ib}}. \tag{2.15}$$

where $\mathbb{1}^{\text{T}}\mathcal{M}^{\text{ib}}U^{\text{ib}}$ represents the contribution of surface terms on Γ^{ib}, which is assumed to be time-independent. The trapezoidal rule provides the diagonal coefficients of the mass matrix in x and y directions for each line (i,j) of the discrete system (2.14) such that:

$$[\mathcal{M}^x]_{\text{P}}\,(i,j) = \frac{1}{2}V_{i,j} + \frac{1}{2}V_{i+1,j}, \tag{2.16a}$$

$$[\mathcal{M}^y]_{\text{P}}\,(i,j) = \frac{1}{2}V_{i,j} + \frac{1}{2}V_{i,j+1}, \tag{2.16b}$$

[1] Left-Hand Side

where the subscript P refers to the main diagonal elements. The above expressions indicate that the mass matrices for $u_{i,j}$ and $v_{i,j}$ components are built regardless of the shape of control volumes $\Omega^u_{i,j}$ and $\Omega^v_{i,j}$. The semi-discrete counterpart of momentum equations (2.6b) is obtained by multiplying Equation (2.14) with vector $\mathbb{1}$ as:

$$\frac{\mathrm{d}\mathbf{P}^h}{\mathrm{d}t} = -\left[\rho\,\mathbb{1}^{\mathrm{T}}\mathcal{C}[\overline{U}]U + \rho\,\mathbb{1}^{\mathrm{T}}S^{\mathrm{ib,c}}\right] - \left[\mathbb{1}^{\mathrm{T}}\mathcal{G}P - \mathbb{1}^{\mathrm{T}}\eta\left(\mathcal{K}U + S^{\mathrm{ib,v}}\right)\right]. \tag{2.17}$$

In this expression, the quadratic terms in the RHS[1] correspond to summation of the convective, pressure and viscous fluxes from all control volumes. The fluxes through the fluid faces are canceled out owing to the property of local conservation of the fluxes. The remaining terms appear only at the solid boundary faces of the cut-cells. Thus, the conservation of discrete momentum is ensured by considering these terms as the forces acting on the immersed boundary Γ^{ib} such that:

- the remaining terms in sum $\rho\,\mathbb{1}^{\mathrm{T}}\,\mathcal{C}[\overline{U}]\,U + \rho\,\mathbb{1}^{\mathrm{T}}S^{\mathrm{ib,c}}$ correspond to the discretization of inertial forces $\int_{\Gamma^{\mathrm{ib}}} \boldsymbol{v}\,\boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S$ acting on the boundary, and analogously,
- the remaining terms in sum $\mathbb{1}^{\mathrm{T}}\,\mathcal{G}P - \mathbb{1}^{\mathrm{T}}\,\eta(\mathcal{K}U + S^{\mathrm{ib,v}})$ correspond to the discretization of hydrodynamic forces.

Conservation of total kinetic energy

The conservation of kinetic energy $\mathrm{E_c} = \int_{\Omega^{\mathrm{f}}} \rho\boldsymbol{v}^2/2\ \mathrm{d}V$ is obtained by algebraic manipulation of the Navier-Stokes equations. The scalar multiplication of momentum equation with $\boldsymbol{v}$ results in:

$$\rho\left(\frac{\partial}{\partial t}\left(\frac{\boldsymbol{v}^2}{2}\right) + \boldsymbol{v}\cdot(\boldsymbol{v}\cdot\nabla)\boldsymbol{v}\right) = \boldsymbol{v}\cdot(-\nabla p + \eta\nabla\cdot\nabla\boldsymbol{v}). \tag{2.18}$$

Thereafter, by considering the tensor identities:

$$\boldsymbol{v}\cdot(\boldsymbol{v}\cdot\nabla)\boldsymbol{v} = \boldsymbol{v}\cdot\nabla\left(\frac{\boldsymbol{v}^2}{2}\right), \tag{2.19a}$$

$$\boldsymbol{v}\cdot(\nabla\cdot\nabla\boldsymbol{v}) = \nabla\cdot(\nabla\boldsymbol{v}\cdot\boldsymbol{v}) - \nabla\boldsymbol{v} : \boldsymbol{v}, \tag{2.19b}$$

$$\boldsymbol{v}\cdot\nabla p = \nabla\cdot(p\,\boldsymbol{v}) - p\nabla\cdot\boldsymbol{v}, \tag{2.19c}$$

we can rewrite the Equation (2.18) such that:

$$\frac{\partial}{\partial t}\left(\frac{\rho\boldsymbol{v}^2}{2}\right) = (-\boldsymbol{v}\cdot\nabla)\left(\frac{\boldsymbol{v}^2}{2} + p\right) + \nabla\cdot(\nabla\boldsymbol{v}\cdot\boldsymbol{v}) - \eta\nabla\boldsymbol{v} : \nabla\boldsymbol{v}. \tag{2.20}$$

Now, by applying the tensor identity:

$$\nabla\cdot\left[\left(\frac{\rho\boldsymbol{v}^2}{2} + p\right)\boldsymbol{v}\right] = \boldsymbol{v}\cdot\nabla\left(\frac{\rho\boldsymbol{v}^2}{2} + p\right) + \left(\frac{\rho\boldsymbol{v}^2}{2} + p\right)\nabla\cdot\boldsymbol{v} \tag{2.21}$$

[1] Right-Hand Side

we can determine a local expression for conservation of kinetic energy such that:

$$\frac{\partial}{\partial t}\left(\frac{\rho \boldsymbol{v}^2}{2}\right) = \left(\frac{\rho \boldsymbol{v}^2}{2} + p\right)\nabla\cdot\boldsymbol{v} - \nabla\cdot\left[\left(\frac{\rho \boldsymbol{v}^2}{2} + p\right)\boldsymbol{v}\right] + \nabla\cdot(\eta\nabla\boldsymbol{v}\cdot\boldsymbol{v}) - \eta\nabla\boldsymbol{v} : \nabla\boldsymbol{v}. \tag{2.22}$$

After integrating above expression on the whole fluid domain Ω^{f} and applying the flux divergence theorem, the global conservation of kinetic energy $\mathrm{E_c}$ reads:

$$\begin{aligned}\frac{\mathrm{dE_c}}{\mathrm{d}t} =& -\int_{\Gamma^{\mathrm{ib}}}\left(\frac{\rho \boldsymbol{v}^2}{2}\right)\boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S \\ &-\int_{\Gamma^{\mathrm{ib}}} p\,\boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S + \int_{\Gamma^{\mathrm{ib}}}(\eta\nabla\boldsymbol{v}\cdot\boldsymbol{v})\cdot\boldsymbol{n}\ \mathrm{d}S \\ &-\int_{\Omega^{\mathrm{f}}}\left(\frac{\rho \boldsymbol{v}^2}{2}\right)\nabla\cdot\boldsymbol{v}\ \mathrm{d}V + \int_{\Omega^{\mathrm{f}}} p\nabla\cdot\boldsymbol{v}\ \mathrm{d}V \\ &-\int_{\Omega^{\mathrm{f}}}\eta\nabla\boldsymbol{v} : \nabla\boldsymbol{v}\ \mathrm{d}V.\end{aligned} \tag{2.23}$$

Now, thanks to the continuity equation (2.6a), this conservation is simplified to:

$$\begin{aligned}\frac{\mathrm{dE_c}}{\mathrm{d}t} =& -\int_{\Gamma^{\mathrm{ib}}}\left(\frac{\rho \boldsymbol{v}^2}{2}\right)\boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S \\ &-\int_{\Gamma^{\mathrm{ib}}} p\,\boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S + \int_{\Gamma^{\mathrm{ib}}}(\eta\nabla\boldsymbol{v}\cdot\boldsymbol{v})\cdot\boldsymbol{n}\ \mathrm{d}S \\ &-\int_{\Omega^{\mathrm{f}}}\eta\nabla\boldsymbol{v} : \nabla\boldsymbol{v}\ \mathrm{d}V,\end{aligned} \tag{2.24}$$

In this expression, the first term corresponds to the global flux of kinetic energy per unit volume $\rho\boldsymbol{v}^2/2$ transported by the fluid through the surface Γ_{ib}, the second and third terms represent the work of pressure and viscous forces exerted normally on Γ_{ib}, and the last term expresses the loss of energy by viscous dissipation per unit volume that is always negative. Note that the terms in surface integral affect only the kinetic energy budget by their action at the immersed boundary. In original MAC method, the discrete kinetic energy budget mimics Equation (2.24) only on the uniform Cartesian grids. However, for more general type of meshes the construction of *energy preserving* methods is not a trivial task. Verstappen and Veldman [102] have constructed an energy preserving finite-volume method on non-uniform grids in which the convective and viscous terms are discretized with skew-symmetric and symmetric positive-definite operators, respectively.

In discrete level, the kinetic energy budget $\mathrm{E_c}(t)$ in the fluid cell $\Omega_{i,j}$ is written in its matrix form as:

$$\mathrm{E_c}(t) \cong \mathrm{E_c^h}(t) = \rho\left(\frac{1}{2}U^{\mathrm{T}}\mathcal{M}U + \frac{1}{2}U^{\mathrm{ib,T}}\mathcal{M}^{\mathrm{ib}}U^{\mathrm{ib}}\right), \tag{2.25}$$

where the time-independent term $U^{\mathrm{ib,T}}\mathcal{M}^{\mathrm{ib}}U^{\mathrm{ib}}$ represents the contribution of surface terms on Γ^{ib}. After time-differentiation of above expression and by employing Equation (2.14), the discrete form of conservation of kinetic energy budget is given such that:

$$\frac{\mathrm{dE_c^h}}{\mathrm{d}t} = -\rho\,U^{\mathrm{T}}\,\frac{\mathcal{C}[\overline{U}]^{\mathrm{T}} + \mathcal{C}[\overline{U}]}{2}\,U - P^{\mathrm{T}}\mathcal{G}^{\mathrm{T}}U - U^{\mathrm{T}}\,\frac{\eta(\mathcal{K}^{\mathrm{T}} + \mathcal{K})}{2}\,U - \rho\,U^{\mathrm{T}}\,S^{\mathrm{ib,c}} + \eta\,U^{\mathrm{T}}\,S^{\mathrm{ib,v}}. \tag{2.26}$$

The term $-U^{\mathrm{T}}\left(\mathcal{K}^{\mathrm{T}}+\mathcal{K}\right)U$ corresponds to the viscous dissipation of kinetic energy budget in the whole fluid domain and is strictly negative [22]. This property is verified if the matrix $\mathcal{K}^{\mathrm{T}}+\mathcal{K}$ is always positive definite. According to Equation (2.24), this discrete form is ensured to be energy preserving if the pressure gradient is dual to the divergence operator:

$$\mathcal{G} = -\mathcal{D}^{\mathrm{T}}, \tag{2.27}$$

and the convective terms is a skew-symmetric matrix:

$$\mathcal{C}[\overline{U}] = -\mathcal{C}[\overline{U}]^{\mathrm{T}}. \tag{2.28}$$

2.3 Advantage of LS-STAG over unstructured grid-based methods in terms of memory efficiency

Before we compare the memory efficiency of LS-STAG and unstructured grid-based methods, it is noteworthy to consider the evaluation of geometric parameters such as face area and volume of a cell in an unstructured grid which is necessary to build the discretization scheme. Figure 2.9 indicates a 2D quadrilateral cell C_0 with volume V_0 surrounded by 4 cells C_1 to C_4. This quadrilateral cell consists of 4 vertices, A to D, with the coordinates in basis $(\vec{i},\vec{j})$ such that $\vec{X}_{\mathrm{A}} = x_{\mathrm{A}}\,\vec{i}+y_{\mathrm{A}}\,\vec{j}$. In this figure, $\vec{S}_{01}$ represents the face vector between A and C, and $\vec{r}_{01}$ describes the distance between the centroid of C_0 ($\vec{X}_{01}$) and the center of $\vec{S}_{01}$. The outward-pointing face vector $\vec{S}_{01}$ is given by:

$$\vec{S}_{01} = \begin{pmatrix} S^x_{01} \\ S^y_{01} \end{pmatrix} = \begin{pmatrix} y_{\mathrm{C}} - y_{\mathrm{B}} \\ x_{\mathrm{B}} - x_{\mathrm{C}} \end{pmatrix}, \tag{2.29}$$

where S^x_{01} and S^y_{01} denote the Cartesian component of this face vector. Subsequently, the area of S_{01} is computed such that:

$$\mid \vec{S}_{01} \mid = \sqrt{(S^x_{01})^2 + (S^y_{01})^2}. \tag{2.30}$$

Similar expressions can be also extracted for the face vectors $\vec{S}_{02}$, $\vec{S}_{03}$ and $\vec{S}_{04}$. Now, the volume of C_0 can be exactly calculated by using the Gauss' formula as:

$$V_0 = \frac{1}{2}\sum_{i=1}^{4} \vec{X}_{0i}\cdot\vec{S}_{0i} = \frac{1}{2}\left[\Delta x_{\mathrm{AC}}\,\Delta y_{\mathrm{BD}} - \Delta x_{\mathrm{BD}}\,\Delta y_{\mathrm{AC}}\right], \tag{2.31}$$

where $\Delta x_{\mathrm{AC}} = x_{\mathrm{C}} - x_{\mathrm{A}}$ and $\Delta y_{\mathrm{AC}} = y_{\mathrm{C}} - y_{\mathrm{A}}$ and *etc.*. Since computing these geometric parameters in unstructured grids are used in discretization, one requires to store those quantities which entails a significant increase in computer storage requirements. For instance, for a quadrilateral cell in the two-dimensional case, it is necessary to store 17 arrays including one for the volume of cell, V_0, 8 arrays for the area vector components (*e.g.* S^x_{01} and S^y_{01}) and 8 arrays for the vectors describing the distance between the centroid of cell and center of each face (*e.g.* r^x_{01} or r^y_{01}). For three-dimensional cases, for instance, 28 arrays have to be stored for a hexahedral element. The

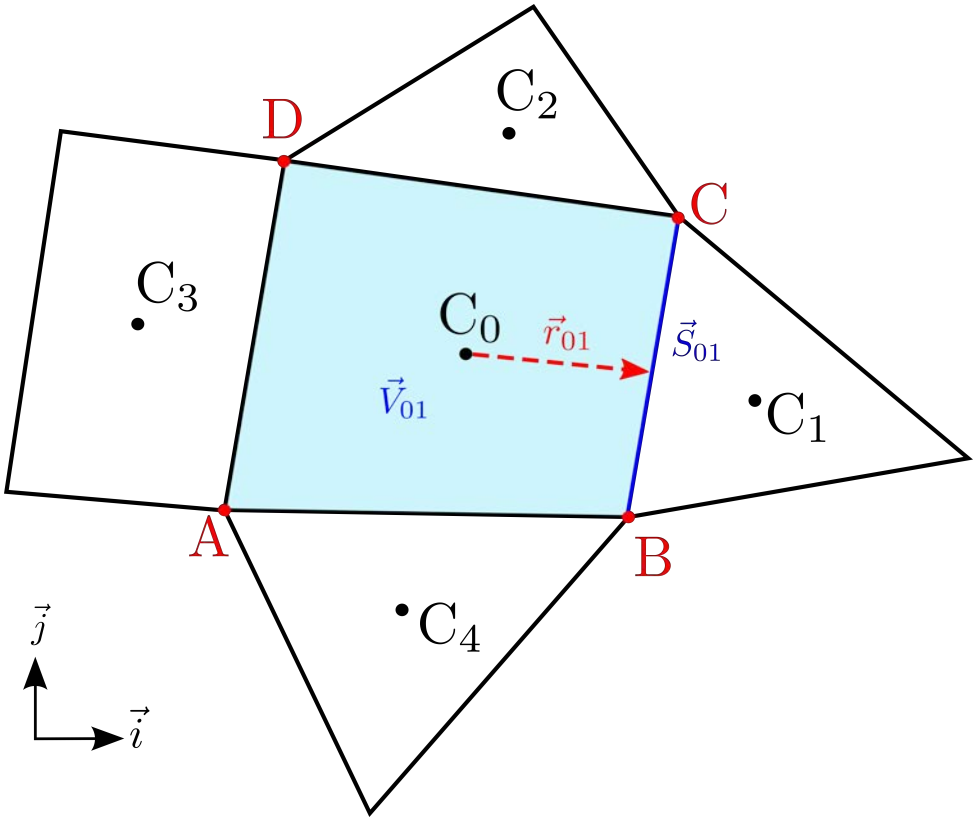


Figure 2.9: Example of unstructured mixed (triangular, quadrilateral) mesh in 2D.

high computational cost and memory requirements especially in considering flows with complex geometries are accounted as a redundant disadvantage in unstructured grid methods. In contrast, the LS-STAG method takes into account irregular domains on staggered Cartesian grids at low computational costs based on level-set techniques. For example, we only store 3 arrays in two-dimensional cases; one array for level-set variable which stored at the vertices of control volume and two arrays for the cell-face fraction ratios. For three-dimensional cases, we totally require to store 4 arrays including the level-set (1 array) and cell-face fraction ratio (3 arrays in all coordinate directions). The edge-fraction ratio which represents the fluid portion of the edges (required only in 3D) and the volumes are computed on the fly.

Although robustness and accuracy of numerical methods are important factors, the computational resources should also be considered in choosing a numerical method. These resources include the computation time and memory space require for storing and retrieving data. Building a method that mitigates the essential resources to a feasible extent without reducing its consistency and accuracy, is by no means a trivial task. The comparison of memory capacity required to store the arrays of geometric parameters, shows the superiority of LS-STAG compared to the methods using unstructured grids. However, a detailed comparison between LS-STAG method and an unstructured solver (FLUENT©) will be carried out in terms of computational resources and accuracy in Chapter 4. This comparison reveals that LS-STAG is a promising

and efficient method in both aspects and can be evaluated as a desirable method that enable us to simulate a wide range of flows with less computational overhead.

2.4 Conclusion

In this chapter, we have first carried out an overview of two general classes of immersed boundary methods *i.e.* continuous and discrete forcing approaches. The main advantages of discrete forcing approach over the continuous one are its ability to sharply represent the immersed boundary, elimination of the stiffness of the system of equations for the rigid bodies and no need of user-defined parameters which may affect the stability. Among the methods which are categorized in discrete forcing approach, only the cut-cell methods can satisfy the underlying conservation laws. However, the simplicity of the 5-point structure of stencil (in 2D) is lost near the immersed boundary in most of these methods. To circumvent this issue, an innovative immersed boundary/cut-cell method (LS-STAG) based on the staggered MAC arrangement have been introduced [22]. In LS-STAG method which is designed for computations of rheologically complex fluid flows, the geometry is sharply represented by using the level-set function. In this method the conservation of global invariants such as total mass, momentum and kinetic energy are strictly preserved in the whole fluid domain both at discrete and continuous levels. The conservation of linear invariants of flow, *i.e.* total mass and momentum, is ensured by verifying the property of local conservation of the fluxes. For the quadratic invariants such as kinetic energy budget the conservation is achieved by applying the symmetry preserving discretization of Verstappen and Veldman [102] results in a method of second-order accuracy in both space and time. In 2D geometries, the LS-STAG method have been successfully applied to Newtonian flows at moderate Reynolds number in fixed and moving geometries [22], pseudoplastic flows [15], and viscoelastic flows [16, 18]. In addition, we mention that the LS-STAG discretization has also been implemented in the ship hydrodynamics software ComFLOW [99] for the handling of irregular geometries, and has been extended to the computation of coupled aero-elastic problems [80] and turbulent flows with RANS equations modeling [81].

Chapter 3

The LS-STAG discretization for 3D extruded geometries

Contents

3.1 Introduction 43

3.2 Preliminaries and principles of LS-STAG method in 3D extruded meshes 48

3.2.1 Governing equations for non-Newtonian flows 48

3.2.2 LS-STAG mesh in 3D extruded geometries 48

3.3 Discretization of the continuity equation 52

3.4 Discretization of the momentum equation 55

3.4.1 Discretization of the rate of change of the momentum 58

3.4.2 Discretization of the pressure gradient 58

3.4.3 Skew-symmetric discretization of the convective fluxes 58

3.4.4 Discretization of velocity gradients on LS-STAG mesh 62

3.4.5 Discretization of viscous fluxes 71

3.5 Extension of LS-STAG method to non-Newtonian fluids 76

3.5.1 Interpolation of the shear-rates at the centroids 76

3.5.2 Interpolation of the shear-rates on edges 78

3.6 LS-STAG time stepping method and solution of linear systems . . . 78

3.7 Conclusion 80

3.1 Introduction

The development of an accurate and cost-effective cut-cell solver, optimized for high performance computing (HPC) in 3D geometries is a challenging task. One of the main reasons is the large number of 3D cut-cells that needs to be considered. In the 2D version of the LS-STAG method

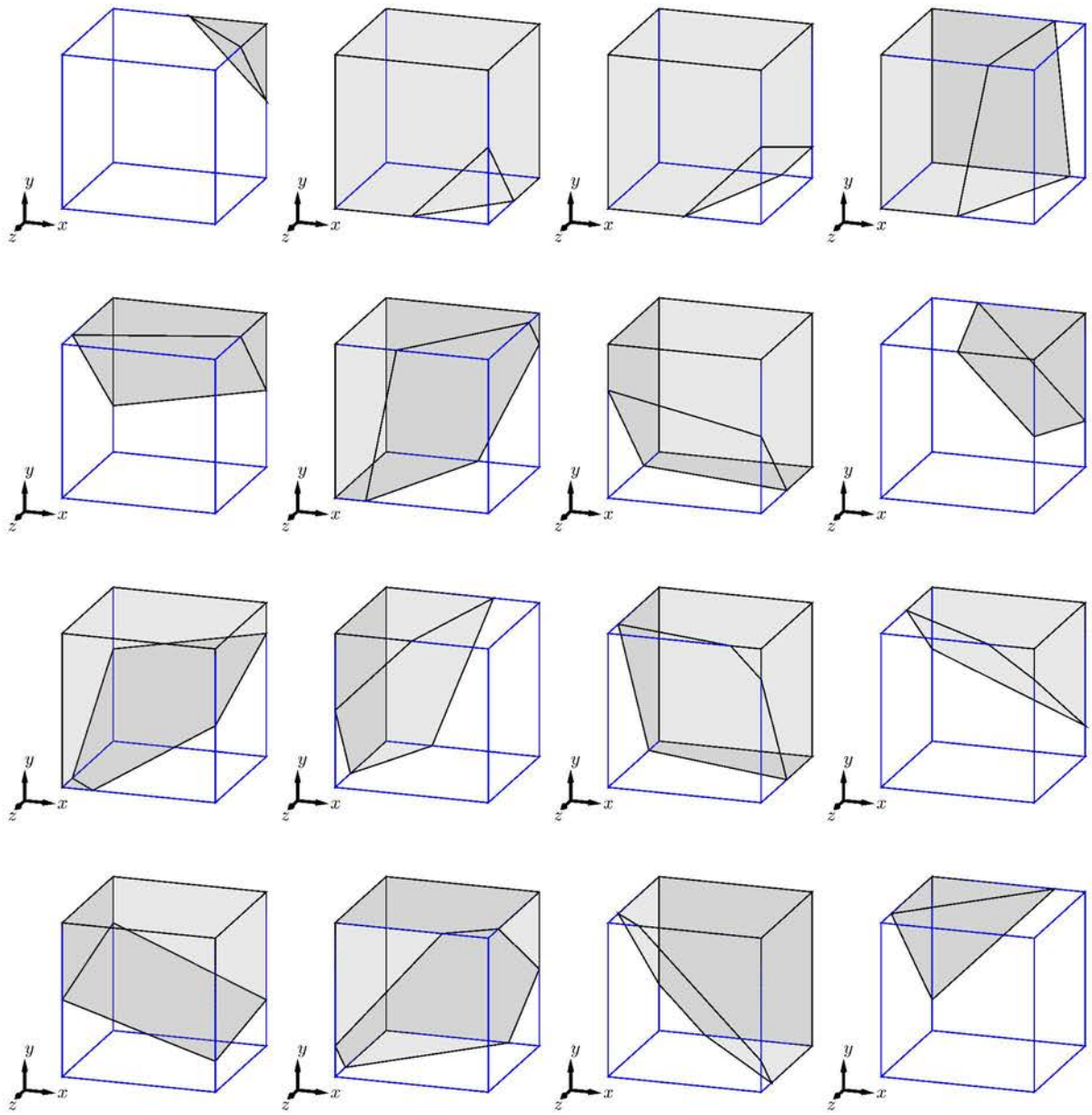


Figure 3.1: 16 types of cells existed in full 3D geometries.

Section 2.2.1, there is exactly 12 distinct types of cut-cells, which are obtained by rotation of the 3 generic cut-cells (pentagonal, trapezoidal and triangular cells) shown in Figure 2.7. The discretization was constructed for these 3 generic types only, and then adapted to each of the 12 distinct types and implemented in the Fortran 90 solver. In 3D however, Cheny [20] enumerated the 16 generic types of cut-cells shown in Figure 3.1 which generate 108 distinct types of cut-cells.

Due to the large implementation efforts demanded by the development, optimization and validation of a novel 3D parallel solver, it is not practical to directly take into account the full

three-dimensional cases. Hence, we consider in this thesis the development of the LS-STAG method for the special case of 3D configurations with translational symmetry in the z direction, hereinafter called *3D extruded* configurations. This intermediate step towards the fully 3D implementation can be applied to a wide variety of canonical flows (flow in Taylor-Couette reactors, flow past cylinder of arbitrary cross section shape, abrupt axisymmetric expansion/contraction flow, ... see Figure 3.2) and will be regarded as the milestone for the full 3D solver, since both discretization and implementation issues on distributed memory machines are tackled at this stage of development.

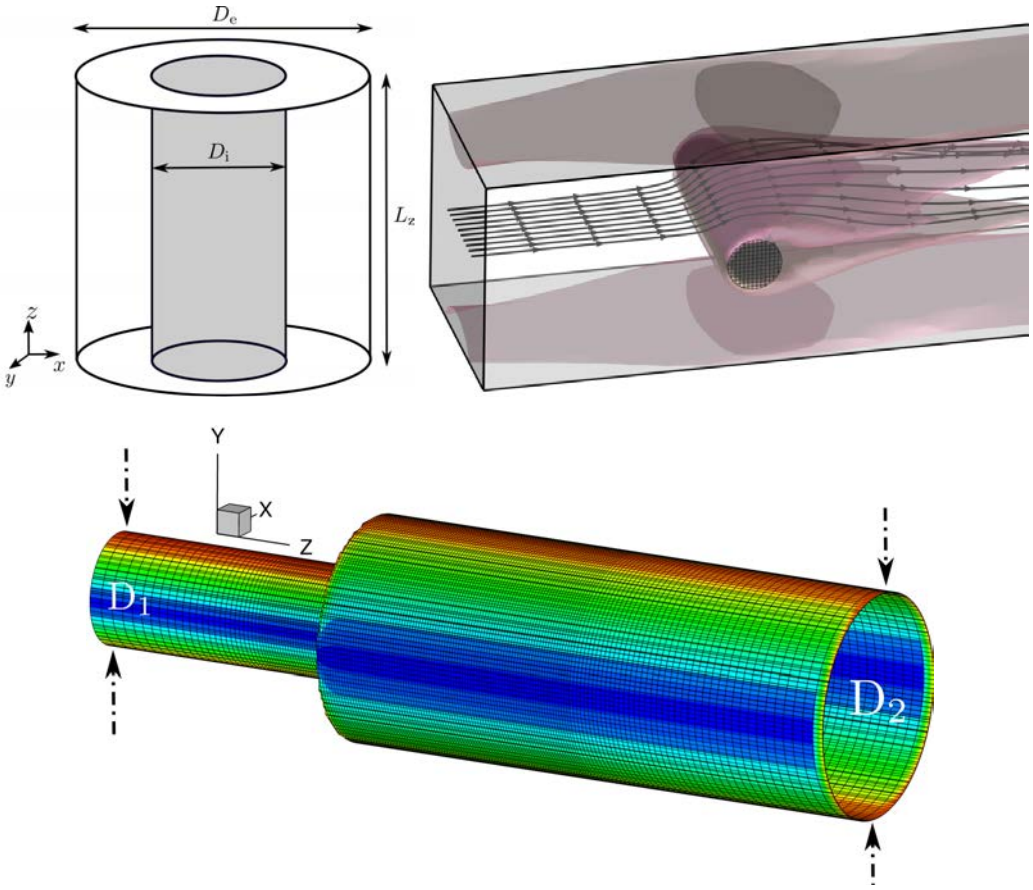


Figure 3.2: Examples of flows in 3D geometries extruded in the z direction..

In these geometries, 3D cut-cells are only extruded extensions of pentagonal, trapezoidal and triangular cut-cells in 2D (see Figure 3.3), which enables us to extend the principles of the energy-conserving discretization discussed in Section 2.2. Figure 3.4 shows the 14 distinct cells which are present in the LS-STAG mesh for 3D extruded geometries. For ease of implementation in a computer code, each cell is flagged by an integer defined by:

$$flag = \sum_{i=0}^{7} 2^i \times a_i, \qquad a_i = \begin{cases} 1, & \text{for fluid vertex} \\ 0, & \text{for solid vertex.} \end{cases}$$

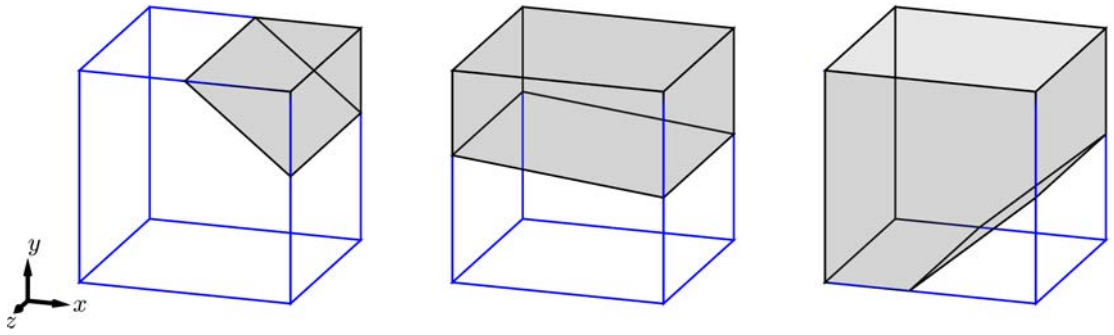


Figure 3.3: 3 basic types of cut-cells in 3D extruded geometries.

In this chapter, we aim to extend the principles of energy-conserving discretization of 2D LS-STAG to the 3D extruded configurations. As discussed in the previous chapter, it will be built based on the global conservation of the flow invariants both at continuous and discrete levels, refer to Section 2.2.2. In addition, the discretization is performed such that the original MAC method is recovered in Cartesian fluid cells, and has the ability to preserve the 7-point Cartesian structure of the stencil. In the following, we will first describe the LS-STAG mesh in 3D extruded geometries and determine the geometric parameters for each type of cut-cells by using the level-set function [35]. Subsequently, the continuity equation will be discretized based on the conservation of total mass (2.2.2) which leads to a unique formula valid for all cell types. The different terms existing in momentum equations *i.e.* pressure gradient, convective and diffusive terms, will be discretized in the staggered control volumes around each velocity component. The pressure gradient is treated as the dual of the divergence operator (2.27) yielding a unique expression valid for all cell types. For the convective terms, the discretization will be constructed in the half control volumes such as the skew-symmetric condition (2.28) is verified for any combinations of these half control volumes.

The discretization of the velocity gradients, which plays an important role of in the numerical modeling of the viscous flows considered in this thesis, is the most involved part of the discretization. The discretization of the diagonal terms which corresponds to the normal diffusive fluxes and are located at the centroid of each cell, is performed by employing the divergence (or Green) theorem. For the off-diagonal elements corresponding to shear fluxes, which are located the midpoints of the edges, various procedures have to be applied depending on the location of the immersed boundary. The terms $\partial u/\partial y$ and $\partial v/\partial x$ are discretized in the same manner as done for their counterparts in 2D geometries, refer to [22]. The discretization of $\partial u/\partial z$ and $\partial v/\partial z$ are carried out as in the standard MAC method thanks to the extrusion in z direction. However, the major issue concerns the discretization of $\partial w/\partial x$ and $\partial w/\partial y$ which require further attention as there is no counterpart in 2D LS-STAG meshes. In this thesis, these terms will be tentatively discretized by the *ad-hoc* formula proposed by Van der Heiden *et al.* [99]. Based on the discretization of the velocity gradients, the computation of the shear-rate

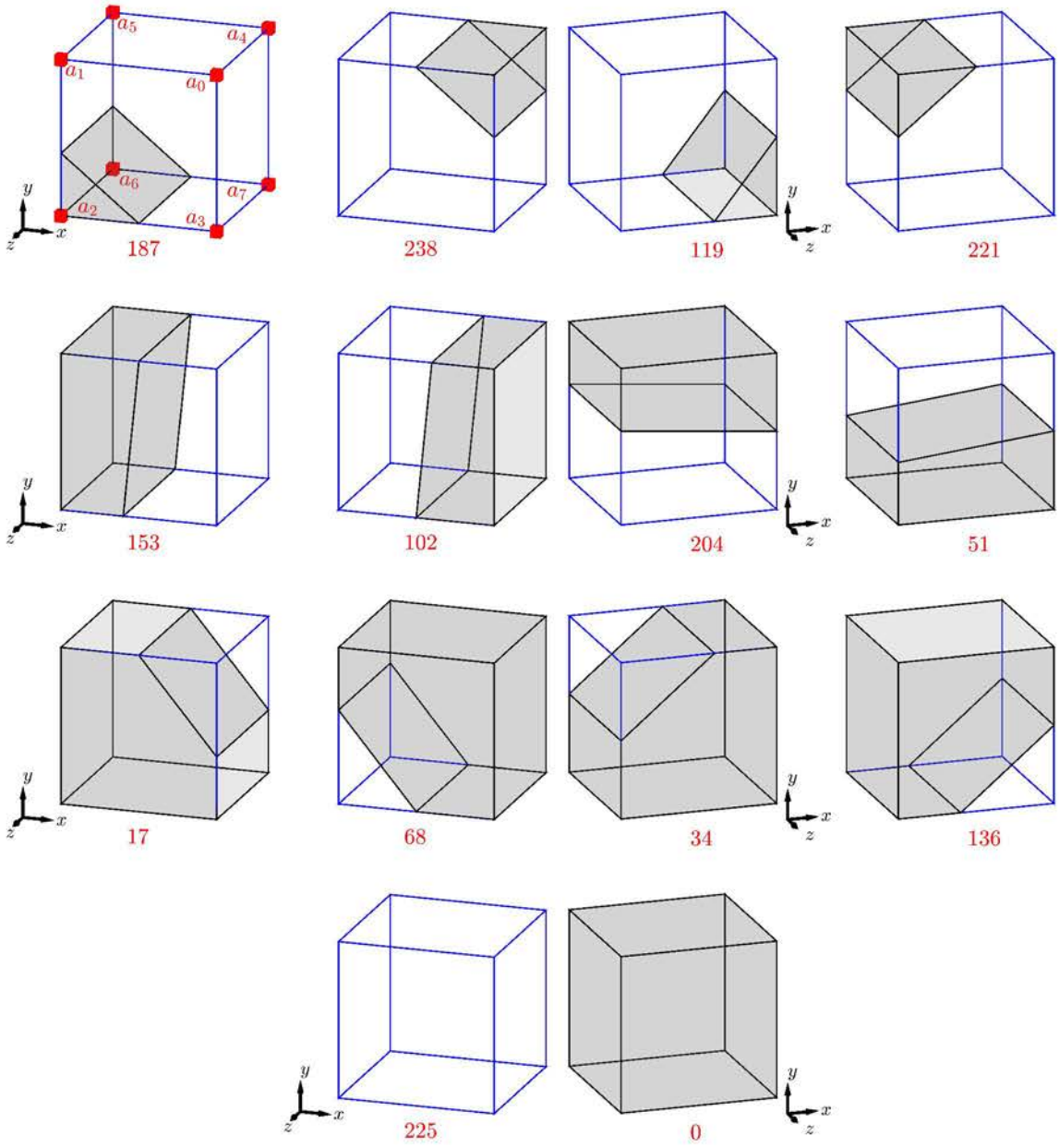


Figure 3.4: 14 distinct cells (12 cut-cells, a Cartesian cell and a solid cell) present in 3D extruded geometries. Each cell is identified with its flag in red.

will be presented for non-Newtonian flows. Finally, the time integration of governing equations and solving the linear systems will be explained.

3.2 Preliminaries and principles of LS-STAG method in 3D extruded meshes

3.2.1 Governing equations for non-Newtonian flows

We consider the incompressible equations for non-Newtonian fluids where the stress is given by (1.8), which are represented by the continuity equation integrated in control volume Ω (of surface Γ):

$$\int_\Gamma \boldsymbol{v}\cdot\boldsymbol{n}\ \mathrm{d}S = 0, \tag{3.1}$$

and the momentum equations (1.1) written for each velocity component:

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega u\ \mathrm{d}V + \rho\int_\Gamma (\boldsymbol{v}\cdot\boldsymbol{n})\,u\ \mathrm{d}S + \int_\Gamma p\,\boldsymbol{e}_x\cdot\boldsymbol{n}\ \mathrm{d}S - \int_\Gamma \eta(\dot\gamma)\,(\dot{\boldsymbol{\gamma}}\cdot\boldsymbol{e}_x)\cdot\boldsymbol{n}\ \mathrm{d}S = 0, \tag{3.2a}$$

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega v\ \mathrm{d}V + \rho\int_\Gamma (\boldsymbol{v}\cdot\boldsymbol{n})\,v\ \mathrm{d}S + \int_\Gamma p\,\boldsymbol{e}_y\cdot\boldsymbol{n}\ \mathrm{d}S - \int_\Gamma \eta(\dot\gamma)\,(\dot{\boldsymbol{\gamma}}\cdot\boldsymbol{e}_y)\cdot\boldsymbol{n}\ \mathrm{d}S = 0, \tag{3.2b}$$

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega w\ \mathrm{d}V + \rho\int_\Gamma (\boldsymbol{v}\cdot\boldsymbol{n})\,w\ \mathrm{d}S + \int_\Gamma p\,\boldsymbol{e}_z\cdot\boldsymbol{n}\ \mathrm{d}S - \int_\Gamma \eta(\dot\gamma)\,(\dot{\boldsymbol{\gamma}}\cdot\boldsymbol{e}_z)\cdot\boldsymbol{n}\ \mathrm{d}S = 0, \tag{3.2c}$$

where $\dot{\boldsymbol{\gamma}} = (\nabla\boldsymbol{v} + \nabla\boldsymbol{v}^{\mathrm{T}})$ is the rate-of-strain tensor.

3.2.2 LS-STAG mesh in 3D extruded geometries

We consider the computational domain is divided into the background Cartesian cells:

$$\Omega_{i,j,k} =]x_{i-1}, x_i[\times]y_{j-1}, y_j[\times]z_{k-1}, z_k[,$$

with the volume of $\Delta x_i \times \Delta y_j \times \Delta z_k$. The coordinates of a Cartesian cell and the compass notations used for its elementary faces are represented in Figure 3.5. The immersed boundary Γ^{ib} may cut these cells and generate the cut-cells. In each cut-cells, the immersed boundary is represented by a rectangular plane whose extremities are defined by linear interpolation of the level-set variable $\phi_{i,j,k}$. It takes the value of level-set function $\phi(x_i, y_j, z_k)$ at the vertices of rectangular cell. The 4 basic types of cells (Cartesian, pentagonal, trapezoidal and triangular) in 3D extruded geometries are represented in Figure 3.6(c). Similar notations to the 2D case (Section 2.2.1) are used in 3D. For instance, the surface of pentagonal cut-cell $\Omega_{i,j,k}$ indicated in Figure 3.6(c) is subdivided into 7 elementary faces that are denoted with the usual compass notations:

$$\Gamma_{i,j,k} = \Gamma^{\mathrm{w}}_{i,j,k} \cup \Gamma^{\mathrm{e}}_{i,j,k} \cup \Gamma^{\mathrm{s}}_{i,j,k} \cup \Gamma^{\mathrm{n}}_{i,j,k} \cup \Gamma^{\mathrm{t}}_{i,j,k} \cup \Gamma^{\mathrm{b}}_{i,j,k} \cup \Gamma^{\mathrm{ib}}_{i,j,k}, \tag{3.3}$$

where $\Gamma^{\mathrm{ib}}_{i,j,k}$ represents the solid face of the cut-cell $\Omega_{i,j,k}$.

In a 3D cell, the velocities unknowns are located at the barycenter of faces, the discrete pressure $p_{i,j,k}$ and normal velocity gradients (*i.e.* $\partial u/\partial x$, $\partial v/\partial y$ and $\partial w/\partial z$) are positioned at the centroids and the tangential velocity gradients (*e.g.* $\partial u/\partial y$, $\partial u/\partial z$) are located at the edge centers, see Figure 3.6. As mentioned earlier in Section 2.2, the pressure and normal velocity

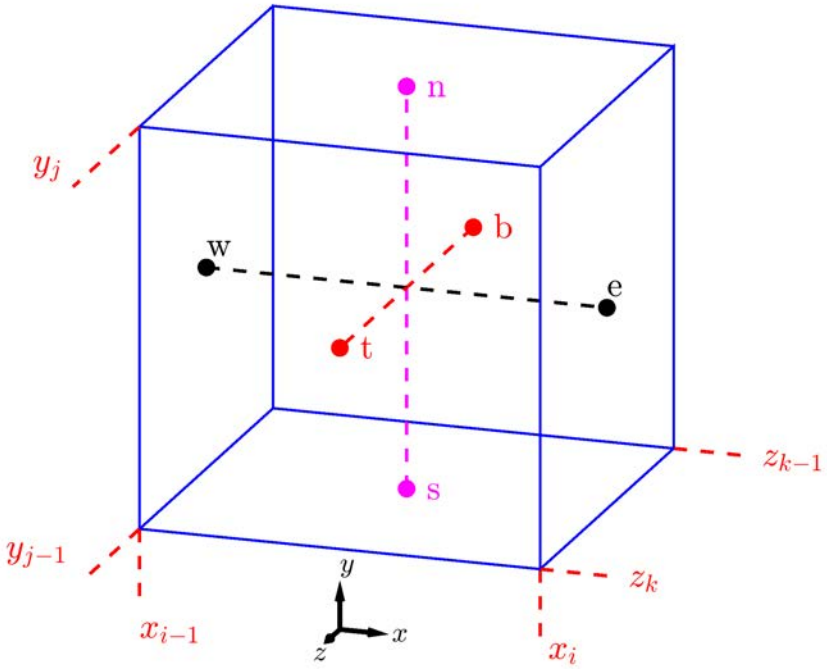


Figure 3.5: The elementary faces of a Cartesian cell denoted with the usual compass notations (w: West, e: East, s: South, n: North, b: Bottom and t: Top).

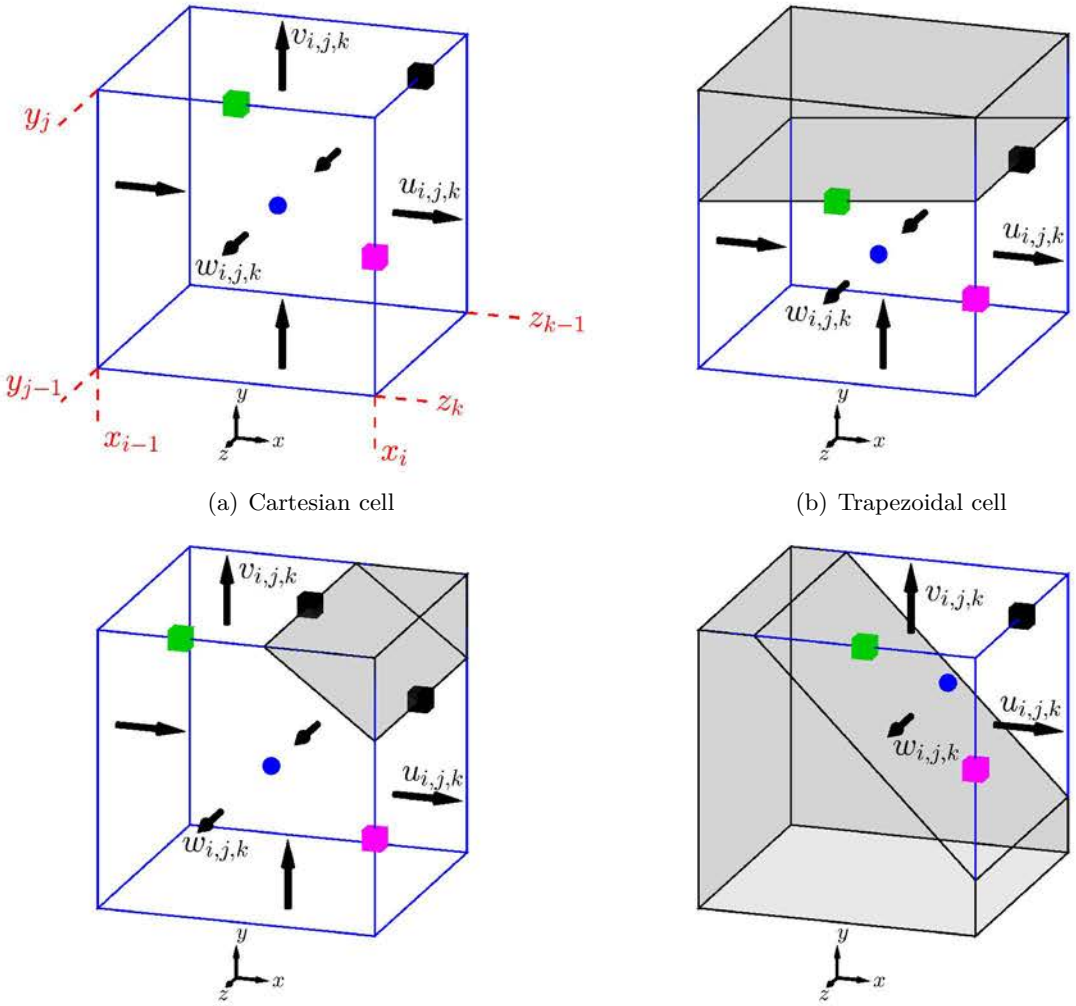
gradients are piecewise constants whose locations are not required to be precisely defined inside the cells.

In [20], the level-set function was employed to obtain analytical expressions of the geometry parameters of 2D cut-cells, such as their face areas and volume. These definitions are now extended to the 3D case, by calculating first the fluid portion of the cut-cell edges. This scalar quantity is denoted with α and will be subsequently called *cell-edge fraction* ratio. For the pentagonal cut-cell shown in Figure 3.7, the length of edge $\mathcal{E}^y_{i,j,k} = \{(x_i, y, z_k),\, y \in [y_{j-1}, y^{\text{ib}}]\}$ is given by $y^{\text{ib}} - y_{j-1} = \alpha^y_{i,j,k}\, \Delta y_j$, where the cell-edge fraction ratio is computed by applying one-dimensional linear interpolation of $\phi(x_i, y, z_k)$:

$$\alpha^y_{i,j,k} = \frac{\phi_{i,j-1,k}}{\phi_{i,j-1,k} - \phi_{i,j,k}}, \tag{3.4}$$

since $\phi(x_i, y^{\text{ib}}, z_k) = 0$. This quantity is equivalent to $\theta^u_{i,j}$ in 2D meshes (Section 2.2) and possesses values between 0 to 1. Similar formulas can be derived for cell-edge fraction ratios $\alpha^x_{i,j,k}$ and $\alpha^z_{i,j,k}$ in the other spatial directions. These ratios correspond to the fluid portion of edges $\mathcal{E}^x_{i,j,k}$ and $\mathcal{E}^z_{i,j,k}$, respectively, see Figure 3.7. Note that due to the extrusion in z direction, $\alpha^x_{i,j,k} = \alpha^x_{i,j,k-1}$, $\alpha^y_{i,j,k} = \alpha^y_{i,j,k-1}$ and $\alpha^z_{i,j,k}$ is always equal to 0 or 1.

A straightforward application of this scalar quantity is in the computation of its surface equivalent, namely the *cell-face fraction* ratio denoted with θ. For example, for the pentagonal cut-cell in Figure 3.7, the surface area of the top face $\Gamma^{\text{t}}_{i,j,k}$ (shaded in green) is given



(a) Cartesian cell (b) Trapezoidal cell

(c) Pentagonal cell (d) Triangular cell

Figure 3.6: Cartesian cell and 3 basic types of cut-cells in 3D extruded configurations. The unknown velocities are discretized at faces centers, pressure and normal velocity gradients (*i.e.* $\partial u/\partial x$, $\partial v/\partial y$ and $\partial w/\partial z$) at cell centroid (●), tangential velocity gradients at edges centers: $\partial u/\partial y$, $\partial v/\partial x$ (■) and $\partial w/\partial x$, $\partial u/\partial z$ (■) and $\partial w/\partial y$, $\partial v/\partial z$ (■).

by $\theta^w_{i,j,k}\,\Delta x_i\,\Delta y_j$ where the scalar quantity $\theta^w_{i,j,k}$ expresses the fluid portion of this face. By employing the neighboring α values, the analytical value of $\theta^w_{i,j,k}$ is given by the formula:

$$\begin{aligned}\theta^w_{i,j,k} &= \min(\alpha^x_{i,j,k}, \alpha^x_{i,j-1,k}) + \frac{1}{2}(\alpha^y_{i-1,j,k} + \alpha^y_{i,j,k})\left|\alpha^x_{i,j,k} - \alpha^x_{i,j-1,k}\right| \\ &= \alpha^x_{i,j,k} + \frac{1}{2}(1+\alpha^y_{i,j,k})(1-\alpha^x_{i,j,k}) = 1 - \frac{1}{2}\left(1-\alpha^x_{i,j,k}\right)(1-\alpha^y_{i,j,k}),\end{aligned} \tag{3.5}$$

which is valid for any type of cut-cells. Similarly, the cell-face fraction ratios of eastern ($\Gamma^e_{i,j,k}$)

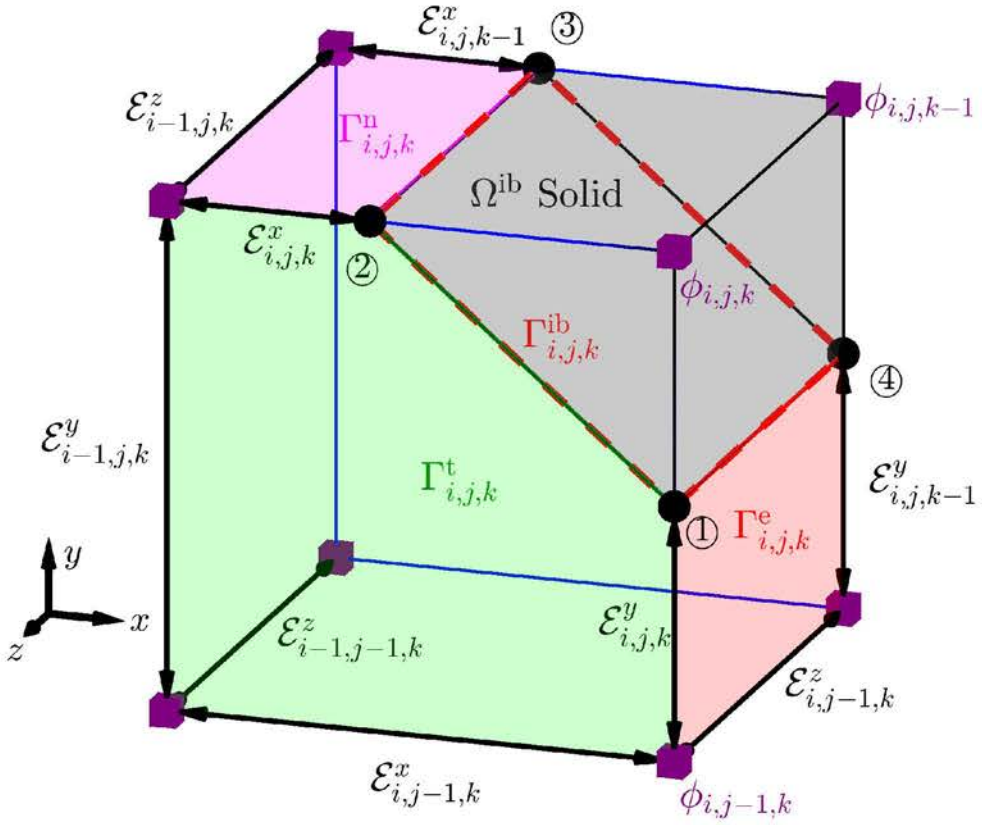


Figure 3.7: Geometric parameters of pentagonal cut-cell $\Omega_{i,j,k}$ in which the immersed boundary $\Gamma^{\mathrm{ib}}_{i,j,k}$ is outlined with red dashed line (--). The extremities of $\Gamma^{\mathrm{ib}}_{i,j,k}$, *i.e.* $(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k)$, $(x^{\mathrm{ib}}_{i,j,k}, y_j, z_k)$, $(x^{\mathrm{ib}}_{i,j,k}, y_j, z_{k-1})$ and $(x_i, y^{\mathrm{ib}}_{i,j,k}, z_{k-1})$ are denoted with ①, ②, ③ and ④, respectively. The level-set function values ϕ (■) which stored at the vertices of cell, are used to compute the cell-edge fraction ratios of edges $\mathcal{E}^x$, $\mathcal{E}^y$ and $\mathcal{E}^z$ in x, y and z directions, respectively. The edge $\mathcal{E}^y_{i,j,k}$ endpoints are $(x_i, y^{\mathrm{ib}}, z_k)$ and (x_i, y_{j-1}, z_k). For the solid edge of $\mathcal{E}^z_{i,j,k}$, the cell-edge fraction ratio is $\alpha^z_{i,j,k} = 0$. The eastern, northern and top elementary faces, namely $\Gamma^{\mathrm{e}}_{i,j,k}$, $\Gamma^{\mathrm{n}}_{i,j,k}$ and $\Gamma^{\mathrm{t}}_{i,j,k}$, are highlighted in red, magenta and green, respectively.

and northern ($\Gamma^{\mathrm{n}}_{i,j,k}$) faces are respectively:

$$\theta^u_{i,j,k} = \min(\alpha^y_{i,j,k}, \alpha^y_{i,j,k-1}) + \frac{1}{2}(\alpha^z_{i,j,k} + \alpha^z_{i,j-1,k}) \left|\alpha^y_{i,j,k} - \alpha^y_{i,j,k-1}\right| = \alpha^y_{i,j,k}, \tag{3.6a}$$

$$\theta^v_{i,j,k} = \min(\alpha^x_{i,j,k}, \alpha^x_{i,j,k-1}) + \frac{1}{2}(\alpha^z_{i,j,k} + \alpha^z_{i-1,j,k}) \left|\alpha^x_{i,j,k} - \alpha^x_{i,j,k-1}\right| = \alpha^x_{i,j,k}. \tag{3.6b}$$

Due to the extrusion in the z direction, the cell-face fraction ratio takes a simpler expression as reported in Table 3.1, since the faces normal to the x and y directions have a rectangular shape, and in the z direction the face area $\theta^w_{i,j,k}\ \Delta x_i\, \Delta y_j$ correspond to the volume of the 2D cut-cells given in Table 2.1.

The last geometry parameter to compute is the volume of each cell $V_{i,j,k}$. Thanks to the extrusion in z direction, each cell is a right prism whose base is $\Gamma^{\mathrm{t}}_{i,j,k}$, thus the volume $V_{i,j,k}$ of

Table 3.1: The cell-face fraction ratios for the cut-cells shown in Figure 3.6.

Cell type	$\theta^u_{i,j,k}$	$\theta^v_{i,j,k}$	$\theta^w_{i,j,k}$
Trapezoidal cell (3.6(b))	$\alpha^y_{i,j,k}$	0	$\frac{1}{2}\left(\alpha^y_{i-1,j,k}+\alpha^y_{i,j,k}\right)$
Pentagonal cell (3.6(c))	$\alpha^y_{i,j,k}$	$\alpha^x_{i,j,k}$	$\alpha^x_{i,j,k}+\frac{1}{2}\left(1+\alpha^y_{i,j,k}\right)(1-\alpha^x_{i,j,k})$
Triangular cell (3.6(d))	$\alpha^y_{i,j,k}$	$\alpha^x_{i,j,k}$	$\frac{1}{2}\,\alpha^y_{i,j,k}\,\alpha^x_{i,j,k}$

every cell is equal to the surface of $\Gamma^{\mathrm{t}}_{i,j,k}$ multiplied by Δz_k:

$$V_{i,j,k} = \theta^w_{i,j,k}\,\Delta x_i\,\Delta y_j\,\Delta z_k. \tag{3.7}$$

All these geometry parameters (cell-edge fraction ratios α, cell-face fraction ratios θ, cell volume V) will be used for discretization of flow equations.

3.3 Discretization of the continuity equation

The starting point of LS-STAG discretization concerns the continuity equation (2.1) in cell $\Omega_{i,j,k}$. For any fluid cell (cut-cell or Cartesian), we decompose the continuity equation as the net mass flux through its elementary faces (Equation (3.3)):

$$\dot{m}_{i,j,k} \equiv \int_{\Gamma_{i,j,k}} \boldsymbol{v}\cdot\boldsymbol{n}\ \ \mathrm{d}S \equiv -\overline{u}_{i-1,j,k}+\overline{u}_{i,j,k}-\overline{v}_{i,j-1,k}+\overline{v}_{i,j,k}-\overline{w}_{i,j,k-1}+\overline{w}_{i,j,k}+\overline{U}^{\mathrm{ib}}_{i,j,k}=0, \tag{3.8}$$

where $\overline{U}^{\mathrm{ib}}_{i,j,k} \equiv \int_{\Gamma^{\mathrm{ib}}_{i,j,k}} \boldsymbol{v}^{\mathrm{ib}}\cdot\boldsymbol{n}^{\mathrm{ib}}_{i,j,k}\ \ \mathrm{d}S$ is the mass flux through the solid part of the cut-cell boundary. The mass flux through the fluid part of each face is denoted with a bar. For instance, $\overline{u}_{i,j,k}$, $\overline{v}_{i,j,k}$ and $\overline{w}_{i,j,k}$ represent the mass fluxes through the eastern $\Gamma^{\mathrm{e}}_{i,j,k}$, northern $\Gamma^{\mathrm{e}}_{i,j,k}$ and top $\Gamma^{\mathrm{t}}_{i,j,k}$ faces, respectively. They are computed such that:

$$\overline{u}_{i,j,k} \equiv \int_{\Gamma^{\mathrm{e}}_{i,j,k}} \boldsymbol{v}\cdot\boldsymbol{e}_x\ \ \mathrm{d}S, \tag{3.9a}$$

$$\overline{v}_{i,j,k} \equiv \int_{\Gamma^{\mathrm{n}}_{i,j,k}} \boldsymbol{v}\cdot\boldsymbol{e}_y\ \ \mathrm{d}S, \tag{3.9b}$$

$$\overline{w}_{i,j,k} \equiv \int_{\Gamma^{\mathrm{t}}_{i,j,k}} \boldsymbol{v}\cdot\boldsymbol{e}_z\ \ \mathrm{d}S. \tag{3.9c}$$

As mentioned earlier, the velocity unknowns are located at the centroids of the cut-cells:

$$u_{i,j,k} \equiv u(x_i, y_{j-1}+\tfrac{1}{2}\,\theta^u_{i,j,k}\,\Delta y_j, z_{k-1}+\tfrac{1}{2}\Delta z_k), \tag{3.10a}$$

$$v_{i,j,k} \equiv v(x_{i-1}+\tfrac{1}{2}\,\theta^v_{i,j,k}\,\Delta x_i, y_j, z_{k-1}+\tfrac{1}{2}\Delta z_k), \tag{3.10b}$$

$$w_{i,j,k} \equiv u(x^{\mathrm{c}}_{i,j,k}, y^{\mathrm{c}}_{i,j,k}, z_k), \tag{3.10c}$$

where $(x^{\mathrm{c}}_{i,j,k}, y^{\mathrm{c}}_{i,j,k})$ is the centroid coordinate of face $\Gamma^{\mathrm{t}}_{i,j,k}$. Due to extrusion in z direction, $\Gamma^{\mathrm{e}}_{i,j,k}$ and $\Gamma^{\mathrm{n}}_{i,j,k}$ are rectangular faces and so their barycenters are located exactly in the middle

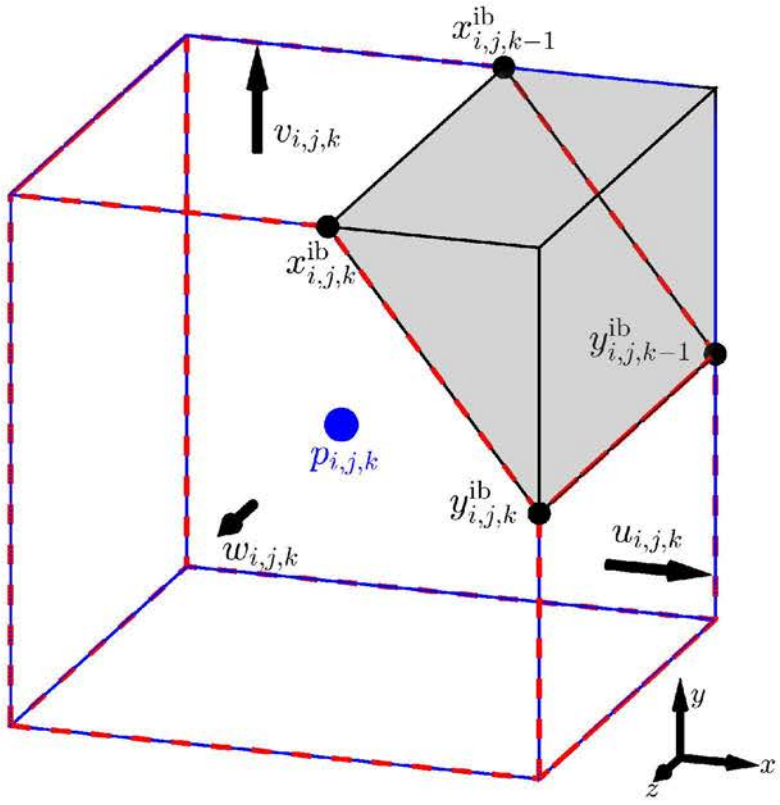


Figure 3.8: Control volume $\Omega_{i,j,k}$ for a pentagonal cut-cell.

of faces. However, for $\Gamma^{\mathrm{t}}_{i,j,k}$, the centroid coordinates is determined by employing the generic formula of centroid of a polygon with $n = 4$ vertices as:

$$x^{\mathrm{c}}_{i,j,k} = \frac{1}{6\,\theta^{w}_{i,j,k}\,\Delta x_i \Delta y_j} \sum_{m=0}^{n} (x_m + x_{m+1})(x_m\, y_{m+1} - x_{m+1}\, y_m), \tag{3.11a}$$

$$y^{\mathrm{c}}_{i,j,k} = \frac{1}{6\,\theta^{w}_{i,j,k}\,\Delta x_i \Delta y_j} \sum_{m=0}^{n} (y_m + y_{m+1})(x_m\, y_{m+1} - x_{m+1}\, y_m). \tag{3.11b}$$

In these formulas, the polygon vertices with coordinates (x_m, y_m) are numbered in order of their occurrence along the polygon's perimeter.

By application of the midpoint quadrature, the mass fluxes (3.9) are approximated as:

$$\overline{u}_{i,j,k} \cong \theta^{u}_{i,j,k}\ \Delta y_j\, \Delta z_k\, u_{i,j,k}, \tag{3.12a}$$

$$\overline{v}_{i,j,k} \cong \theta^{v}_{i,j,k}\ \Delta x_i\, \Delta z_k\, v_{i,j,k}, \tag{3.12b}$$

$$\overline{w}_{i,j,k} \cong \theta^{w}_{i,j,k}\ \Delta x_i\, \Delta y_j\, w_{i,j,k}, \tag{3.12c}$$

and after similar discretization on the remaining faces, the discrete continuity equation reads:

$$\begin{aligned}\dot{m}_{i,j,k} \equiv & \Delta y_j \, \Delta z_k \big(\theta^u_{i,j,k} \, u_{i,j,k} - \theta^u_{i-1,j,k} \, u_{i-1,j,k} \big) \\ & + \Delta x_i \, \Delta z_k \big(\theta^v_{i,j,k} \, v_{i,j,k} - \theta^v_{i,j-1,k} \, v_{i,j-1,k} \big) \\ & + \Delta x_i \, \Delta y_j \big(\theta^w_{i,j,k} \, w_{i,j,k} - \theta^w_{i,j,k-1} \, w_{i,j,k-1} \big) + \overline{U}^{\mathrm{ib}}_{i,j,k} = 0.\end{aligned} \tag{3.13}$$

The last term in this equation is the mass flux through the immersed boundary $\overline{U}^{\mathrm{ib}}_{i,j,k}$ which is computed as:

$$\overline{U}^{\mathrm{ib}}_{i,j,k} \cong u^{\mathrm{ib}}_{i,j,k} \, [n_x \Delta S]^{\mathrm{ib}}_{i,j,k} + v^{\mathrm{ib}}_{i,j,k} \, [n_y \Delta S]^{\mathrm{ib}}_{i,j,k} + w^{\mathrm{ib}}_{i,j,k} \, [n_z \Delta S]^{\mathrm{ib}}_{i,j,k}, \tag{3.14}$$

where $[n_x \Delta S]^{\mathrm{ib}}_{i,j,k}$, $[n_y \Delta S]^{\mathrm{ib}}_{i,j,k}$ and $[n_z \Delta S]^{\mathrm{ib}}_{i,j,k}$ are respectively the projected areas of the solid face of cut-cells along x, y and z axes, and $\boldsymbol{v}^{\mathrm{ib}}_{i,j,k} = (u^{\mathrm{ib}}_{i,j,k}, v^{\mathrm{ib}}_{i,j,k}, w^{\mathrm{ib}}_{i,j,k})$ is an approximation of the velocity at the solid boundary. This last term can be computed with the trapezoidal rule from the velocity boundary conditions $\boldsymbol{v}^{\mathrm{ib}}(x, y, z)$ discretized at the extremities of solid surface segment $\Gamma^{\mathrm{ib}}_{i,j,k}$. For example, the velocity at the immersed boundary of the cut-cell shown in Figure 3.8 is computed as:

$$\boldsymbol{v}^{\mathrm{ib}}_{i,j,k} = \frac{1}{4}\Big[\boldsymbol{v}(x^{\mathrm{ib}}_{i,j,k}, y_j, z_{k-1}) + \boldsymbol{v}(x_i, y^{\mathrm{ib}}_{i,j,k}, z_{k-1}) + \boldsymbol{v}(x^{\mathrm{ib}}_{i,j,k}, y_j, z_k) + \boldsymbol{v}(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k) \Big].$$

The projected areas are readily calculated by using the cell-face fraction ratios as:

$$\begin{aligned}[n_x \Delta S]^{\mathrm{ib}}_{i,j,k} &= (\theta^u_{i-1,j,k} - \theta^u_{i,j,k}) \Delta y_j \, \Delta z_k, \\ [n_y \Delta S]^{\mathrm{ib}}_{i,j,k} &= (\theta^v_{i,j-1,k} - \theta^v_{i,j,k}) \Delta x_i \, \Delta z_k, \\ [n_z \Delta S]^{\mathrm{ib}}_{i,j,k} &= (\theta^w_{i,j,k-1} - \theta^w_{i,j,k}) \Delta x_i \, \Delta y_j.\end{aligned} \tag{3.15}$$

Note that the formulas are valid for any type of 3D cut-cells and, for 3D extruded cut-cells, $[n_z \Delta S]^{\mathrm{ib}}_{i,j,k}$ is always equal to zero owing to extrusion in the z direction ($\theta^w_{i,j,k-1} = \theta^w_{i,j,k}$).

The discrete continuity equation in its matrix form can be written as:

$$\mathcal{D} \, U + \overline{U}^{\mathrm{ib}} = 0, \tag{3.16}$$

where $\mathcal{D}$ is the divergence operator, U is the velocity vector and contains the velocity unknowns ($u_{i,j,k}$,$v_{i,j,k}$,$w_{i,j,k}$) and $\overline{U}^{\mathrm{ib}}$ is the discretization of the boundary terms. It is noteworthy to mention that the discretization of continuity equation is valid for any type of cut-cell, and in the particular case of a Cartesian fluid cell (the cell-face fraction ratios are equal to 1), Equation (3.13) reduces to the discrete continuity equation of original MAC method. In the meantime, for homogeneous boundary conditions, since no velocity is prescribed on the immersed boundary Γ^{ib}, thus we can neglect the mass flux through the solid part of cut-cell, *viz.* $\overline{U}^{\mathrm{ib}} = 0$. In this case, the discrete continuity equation is simplified to:

$$\mathcal{D} \, U = 0. \tag{3.17}$$

3.4 Discretization of the momentum equation

As in 2D geometries, the discretization of x, y and z momentum equations has been performed in distinct staggered control volumes centered around $u_{i,j,k}$, $v_{i,j,k}$ and $w_{i,j,k}$, respectively. The control volume $\Omega^u_{i,j,k}$ for the x momentum equation is formed by the halves of the neighboring cells $\Omega_{i,j,k}$ and $\Omega_{i+1,j,k}$. The discretization of the x momentum equation is then performed independently for each half of the basic types of cells in 3D extruded geometries such that any combination of half cells yields a consistent discretization with the global conservation properties discussed in Section 2.2. To explain the construction of $\Omega^u_{i,j,k}$, let us first examine the decomposition of the northern, southern, top and bottom faces of the Cartesian cell $\Omega_{i,j,k}$ shown in Figure 3.9(a), which are respectively subdivided to:

$$\begin{aligned}
\Gamma^{\mathrm{n}}_{i,j,k} &= \Gamma^{\mathrm{n,w}}_{i,j,k} \cup \Gamma^{\mathrm{n,e}}_{i,j,k}, \\
\Gamma^{\mathrm{s}}_{i,j,k} &= \Gamma^{\mathrm{s,w}}_{i,j,k} \cup \Gamma^{\mathrm{s,e}}_{i,j,k}, \\
\Gamma^{\mathrm{t}}_{i,j,k} &= \Gamma^{\mathrm{t,w}}_{i,j,k} \cup \Gamma^{\mathrm{t,e}}_{i,j,k}, \\
\Gamma^{\mathrm{b}}_{i,j,k} &= \Gamma^{\mathrm{b,w}}_{i,j,k} \cup \Gamma^{\mathrm{b,e}}_{i,j,k}.
\end{aligned} \tag{3.18}$$

The eastern part of this cell (shaded in red) can be combined with the western part of neighbor Cartesian cell $\Omega_{i+1,j,k}$ and formed the Cartesian control volume $\Omega^u_{i,j,k}$ with the volume of $\frac{1}{2}V_{i,j,k} + \frac{1}{2}V_{i+1,j,k}$, see Figure 3.10. The surface of this control volume, $\Gamma^u_{i,j,k}$, can be then decomposed to 6 elementary planes such like:

$$\Gamma^u_{i,j,k} = \Gamma^{u,\mathrm{w}}_{i,j,k} \cup \Gamma^{u,\mathrm{e}}_{i,j,k} \cup \Gamma^{u,\mathrm{s}}_{i,j,k} \cup \Gamma^{u,\mathrm{n}}_{i,j,k} \cup \Gamma^{u,\mathrm{b}}_{i,j,k} \cup \Gamma^{u,\mathrm{t}}_{i,j,k}, \tag{3.19}$$

where:

$$\begin{aligned}
\Gamma^{u,\mathrm{n}}_{i,j,k} &= \Gamma^{\mathrm{n,e}}_{i,j,k} \cup \Gamma^{\mathrm{n,w}}_{i+1,j,k}, \\
\Gamma^{u,\mathrm{s}}_{i,j,k} &= \Gamma^{\mathrm{s,e}}_{i,j,k} \cup \Gamma^{\mathrm{s,w}}_{i+1,j,k}, \\
\Gamma^{u,\mathrm{t}}_{i,j,k} &= \Gamma^{\mathrm{t,e}}_{i,j,k} \cup \Gamma^{\mathrm{t,w}}_{i+1,j,k}, \\
\Gamma^{u,\mathrm{b}}_{i,j,k} &= \Gamma^{\mathrm{b,e}}_{i,j,k} \cup \Gamma^{\mathrm{b,w}}_{i+1,j,k}.
\end{aligned} \tag{3.20}$$

An analogous decomposition can be introduced for an arbitrary control volume in the vicinity of the immersed boundary. For example in Figure 3.11, the control volume $\Omega^u_{i,j,k}$ is constructed from two adjacent cut-cells, $\Omega^u_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i+1,j,k}$ where its surface $\Gamma^u_{i,j,k}$ is decomposed to 7 elementary planes such like:

$$\Gamma^u_{i,j,k} = \Gamma^{u,\mathrm{w}}_{i,j,k} \cup \Gamma^{u,\mathrm{e}}_{i,j,k} \cup \Gamma^{u,\mathrm{s}}_{i,j,k} \cup \Gamma^{u,\mathrm{n}}_{i,j,k} \cup \Gamma^{u,\mathrm{b}}_{i,j,k} \cup \Gamma^{u,\mathrm{t}}_{i,j,k} \cup \Gamma^{u,\mathrm{ib}}_{i,j,k}. \tag{3.21}$$

in which $\Gamma^{u,\mathrm{ib}}_{i,j,k} = \Gamma^{\mathrm{ib,e}}_{i,j,k} \cup \Gamma^{\mathrm{ib,w}}_{i+1,j,k}$ represents the solid face of control volume. Similarly, the control volumes $\Omega^v_{i,j,k}$ and $\Omega^w_{i,j,k}$ can be decomposed to:

$$\Gamma^v_{i,j,k} = \Gamma^{v,\mathrm{w}}_{i,j,k} \cup \Gamma^{v,\mathrm{e}}_{i,j,k} \cup \Gamma^{v,\mathrm{s}}_{i,j,k} \cup \Gamma^{v,\mathrm{n}}_{i,j,k} \cup \Gamma^{v,\mathrm{b}}_{i,j,k} \cup \Gamma^{v,\mathrm{t}}_{i,j,k} \cup \Gamma^{v,\mathrm{ib}}_{i,j,k}, \tag{3.22a}$$

$$\Gamma^w_{i,j,k} = \Gamma^{w,\mathrm{w}}_{i,j,k} \cup \Gamma^{w,\mathrm{e}}_{i,j,k} \cup \Gamma^{w,\mathrm{s}}_{i,j,k} \cup \Gamma^{w,\mathrm{n}}_{i,j,k} \cup \Gamma^{w,\mathrm{b}}_{i,j,k} \cup \Gamma^{w,\mathrm{t}}_{i,j,k} \cup \Gamma^{w,\mathrm{ib}}_{i,j,k}, \tag{3.22b}$$

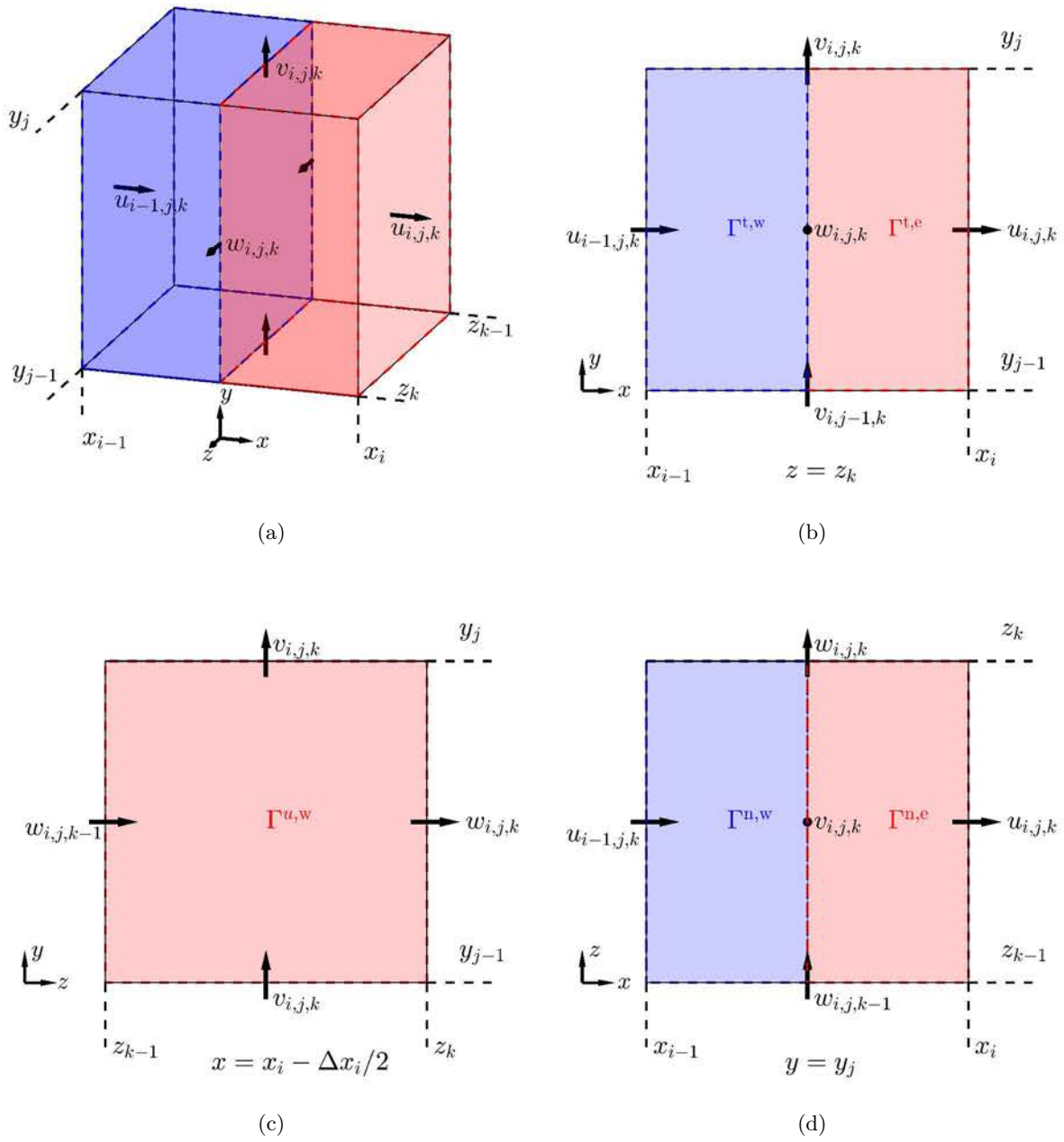


Figure 3.9: Representation of half control volumes for u in a Cartesian cell: (a) the isometric view, (b) top face at plane $z = z_k$, (c) western face at plane $x = x_i - \Delta x_i/2$ and (d) northern face at plane $y = y_j$. The blue and red halves are related to the control volumes $\Omega^u_{i-1,j,k}$ and $\Omega^u_{i,j,k}$, respectively.

where one of the elementary faces may be zero depending on the shape of control volume. For example, for a Cartesian control volume, $\Gamma^{v,\mathrm{ib}}_{i,j,k}$ and $\Gamma^{w,\mathrm{ib}}_{i,j,k}$ are zero as there is no solid face. We will subsequently focus on the discretization of different terms of the momentum equation (3.2) in these control volumes.

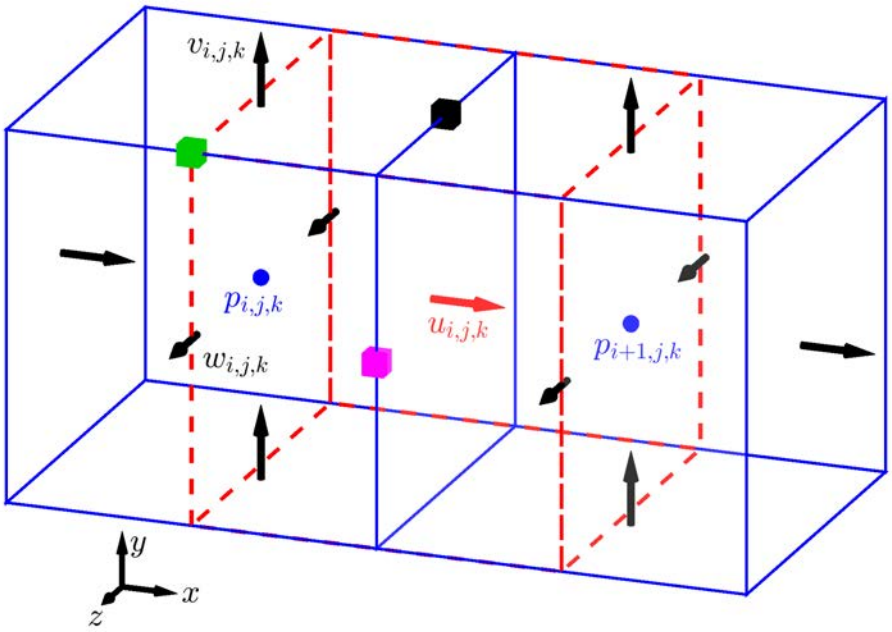


Figure 3.10: Staggered arrangement of variables in control volume $\Omega^u_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i+1,j,k}$. This control volume (--) is constructed from the halves of two adjacent Cartesian cells far away the immersed boundary.

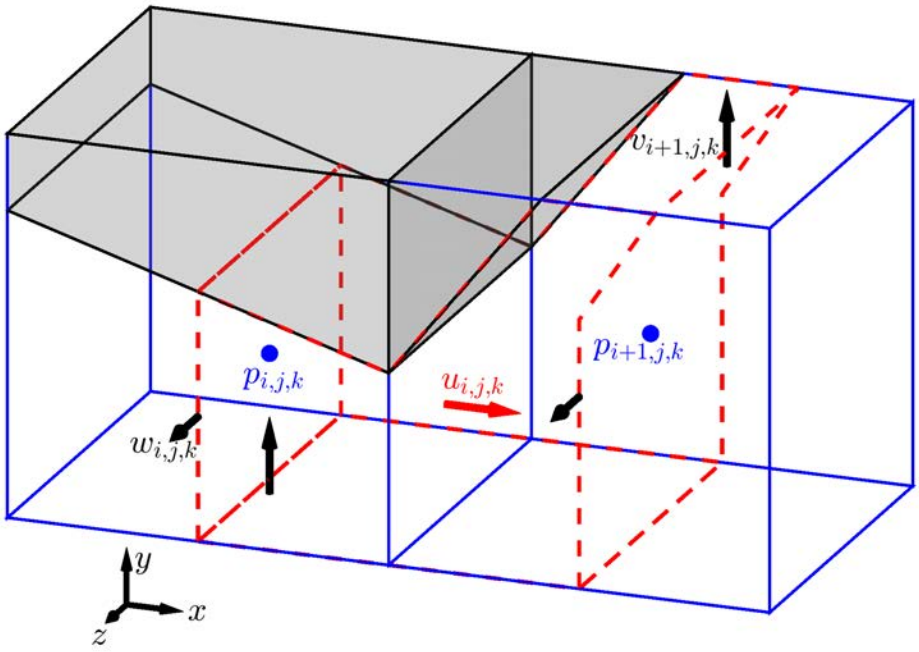


Figure 3.11: Staggered arrangement of the variables for two adjacent cut-cells in the LS-STAG mesh. The control volume $\Omega^u_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i+1,j,k}$ for $u_{i,j,k}$ is indicated in red dashed line (--).

3.4.1 Discretization of the rate of change of the momentum

In x momentum equation (3.2a), the term $\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega^u_{i,j,k}} u \ \mathrm{d}V$ describes the rate of change of momentum within an arbitrary staggered control volume $\Omega^u_{i,j,k}$, see *e.g.* Figure 3.11. The discrete form of this term in $\Omega^u_{i,j,k}$ is given by:

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega^u_{i,j,k}} u \ \mathrm{d}V \simeq \rho\frac{(u_{i,j,k}\widetilde{V}_{i,j,k})^{n+1} - (u_{i,j,k}\widetilde{V}_{i,j,k})^{n}}{\Delta t}, \tag{3.23}$$

where $\widetilde{V}_{i,j,k} = (\frac{1}{2}V_{i,j,k} + \frac{1}{2}V_{i+1,j,k})$ and Δt expresses the time-step. Similar formulas can be derived for the corresponding terms in y and z directions.

3.4.2 Discretization of the pressure gradient

In momentum equations (3.2), the term of pressure gradient is discretized based on its duality to the divergence operator, see Equation (2.27). By evaluating the terms $[-\mathcal{D}^{x\,\mathrm{T}}P]_{i,j,k}$, $[-\mathcal{D}^{y\,\mathrm{T}}P]_{i,j,k}$ and $[-\mathcal{D}^{z\,\mathrm{T}}P]_{i,j,k}$ on staggered control volumes $\Omega^u_{i,j,k}$, $\Omega^v_{i,j,k}$ and $\Omega^w_{i,j,k}$, one gets the formulas:

$$\int_{\Gamma^u_{i,j,k}} p\,\boldsymbol{e}_x\cdot\boldsymbol{n} \ \mathrm{d}S \cong [-\mathcal{D}^{x\,\mathrm{T}}P]_{i,j,k} = \theta^u_{i,j,k}\,\Delta y_j\,\Delta z_k\,(p_{i+1,j,k} - p_{i,j,k})\,, \tag{3.24a}$$

$$\int_{\Gamma^v_{i,j,k}} p\,\boldsymbol{e}_y\cdot\boldsymbol{n} \ \mathrm{d}S \cong [-\mathcal{D}^{y\,\mathrm{T}}P]_{i,j,k} = \theta^v_{i,j,k}\,\Delta x_i\,\Delta z_k\,(p_{i,j+1,k} - p_{i,j,k})\,, \tag{3.24b}$$

$$\int_{\Gamma^w_{i,j,k}} p\,\boldsymbol{e}_z\cdot\boldsymbol{n} \ \mathrm{d}S \cong [-\mathcal{D}^{z\,\mathrm{T}}P]_{i,j,k} = \theta^w_{i,j,k}\,\Delta x_i\,\Delta y_j\,(p_{i,j,k+1} - p_{i,j,k})\,, \tag{3.24c}$$

which are valid for any type of fluid cell. Note that the finite difference pressure gradient of the MAC method is recovered for the Cartesian fluid cells. For example in Figure 3.10, the pressure gradient in the control volume $\Omega^u_{i,j,k}$ far away the immersed boundaries is given by:

$$[-\mathcal{D}^{x\,\mathrm{T}}P]_{i,j,k} = \frac{1}{2}(\Delta x_i + \Delta x_{i+1})\,\Delta y_j\,\Delta z_k\,\frac{p_{i+1,j,k} - p_{i,j,k}}{\frac{1}{2}(\Delta x_i + \Delta x_{i+1})} = \Delta y_j\,\Delta z_k\,(p_{i+1,j,k} - p_{i,j,k})\,. \tag{3.25}$$

Discretization of pressure in the cut-cells can be interpreted by the nonconforming finite element method (P1 nonconforming/P0 element [27]) in which the pressure is a piecewise constant such that its precise location is not need to be defined. Therefore, $p_{i,j,k}$ is a valid approximation of pressure everywhere inside the cut-cell, even on its solid surface.

3.4.3 Skew-symmetric discretization of the convective fluxes

In this section, we aim to describe the discretization of the convective fluxes in x momentum equation (3.2a). For this purpose, we will first examine the discrete form of convective terms in a Cartesian control volume (original MAC method). Then, in order to obtain the discrete form of fluxes in the vicinity of immersed boundary, some terms are modified based on the shape of control volume. Note that similar formulas can be also extracted for the discrete convective terms in y and z momentum equations ((3.2b) and (3.2c)).

In x momentum equation (3.2a), operator $\mathcal{C}^x[\overline{U}]$ corresponds to the discretization of convective fluxes which should be a skew-symmetric matrix thanks to the condition (2.28) to ensure the conservation of kinetic energy budget. Figure 3.10 indicates a Cartesian control volume, $\Omega^u_{i,j,k}$, far away the immersed boundary. Discretization of the convective fluxes in this control volume is written as the 7-point scheme in 3D:

$$\begin{aligned}\int_{\Gamma^u_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n})u \;\; \mathrm{d}S \cong & \mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k)\, u_{i,j,k} \\ & + \mathcal{C}^x[\overline{U}]_{\mathrm{W}}(i,j,k)\, u_{i-1,j,k} + \mathcal{C}^x[\overline{U}]_{\mathrm{E}}(i,j,k)\, u_{i+1,j,k} \\ & + \mathcal{C}^x[\overline{U}]_{\mathrm{S}}(i,j,k)\, u_{i,j-1,k} + \mathcal{C}^x[\overline{U}]_{\mathrm{N}}(i,j,k)\, u_{i,j+1,k} \\ & + \mathcal{C}^x[\overline{U}]_{\mathrm{B}}(i,j,k)\, u_{i,j,k-1} + \mathcal{C}^x[\overline{U}]_{\mathrm{T}}(i,j,k)\, u_{i,j,k+1},\end{aligned} \tag{3.26}$$

such that the coefficients of matrix $\mathcal{C}^x[\overline{U}]$ satisfy the skew-symmetry conditions:

$$\mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k) = 0, \tag{3.27a}$$
$$\mathcal{C}^x[\overline{U}]_{\mathrm{E}}(i,j,k) = -\,\mathcal{C}^x[\overline{U}]_{\mathrm{W}}(i+1,j,k), \tag{3.27b}$$
$$\mathcal{C}^x[\overline{U}]_{\mathrm{N}}(i,j,k) = -\,\mathcal{C}^x[\overline{U}]_{\mathrm{S}}(i,j+1,k), \tag{3.27c}$$
$$\mathcal{C}^x[\overline{U}]_{\mathrm{T}}(i,j,k) = -\,\mathcal{C}^x[\overline{U}]_{\mathrm{B}}(i,j,k+1). \tag{3.27d}$$

On uniform meshes, the second-order central discretization of MAC method trivially satisfies above conditions, while employing other common types of discretization such as upwind, entails adding artificial viscosity to the scheme. In LS-STAG method, the skew-symmetric discretization of the convective terms is achieved by employing the method suggested by Verstappen and Veldman [102]. It can be applied to the non-uniform meshes and will be also useful in discretization of convective fluxes in the cut-cells. For the Cartesian control volume $\Omega^u_{i,j,k}$ in Figure 3.10, the convective terms are written as the net flux through its elementary faces such that:

$$\begin{aligned}\int_{\Gamma^u_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n})\, u \;\; \mathrm{d}S = & -\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\, u \;\; \mathrm{d}S + \int_{\Gamma^{u,\mathrm{e}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\, u \;\; \mathrm{d}S \\ & - \int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\, u \;\; \mathrm{d}S + \int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\, u \;\; \mathrm{d}S \\ & - \int_{\Gamma^{u,\mathrm{b}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\, u \;\; \mathrm{d}S + \int_{\Gamma^{u,\mathrm{t}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\, u \;\; \mathrm{d}S.\end{aligned} \tag{3.28}$$

According to Equation (3.12a), for the eastern face of control volume:

$$\int_{\Gamma^{u,\mathrm{e}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\, u \;\; \mathrm{d}S \cong \Big(\frac{1}{2}\,\overline{u}_{i,j,k} + \frac{1}{2}\,\overline{u}_{i+1,j,k}\Big)\, u_{\mathrm{e}}, \tag{3.29}$$

where u_{e} is a characteristic value of the u velocity on surface $\Gamma^{u,\mathrm{e}}_{i,j,k}$ and can be interpolated such that:

$$u_{\mathrm{e}} = \alpha_{\mathrm{e}}\, u_{i,j,k} + (1-\alpha_{\mathrm{e}})\, u_{i+1,j,k}, \tag{3.30}$$

where α_{e} is the interpolation weight for u_{e}. Similar interpolations can be also applied to other faces. For instance, the discrete convective fluxes in the northern and top faces are described such that:

$$\int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S \cong \Big(\frac{1}{2}\,\overline{v}_{i,j,k} + \frac{1}{2}\,\overline{v}_{i+1,j,k}\Big)\,u_{\mathrm{n}}, \tag{3.31a}$$

$$\int_{\Gamma^{u,\mathrm{t}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u \;\; \mathrm{d}S \cong \Big(\frac{1}{2}\,\overline{w}_{i,j,k} + \frac{1}{2}\,\overline{w}_{i+1,j,k}\Big)\,u_{\mathrm{t}}. \tag{3.31b}$$

Similarly, u_{n} and u_{t} are written as $u_{\mathrm{n}} = \alpha_{\mathrm{n}}\,u_{i,j,k} + (1-\alpha_{\mathrm{n}})\,u_{i,j+1,k}$ and $u_{\mathrm{t}} = \alpha_{\mathrm{t}}\,u_{i,j,k} + (1-\alpha_{\mathrm{t}})\,u_{i,j,k+1}$ in which α_{n} and α_{t} are the weights for u_{n} and u_{t}, respectively. Additionally, considering the property of local conservation of fluxes through the fluid faces:

$$\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\; \mathrm{d}S = \int_{\Gamma^{u,\mathrm{e}}_{i-1,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\; \mathrm{d}S, \tag{3.32a}$$

$$\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S = \int_{\Gamma^{u,\mathrm{n}}_{i,j-1,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S, \tag{3.32b}$$

$$\int_{\Gamma^{u,\mathrm{b}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u \;\; \mathrm{d}S = \int_{\Gamma^{u,\mathrm{t}}_{i,j,k-1}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u \;\; \mathrm{d}S, \tag{3.32c}$$

yields to obtain the coefficients of discretization as follows:

$$\begin{aligned}\mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k) = &\Big[\frac{\alpha_{\mathrm{e}}}{2}\,\overline{u}_{i,j,k} - \frac{(1-\alpha_{\mathrm{e}})}{2}\,\overline{u}_{i-1,j,k}\Big] + \Big[\frac{\alpha_{\mathrm{e}}}{2}\,\overline{u}_{i+1,j,k} - \frac{(1-\alpha_{\mathrm{e}})}{2}\,\overline{u}_{i,j,k}\Big]\\ &+\Big[\frac{\alpha_{\mathrm{n}}}{2}\,\overline{v}_{i,j,k} - \frac{(1-\alpha_{\mathrm{n}})}{2}\,\overline{v}_{i,j-1,k}\Big] + \Big[\frac{\alpha_{\mathrm{n}}}{2}\,\overline{v}_{i+1,j,k} - \frac{(1-\alpha_{\mathrm{n}})}{2}\,\overline{v}_{i+1,j-1,k}\Big]\\ &+\Big[\frac{\alpha_{\mathrm{t}}}{2}\,\overline{w}_{i,j,k} - \frac{(1-\alpha_{\mathrm{t}})}{2}\,\overline{w}_{i,j,k-1}\Big] + \Big[\frac{\alpha_{\mathrm{t}}}{2}\,\overline{w}_{i+1,j,k} - \frac{(1-\alpha_{\mathrm{t}})}{2}\,\overline{w}_{i+1,j,k-1}\Big],\end{aligned} \tag{3.33a}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{W}}(i,j,k) = -\frac{\alpha_{\mathrm{e}}}{2}\,\big(\overline{u}_{i-1,j,k} + \overline{u}_{i,j,k}\big), \tag{3.33b}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{E}}(i,j,k) = \frac{(1-\alpha_{\mathrm{e}})}{2}\,\big(\overline{u}_{i,j,k} + \overline{u}_{i+1,j,k}\big), \tag{3.33c}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{S}}(i,j,k) = -\frac{\alpha_{\mathrm{n}}}{2}\,\big(\overline{v}_{i,j-1,k} + \overline{v}_{i+1,j-1,k}\big), \tag{3.33d}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{N}}(i,j,k) = \frac{(1-\alpha_{\mathrm{n}})}{2}\,\big(\overline{v}_{i,j,k} + \overline{v}_{i+1,j,k}\big), \tag{3.33e}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{B}}(i,j,k) = -\frac{\alpha_{\mathrm{t}}}{2}\,\big(\overline{w}_{i,j,k-1} + \overline{w}_{i+1,j,k-1}\big), \tag{3.33f}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{T}}(i,j,k) = \frac{(1-\alpha_{\mathrm{t}})}{2}\,\big(\overline{w}_{i,j,k} + \overline{w}_{i+1,j,k}\big). \tag{3.33g}$$

These coefficients are skew-symmetric (3.27) solely if we employ the central interpolation with equal weighting for the unknown velocities, *i.e.* $\alpha_{\mathrm{e}} = \alpha_{\mathrm{n}} = \alpha_{\mathrm{t}} = 1/2$, refer to Verstappen and Veldman [102]. Substituting these weights into the main diagonal term $\mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k)$ gives:

$$\mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k) = \tfrac{1}{4}\,\dot{m}_{i,j,k} + \tfrac{1}{4}\,\dot{m}_{i+1,j,k}, \tag{3.34}$$

in which the condition (3.27a) is verified as soon as the continuity equation (3.13) is satisfied in control volumes $\Omega_{i,j,k}$ and $\Omega_{i+1,j,k}$.

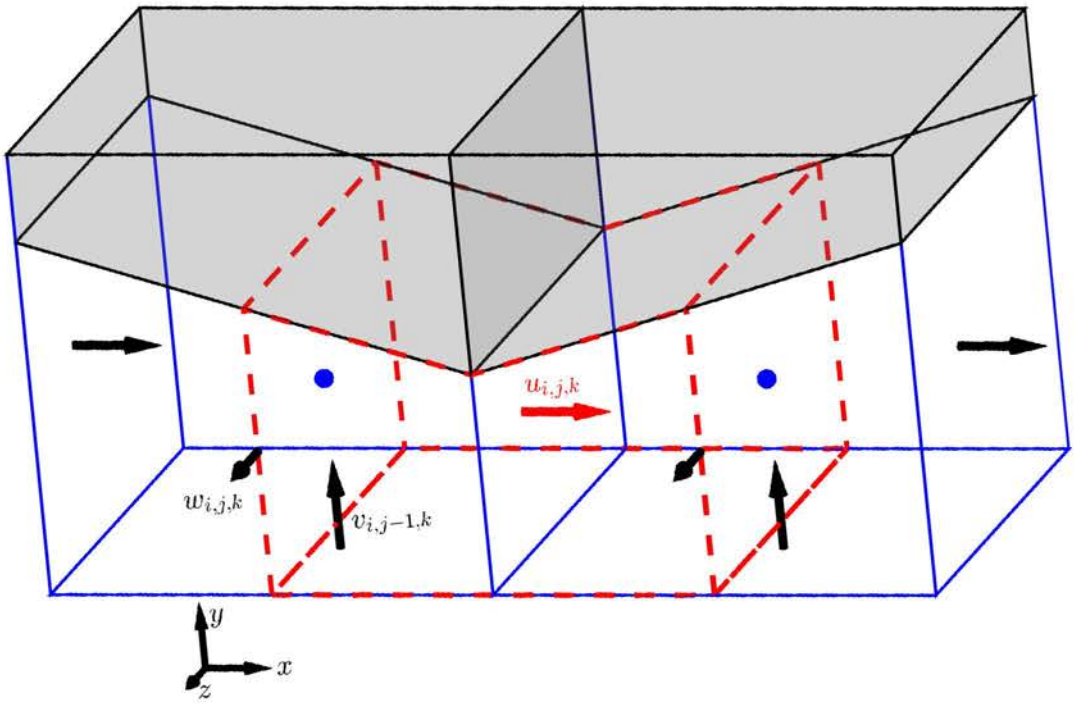


Figure 3.12: Combination of the halves of 2 adjacent trapezoidal cut-cells which forms the control volume $\Omega^u_{i,j,k}$ (Red dashed line (--)).

However in the cut-cells, some modifications must be applied on skew-symmetric discretization of the fluxes given by equations (3.32) and (3.33) to take into account the boundary conditions on the immersed boundary. Discretization of the convective terms in the cut-cells is not as straightforward as that for the pressure gradient since there is no generic formula valid for all cell types. In fact, it is constructed based on the half control volumes inside each basic cell such that any combination of these half control volumes verifies the skew-symmetry condition $\mathcal{C}[U] = -\mathcal{C}[U]^{\mathrm{T}}$. As an example, we consider control volume $\Omega^u_{i,j,k}$ shown in Figure 3.12 which is constructed from the halves of two adjacent trapezoidal cut-cells. Discretization of the convective terms in $\Omega^u_{i,j,k}$ takes the form:

$$\begin{aligned}\int_{\Gamma^u_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n})\, u \;\mathrm{d}S \cong\; & \mathcal{C}^x[U]_{\mathrm{W}}(i,j,k)\, u_{i-1,j,k} + \mathcal{C}^x[U]_{\mathrm{E}}(i,j,k)\, u_{i+1,j,k} \\ & + \mathcal{C}^x[U]_{\mathrm{B}}(i,j,k)\, u_{i,j,k-1} + \mathcal{C}^x[U]_{\mathrm{T}}(i,j,k)\, u_{i,j,k+1} \\ & + \mathcal{C}^x[U]_{\mathrm{S}}(i,j,k)\, u_{i,j-1,k} + \mathcal{C}^x[U]_{\mathrm{P}}(i,j,k)\, u_{i,j,k} \;+\; [S^{\mathrm{ib,c}}_x](i,j,k),\end{aligned} \tag{3.35}$$

where the contribution of northern face, $\mathcal{C}^x[U]_{\mathrm{N}}(i,j,k)$, is discarded since the velocity unknown $u_{i,j+1,k}$ is not a part of the fluid domain. In the meantime, contribution of solid part at the northern face should be taken into account as the source term $[S^{\mathrm{ib,c}}_x](i,j,k)$. Thus, the skew-

symmetry condition (2.28) is written as:

$$\mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k) = 0, \tag{3.36a}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{E}}(i,j,k) = -\,\mathcal{C}^x[\overline{U}]_{\mathrm{W}}(i+1,j,k), \tag{3.36b}$$

$$\mathcal{C}^x[\overline{U}]_{\mathrm{T}}(i,j,k) = -\,\mathcal{C}^x[\overline{U}]_{\mathrm{B}}(i,j,k+1). \tag{3.36c}$$

In order to obtain these conditions, we write the convective terms as the net flux through the faces of control volume $\Omega^u_{i,j,k}$:

$$\begin{aligned}\int_{\Gamma^u_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n})\,u \;\mathrm{d}S = &-\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\mathrm{d}S + \int_{\Gamma^{u,\mathrm{e}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\mathrm{d}S \\ &-\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\mathrm{d}S + \int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n}^{\mathrm{ib}})\,u \;\mathrm{d}S \\ &-\int_{\Gamma^{u,\mathrm{b}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u \;\mathrm{d}S + \int_{\Gamma^{u,\mathrm{t}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u \;\mathrm{d}S.\end{aligned} \tag{3.37}$$

For the fluid faces, fluxes are calculated by equations (3.29) and (3.31), whereas the flux through the solid faces are discretized as:

$$\int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n}^{\mathrm{ib}})\,u\,\mathrm{d}S \cong \frac{\overline{U}^{\mathrm{ib}}_{i,j,k}}{2}\left(\frac{1}{2}\Big[\underline{u_{i,j,k}} + \underline{\underline{u(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k)}}\Big]\right) + \frac{\overline{U}^{\mathrm{ib}}_{i+1,j,k}}{2}\left(\frac{1}{2}\Big[\underline{u_{i,j,k}} + \underline{\underline{u(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k)}}\Big]\right), \tag{3.38}$$

where the underlined terms contribute to the diagonal coefficient $(\mathcal{C}^x[\overline{U}]_{\mathrm{P}}(i,j,k)\,u_{i,j,k})$ in order to recover the discrete continuity expression in control volumes $\Omega_{i,j,k}$ and $\Omega_{i+1,j,k}$, whereas the double underlined terms is the contribution of source terms $S^{\mathrm{ib,c}}_{i,j,k}$. Consequently, discretization of the convective terms in control volume $\Omega^u_{i,j,k}$ indicated in Figure 3.12, will be identical to equation 3.33, except:

$$\begin{aligned}&\mathcal{C}[\overline{U}]_{\mathrm{N}}(i,j,k) = 0, \\ &S^{\mathrm{ib,c}}_{i,j,k} = \frac{1}{2}\Big(\frac{\overline{U}^{\mathrm{ib}}_{i,j,k}}{2} + \frac{\overline{U}^{\mathrm{ib}}_{i+1,j,k}}{2}\Big)\,u(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k).\end{aligned} \tag{3.39}$$

Thus, the skew-symmetry conditions (3.36) is verified. It is important to mention that the source term $S^{\mathrm{ib,c}}_{i,j,k}\,u_{i,j,k}$ in the kinetic energy budget (2.26) is attributed to a discrete approximation of term $\int_{\Gamma^{\mathrm{ib}}} (\rho\boldsymbol{v}^2/2)\,\boldsymbol{v}\cdot\boldsymbol{n}\;\mathrm{d}S$ written on the solid boundary $\Gamma^{u,\mathrm{ib}}_{i,j,k}$. For other combination of cut-cells, discretization of the convective terms is constructed such that the skew-symmetry condition (2.28) is satisfied. The discrete form of convective fluxes for different types of cut-cells is given in appendix A.

3.4.4 Discretization of velocity gradients on LS-STAG mesh

The discretization of velocity gradients plays an important role in the numerical modeling of diffusive terms presented in Section 3.4.5 since it defines the rate-of-strain tensor $\dot{\boldsymbol{\gamma}}$ and shear-rate $\dot{\gamma}$ in non-Newtonian flows. Thus, we will first discretize the components of velocity gradient

tensor. In 3D configurations, the tensor of velocity gradients is written as:

$$\nabla \boldsymbol{v} = \begin{pmatrix} \dfrac{\partial u}{\partial x} & \dfrac{\partial v}{\partial x} & \dfrac{\partial w}{\partial x} \\ \dfrac{\partial u}{\partial y} & \dfrac{\partial v}{\partial y} & \dfrac{\partial w}{\partial y} \\ \dfrac{\partial u}{\partial z} & \dfrac{\partial v}{\partial z} & \dfrac{\partial w}{\partial z} \end{pmatrix}. \tag{3.40}$$

As a key element in discretization of the velocity gradients, there is a distinction between the location of normal gradients (diagonal terms) *i.e.* $\partial u/\partial x$, $\partial v/\partial y$ and $\partial w/\partial z$, and the tangential ones (off-diagonal elements) *e.g.* $\partial u/\partial y$ or $\partial v/\partial z$, [22]. In other word, the diagonal and off-diagonal components of $\nabla \boldsymbol{v}$ are located at different locations in a fluid cell due to the staggered arrangement of velocity unknowns. This distinction is also observed in original MAC method in which the normal velocity gradients are computed by finite difference approximation as:

$$\left.\frac{\partial u}{\partial x}\right|_{i,j,k} = \frac{u_{i,j,k} - u_{i-1,j,k}}{\Delta x_i}, \tag{3.41}$$

and are naturally located at the cells centroids, while the tangential velocity gradients are located at the edges centers. For instance in Figure 3.10, $\partial u/\partial y|_{i,j,k}$ and $\partial u/\partial z|_{i,j,k}$ are discretized:

$$\left.\frac{\partial u}{\partial y}\right|_{i,j,k} = \frac{u_{i,j+1,k} - u_{i,j,k}}{\frac{1}{2}\,\Delta y_{j+1} + \frac{1}{2}\,\Delta y_j}, \tag{3.42a}$$

$$\left.\frac{\partial u}{\partial z}\right|_{i,j,k} = \frac{u_{i,j,k+1} - u_{i,j,k}}{\frac{1}{2}\,\Delta z_{k+1} + \frac{1}{2}\,\Delta z_k}. \tag{3.42b}$$

These terms are located at the centers of edges $\mathcal{E}^z_{i,j,k}$ and $\mathcal{E}^y_{i,j,k}$ respectively, see (■) and (■). This treatment of diagonal and off-diagonal elements has a strong analogy with the finite element approach of Saramito [88] for the viscoelastic fluids where the normal and shear stresses are unknowns of the problem. Saramito proposed a combination of the Raviart and Thomas mixed element [85] for the velocity-pressure, with the piecewise constant polynomials with a degree of freedom at the cell centroid for the normal stresses τ_{xx} and τ_{yy}, and a linear approximation for the shear stresses τ_{xy}, see Figure 3.13.

In the cut-cells, we assume that the normal velocity gradients have a physical origin and a mathematical regularity similar to that of the pressure, thus their discretization must be consistent with that of the pressure terms treated in the Section 3.4.2. In other words, these diagonal components ($\partial u/\partial x$, $\partial v/\partial y$ and $\partial w/\partial z$) are supposed to possess constant values in each cut-cell and their locations are not required to be precisely defined inside the cut-cell. In addition, the tangential velocity gradients are located at the midpoints of fluid edges, see Figure 3.6. It is noteworthy to remark that the terms $\partial u/\partial y$ and $\partial v/\partial x$ in the pentagonal cut-cells are located in 2 distinct points, see for example Figure 3.6(c).

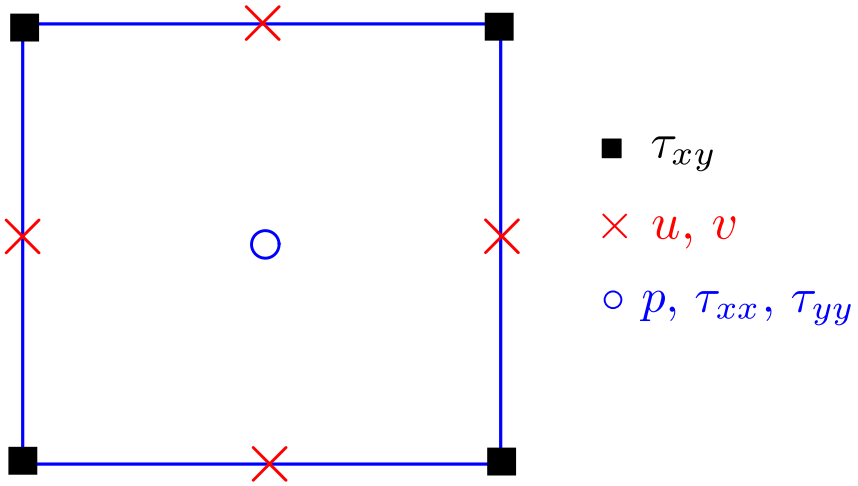


Figure 3.13: The rectangular element of Saramito [88].

Discretization of diagonal elements of velocity gradient tensor

In this part, we will pursue discretization of the diagonal elements of $\nabla \boldsymbol{v}$ in the cut-cells. The first and simplest idea is to discretize these terms based on the finite difference approximations. In 3D extruded LS-STAG, the orthogonality property of MAC mesh is still retained in the direction of extrusion (z). For example in Figure 3.14, we can see that the line connecting two consecutive w velocity vectors, *e.g.* $w_{i,j,k}$ and $w_{i-1,j,k}$, is perpendicular to the face $\Gamma^{w,\mathrm{b}}_{i,j,k}$. Thus, we can adopt a backward difference approximation such like Equation (3.41) for the discretization of $\partial w/\partial z$. However, for other diagonal components, *i.e.* $\partial u/\partial x$ and $\partial v/\partial y$, LS-STAG mesh in the cut-cells is no longer admissible in the sense of Eymard *et al.* [30] as the orthogonality property is lost. Applying standard two-point finite difference formula to these terms would not be successful and yields large numerical errors. Similar issue is also observed for the off-diagonal terms $\partial w/\partial x$ and $\partial w/\partial y$ in z momentum equation (3.2c) which will be discussed later in Section 3.4.4.

It is shown by Droniou [29] that when the orthogonality property is lost, linearly exact schemes cannot be achieved with a two-points formula and requires multi-points formulas or introduction of extra vertex unknowns. In this direction, the advanced finite volume schemes presented in [29] such as the *MPFA*[1] [1], or *HFV*[2] [31], or *DDFV*[3] [49] methods are applicable on generic meshes. In addition, according to Moukalled *et al.* [70], we can take into account the non-orthogonality contribution (cross-diffusion) by adding the non-orthogonal correction. However, employing these methods requires data interpolations and leads to an enlarged stencil in the vicinity of the cut-cells. Thus, the design of such schemes is left for a future work.

Hereon, for the normal velocity gradients, the non-orthogonality issue is tackled by employing

[1]Multi-Point Flux Approximation
[2]Hybrid Finite Volume
[3]Discrete Duality Finite Volume

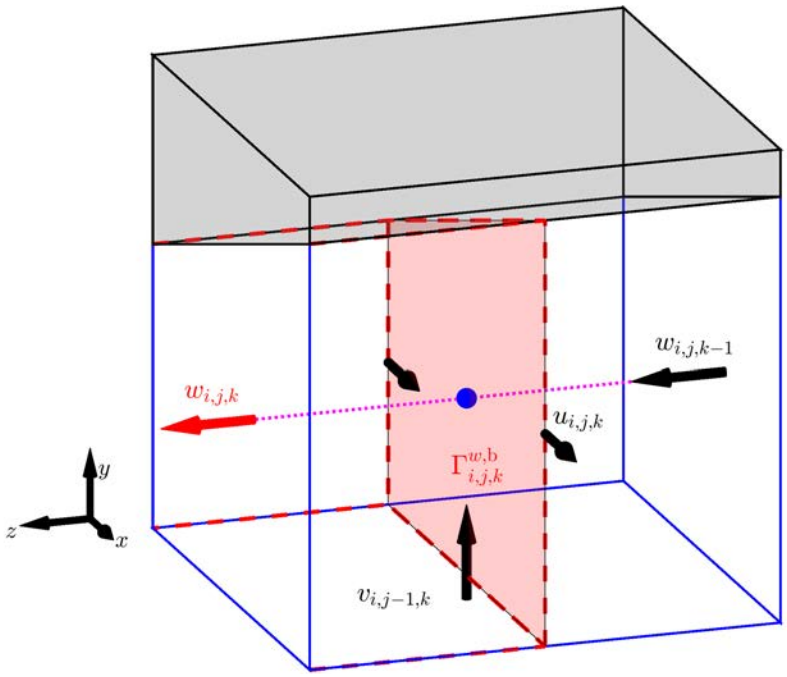


Figure 3.14: Representation of the half control volumes (--) for $w_{i,j,k}$. The line (··) connecting two consecutive velocity vectors $w_{i,j,k}$ and $w_{i,j,k-1}$ is orthogonal to $\Gamma^{w,\mathrm{b}}_{i,j,k}$ due to the extrusion in z direction.

the divergence theorem:

$$\int_{\Omega_{i,j,k}} \nabla \cdot \boldsymbol{v} \; \mathrm{d}V = \int_{\Omega_{i,j,k}} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \; \mathrm{d}V = \int_{\Gamma_{i,j,k}} \boldsymbol{v} \cdot \boldsymbol{n} \; \mathrm{d}S, \tag{3.43}$$

which should be valid at the discrete level in the cut-cells. Note that the divergence theorem is trivially verified by the MAC method in a Cartesian cell. In Equation (3.43), the volume integral is discretized as:

$$\int_{\Omega_{i,j,k}} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \; \mathrm{d}V \cong \left(\left.\frac{\partial u}{\partial x}\right|_{i,j,k} + \left.\frac{\partial v}{\partial y}\right|_{i,j,k} + \left.\frac{\partial w}{\partial z}\right|_{i,j,k} \right) V_{i,j,k}, \tag{3.44}$$

and the surface integral is replaced by the discrete continuity Equation (3.13) such that:

$$\begin{aligned} \int_{\Gamma_{i,j,k}} \boldsymbol{v} \cdot \boldsymbol{n} \; \mathrm{d}S \equiv{}& \Delta y_j \, \Delta z_k \left(\theta^u_{i,j,k} \, u_{i,j,k} - \theta^u_{i-1,j,k} \, u_{i-1,j,k} \right) + \Delta y_j \, \Delta z_k \left(\theta^u_{i-1,j,k} - \theta^u_{i,j,k} \right) u^{\mathrm{ib}}_{i,j,k} \\ &+ \Delta x_i \, \Delta z_k \left(\theta^v_{i,j,k} \, v_{i,j,k} - \theta^v_{i,j-1,k} \, v_{i,j-1,k} \right) + \Delta x_i \, \Delta z_k \left(\theta^v_{i,j-1,k} - \theta^v_{i,j,k} \right) v^{\mathrm{ib}}_{i,j,k} \\ &+ \Delta x_i \, \Delta y_j \left(\theta^w_{i,j,k} \, w_{i,j,k} - \theta^w_{i,j,k-1} \, w_{i,j,k-1} \right) + \Delta x_i \, \Delta y_j \left(\theta^w_{i,j,k-1} - \theta^w_{i,j,k} \right) w^{\mathrm{ib}}_{i,j,k}. \end{aligned} \tag{3.45}$$

By substitution of (3.44) and (3.45) in (3.43), one gets the discrete forms of the normal velocity

gradients as:

$$\left.\frac{\partial u}{\partial x}\right|_{i,j,k} \cong \frac{\theta^u_{i,j,k}\, u_{i,j,k} - \theta^u_{i-1,j,k}\, u_{i-1,j,k} + (\,\theta^u_{i-1,j,k} - \theta^u_{i,j,k}\,)\, u^{\text{ib}}_{i,j,k}}{V_{i,j,k}/\Delta y_j \Delta z_k}, \tag{3.46a}$$

$$\left.\frac{\partial v}{\partial y}\right|_{i,j,k} \cong \frac{\theta^v_{i,j,k}\, v_{i,j,k} - \theta^v_{i,j-1,k}\, v_{i,j-1,k} + (\,\theta^v_{i,j-1,k} - \theta^v_{i,j,k}\,)\, v^{\text{ib}}_{i,j,k}}{V_{i,j,k}/\Delta x_i \Delta z_k}, \tag{3.46b}$$

$$\left.\frac{\partial w}{\partial z}\right|_{i,j,k} \cong \frac{\theta^w_{i,j,k}\, w_{i,j,k} - \theta^w_{i,j,k-1}\, w_{i,j,k-1} + (\,\theta^w_{i,j,k-1} - \theta^w_{i,j,k}\,)\, w^{\text{ib}}_{i,j,k}}{V_{i,j,k}/\Delta x_i \Delta y_j}. \tag{3.46c}$$

These formulas are valid for any type of cut-cells with naturally embedded boundary conditions. In addition, for the Cartesian fluid cells in which all cell-face fraction ratios (θ^u, θ^v and θ^w) are equal to 1, the standard finite difference approximations in original MAC method are also recovered, see *e.g.* Equation (3.41). We will subsequently take into account the discretization of off-diagonal components in the tensor of velocity gradients.

Discretization of the off-diagonal elements: $\partial u/\partial y$ and $\partial v/\partial x$

The discretization of $\partial u/\partial y$ in control volume $\Omega^u_{i,j,k}$ will be investigated in this section. According to Figure 3.15, the discrete form of $\partial u/\partial y|_{i,j-1,k}$ which is located at the center of edge $\mathcal{E}^z_{i,j-1,k}$, is obtained by differentiating the interpolation polynomial of $u(x_i, \cdot)$ in the y direction as:

$$\left.\frac{\partial u}{\partial y}\right|_{i,j-1,k} = \frac{u_{i,j,k} - u_{i,j-1,k}}{\frac{1}{2}\,\theta^u_{i,j-1,k}\,\Delta y_{j-1} + \frac{1}{2}\,\theta^u_{i,j,k}\,\Delta y_j}. \tag{3.47}$$

This formula is valid only when both faces $\Gamma^{\text{e}}_{i,j,k}$ and $\Gamma^{\text{e}}_{i,j-1,k}$ are fluid *i.e.* $\theta^u_{i,j,k} > 0$ and $\theta^u_{i,j-1,k} > 0$. However, if one of this faces are solid, we modify this formula in the manner of *ghost fluid method* for elliptic problems [41] in order to implement the boundary condition. For example in Figure 3.15, the northern face of $\Omega^u_{i,j,k}$ is solid, thus $\theta^u_{i,j+1,k} = 0$. By employing a one-sided formula, $\partial u/\partial y|_{i,j,k}$ is discretized such that:

$$\left.\frac{\partial u}{\partial y}\right|_{i,j,k} = \frac{u(x_i, y^{\text{ib}}_{i,j,k}, z_k) - u_{i,j,k}}{\frac{1}{2}\,\theta^u_{i,j,k}\,\Delta y_j}. \tag{3.48}$$

There is also another possibility in which both adjacent faces are solid, for example in Figure 3.16, both $\theta^u_{i,j,k}$ and $\theta^u_{i,j+1,k}$ are equal to zero. This special case happens only for $\partial u/\partial y$ and $\partial v/\partial x$ in 3D extruded meshes. To derive a discrete expression for $\partial u/\partial y|_{i,j,k}$ in Figure 3.16 (■), a naive approach is to neglect its contribution and set it zero, as previously done in 2D version [22, 15]. This is true if the geometry is horizontal. However, in 3D configurations, the discretization of these velocity gradients are derived by employing the Schwarz theorem (symmetry of second derivatives). This theorem is applied to $\partial u/\partial y$ such that:

$$\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}\right) = \frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x}\right), \tag{3.49}$$

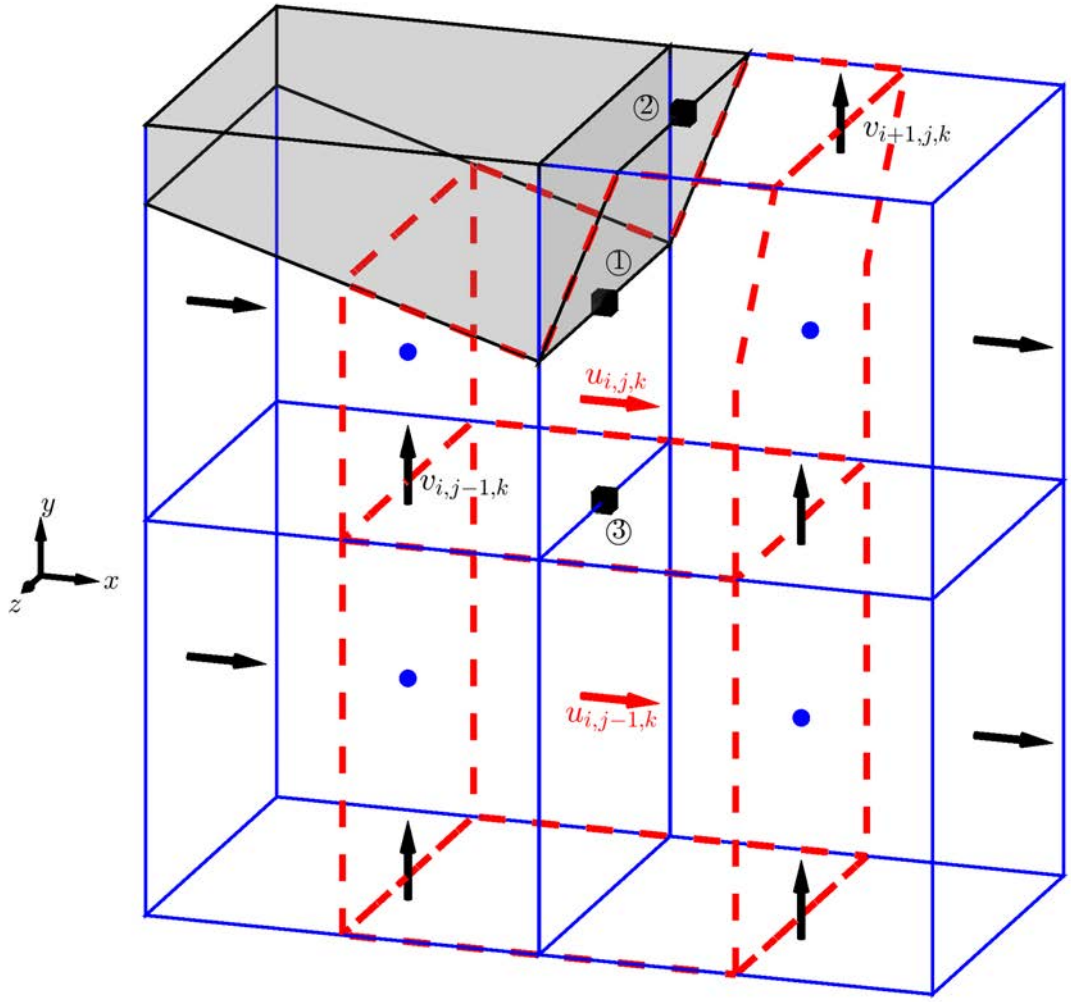


Figure 3.15: Representation of the control volumes $\Omega^u_{i,j,k}$ and $\Omega^u_{i,j-1,k}$ in red dashed line (--). $\Omega^u_{i,j-1,k}$ is formed from 2 Cartesian cells and $\Omega^u_{i,j,k}$ is in the vicinity of immersed boundary. The velocity gradients $\partial u/\partial y|_{i,j,k}$, $\partial v/\partial x|_{i,j,k}$ and $\partial u/\partial y|_{i,j-1,k}$ ($\partial v/\partial x|_{i,j-1,k}$) are denoted with ①, ② and ③, respectively.

holds at the center of face $\Gamma^{\mathrm{n}}_{i,j,k}$ in Figure 3.16. A straightforward discretization of $\partial(\cdot)/\partial x$ in the LHS of Equation (3.49) leads to:

$$\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}\right)\bigg|_{i,j,k} = \frac{1}{\alpha^x_{i,j,k}\,\Delta x_i}\left(\frac{\partial u}{\partial y}\bigg|_{i,j,k} - \frac{\partial u}{\partial y}\bigg|_{i-1,j,k}\right). \tag{3.50}$$

A finite difference formula has to be designed to approximate the operator $\partial(\cdot)/\partial y$ in the RHS of Equation (3.49) in order to complete the discretization. Since both pressure unknowns and $\partial u/\partial x$ are located at the cells centroids, we can employ similar approach as that in discretization of the pressure gradient (3.24), such that:

$$\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x}\right)\bigg|_{i,j,k} = \frac{\theta^v_{i,j,k}\,\Delta x_i\,\Delta z_k}{\frac{1}{2}(V_{i,j+1,k} + V_{i,j,k})}\left(\frac{\partial u}{\partial x}\bigg|_{i,j+1,k} - \frac{\partial u}{\partial x}\bigg|_{i,j,k}\right). \tag{3.51}$$

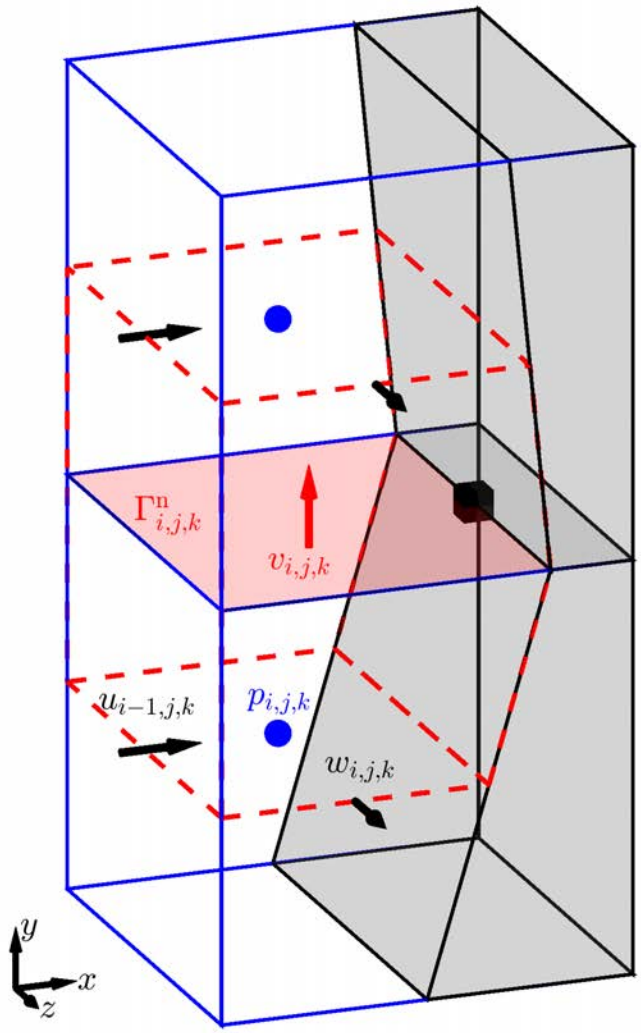


Figure 3.16: Two consecutive solid faces form the eastern face of v control volume, $\Omega^{v}_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i,j+1,k}$ (--). In this particular case, the velocity gradient $\partial u/\partial y|_{i,j,k}$ (■) is computed by employing the Schwarz theorem. The northern face $\Gamma^{\mathrm{n}}_{i,j,k}$ of control volume $\Omega_{i,j,k}$ is shaded in red.

After identification of Equation (3.50) with Equation (3.51), we finally get:

$$\left.\frac{\partial u}{\partial y}\right|_{i,j,k} = \left.\frac{\partial u}{\partial y}\right|_{i-1,j,k} + \frac{2\alpha^{x}_{i,j,k}\,\theta^{v}_{i,j,k}\,\Delta z_k\,(\Delta x_i)^2}{V_{i,j+1,k}+V_{i,j,k}}\left(\left.\frac{\partial u}{\partial x}\right|_{i,j+1,k} - \left.\frac{\partial u}{\partial x}\right|_{i,j,k}\right), \tag{3.52}$$

in which $\partial u/\partial y|_{i-1,j,k}$ has an analogous form to (3.47), and the normal velocity gradients are computed by the generic formula (3.46). Similar formulas as (3.47), (3.48) and (3.52) are also obtained for the term $\partial v/\partial x$.

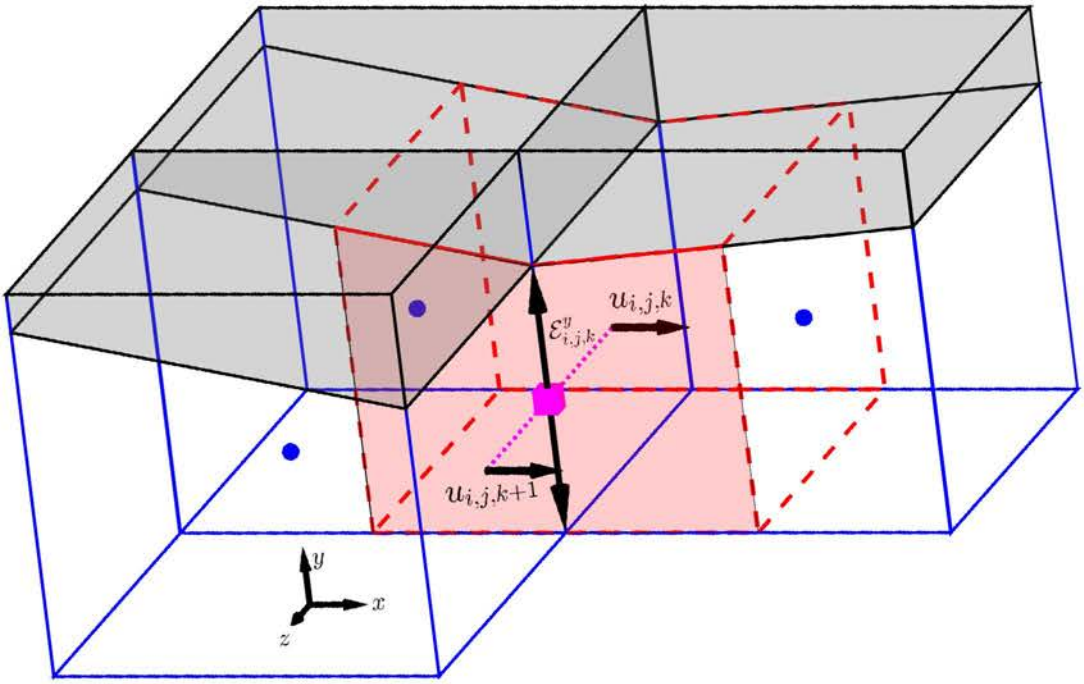


Figure 3.17: Velocity gradient $\partial u/\partial z|_{i,j,k}$ (■) is located at the center of edge $\mathcal{E}^y_{i,j,k}$. The face $\Gamma^{u,\mathrm{t}}$ in the control volume $\Omega^u_{i,j,k,}$ is shaded in red.

Discretization of the off-diagonal elements: $\partial u/\partial z$ and $\partial v/\partial z$

Now, we take into account the new terms introduced in x and y momentum equations, *i.e.* $\partial u/\partial z$ and $\partial v/\partial z$ in Equations (3.2a) and (3.2b), respectively. In the following, we will explain the discretization of $\partial u/\partial z$. Analogous formula is also obtained for $\partial v/\partial z$. According to Figure 3.17, the velocity gradient $\partial u/\partial z|_{i,j,k}$ is located at the center of edge $\mathcal{E}^y_{i,j,k}$ in control volume $\Omega^u_{i,j,k}$ (■). Due to the extrusion in z direction, the line connecting two vectors $u_{i,j,k}$ and $u_{i,j,k+1}$ are always normal to the top face of control volume, $\Gamma^{u,\mathrm{t}}$. So there is no non-orthogonality issue as discussed for the normal velocity gradients, and we can discretize $\partial u/\partial z|_{i,j,k}$ based on the basic MAC method formulas. For this purpose, by differentiating u velocity unknown in z direction, one gets:

$$\left.\frac{\partial u}{\partial z}\right|_{i,j,k} = \frac{u_{i,j,k+1} - u_{i,j,k}}{\frac{1}{2}\,\theta^u_{i,j,k+1}\,\Delta z_{k+1} + \frac{1}{2}\,\theta^u_{i,j,k}\,\Delta z_k}. \tag{3.53}$$

Discretization of the off-diagonal elements: $\partial w/\partial x$ and $\partial w/\partial y$

Discretization of the remaining terms *i.e.* $\partial w/\partial x$ and $\partial w/\partial y$ in z momentum equation (3.2c), is the most intricate part as the orthogonality property is lost for these cases. For example in Figure 3.18, the line connecting th vectors $w_{i,j,k}$ and $w_{i+1,j,k}$ is not perpendicular to the face $\Gamma^{w,\mathrm{e}}_{i,j,k}$ in control volume $\Omega^w_{i,j,k}$. Thus, using the formal two-points formula (3.47) may degrade the

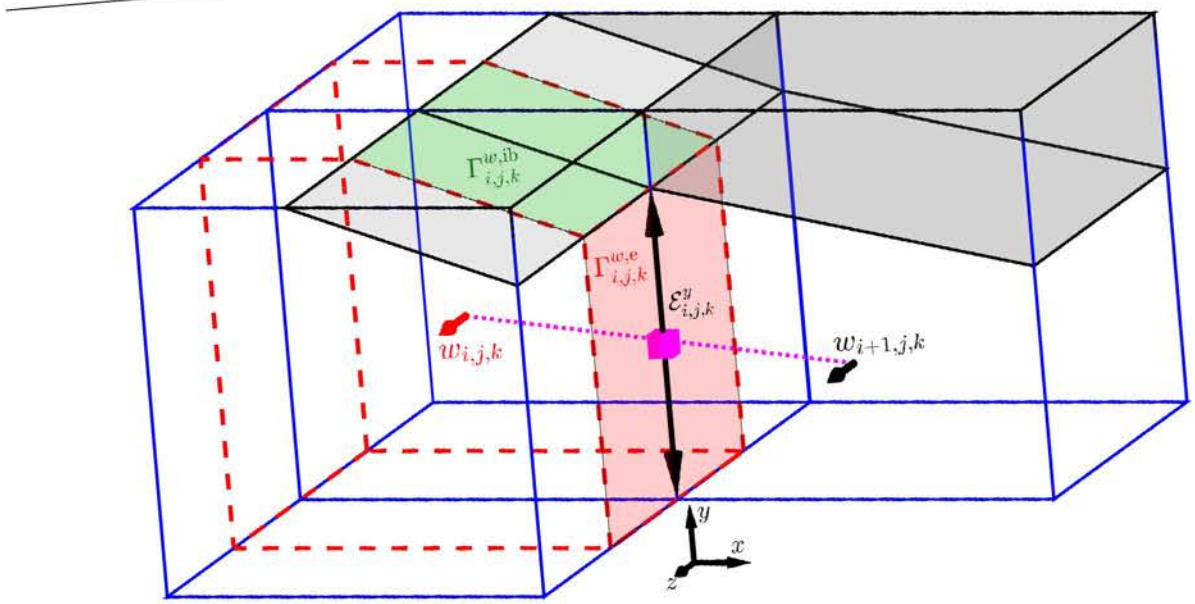


Figure 3.18: Illustration of an arbitrary staggered control volume $\Omega^{w}_{i,j,k} \in \Omega_{i,j,k} \cup \Omega_{i,j,k+1}$ (red dashed line (--)) whose eastern face $\Gamma^{w,\mathrm{e}}_{i,j,k}$ is shaded in red and its solid face $\Gamma^{w,\mathrm{ib}}_{i,j,k}$ is highlighted in green. Velocity gradient $\partial w/\partial x|_{i,j,k}$ (■) is located at the center of edge $\mathcal{E}^{y}_{i,j,k}$. The adjacent velocity vectors, $w_{i,j,k}$ and $w_{i+1,j,k}$, are connected by a dotted line (··) which is not perpendicular to the face $\Gamma^{w,\mathrm{e}}_{i,j,k}$.

second-order accuracy discretization of these terms. Instead, we can employ the discretization suggested by Van der Heiden [99] for $\partial w/\partial x$ in the fashion as done for the pressure gradients in Equation (3.24) since the location of w velocity vector on a 3D extruded mesh (barycenter of the fluid part of cell faces) corresponds to the pressure counterpart in 2D. Therefore, the discrete form of $\partial w/\partial x|_{i,j,k}$ (■) in Figure 3.18 is built such that:

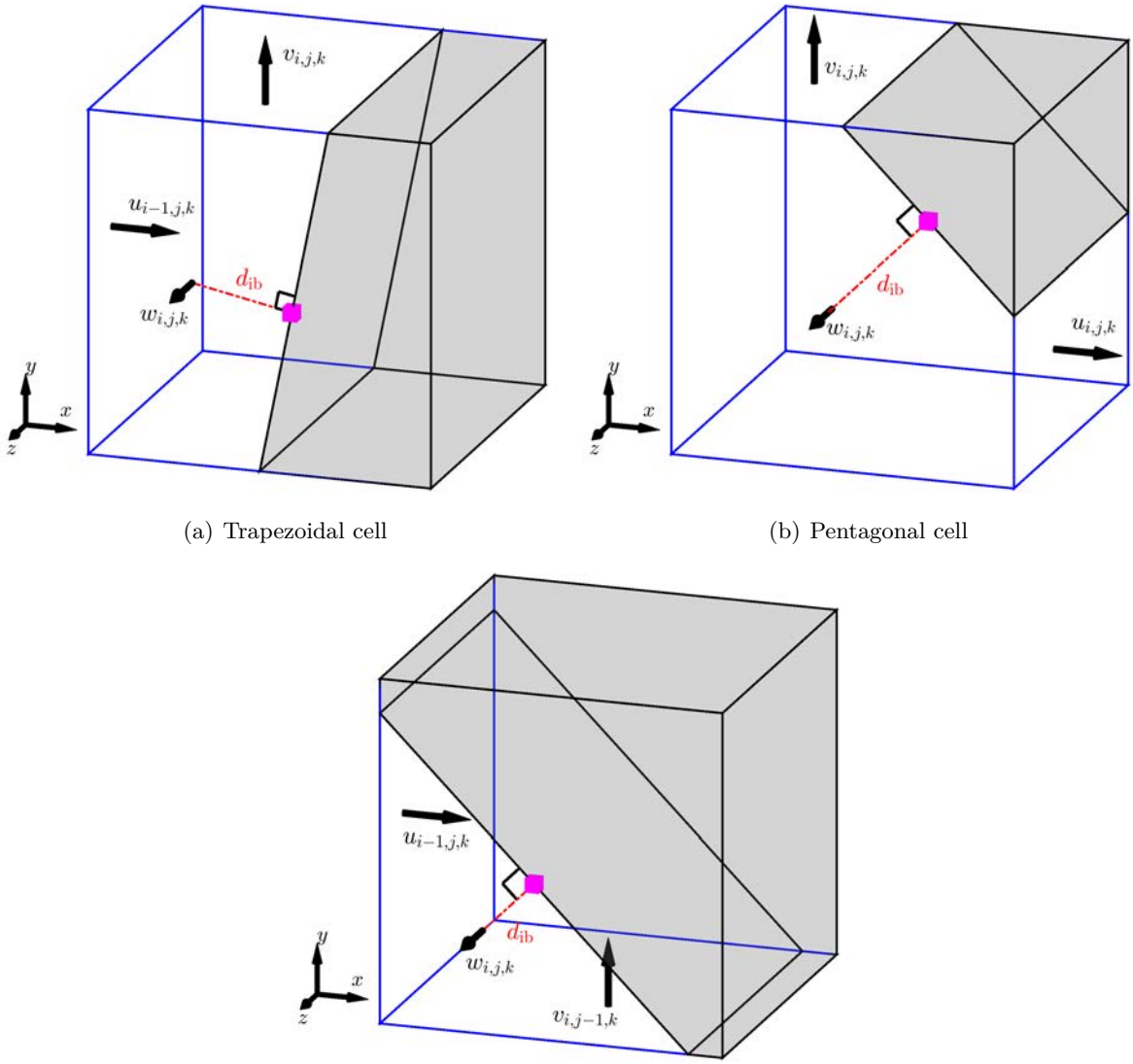
$$\left.\frac{\partial w}{\partial x}\right|_{i,j,k} \cong \frac{\mathcal{E}^{y}_{i,j,k}\,(w_{i+1,j,k} - w_{i,j,k})}{\frac{1}{2}(\theta^{w}_{i+1,j,k}\,\Delta x_{i+1}\,\Delta y_j + \theta^{w}_{i,j,k}\,\Delta x_i\,\Delta y_j)} = \frac{\alpha^{y}_{i,j,k}\,(w_{i+1,j,k} - w_{i,j,k})}{\frac{1}{2}(\theta^{w}_{i+1,j,k}\,\Delta x_{i+1} + \theta^{w}_{i,j,k}\,\Delta x_i)}. \tag{3.54}$$

The performance of this scheme will be assessed in Chapter 4.

In the vicinity of immersed boundary, the line connecting velocity vector w and immersed boundary is not necessarily aligned with x or y axis, see Figure 3.19. Thus, the velocity gradient is computed in direction of the normal unit vector of solid face in the cut-cells in order to keep the orthogonality property. According to Figure 3.19, the wall shear $\partial w/\partial \boldsymbol{n}$ is discretized by employing a one-sided formula as:

$$\left.\frac{\partial w}{\partial \boldsymbol{n}}\right|_{i,j,k} \cong \frac{w^{\mathrm{ib}} - w_{i,j,k}}{d_{\mathrm{ib}}}. \tag{3.55}$$

where w^{ib} is given by the boundary conditions and d_{ib} is the normal distance between the location of $w_{i,j,k}$ (barycenter of open cell face) and the solid face, see Figure 3.19.



(a) Trapezoidal cell

(b) Pentagonal cell

(c) Triangular cell

Figure 3.19: The normal distance d_{ib} between the w velocity location (open cell face barycenter) and the immersed boundary in the case of the wall shear stress $\partial w/\partial \boldsymbol{n}|_{i,j,k}$ (■) for 3 types of cut-cells.

3.4.5 Discretization of viscous fluxes

In LS-STAG method, the viscous fluxes $\boldsymbol{F}^{\text{d}} = (F_u^{\text{d}}, F_v^{\text{d}}, F_w^{\text{d}})$ in the momentum equations are written as:

$$F_u^{\text{d}} = \int_{\Gamma^u} \eta(\dot{\gamma}) \left(\dot{\gamma}_{xx}\, \boldsymbol{e}_x + \dot{\gamma}_{xy}\, \boldsymbol{e}_y + \dot{\gamma}_{xz}\, \boldsymbol{e}_z\right) \cdot \boldsymbol{n}\ \mathrm{d}S, \tag{3.56a}$$

$$F_v^{\text{d}} = \int_{\Gamma^v} \eta(\dot{\gamma}) \left(\dot{\gamma}_{yx}\, \boldsymbol{e}_x + \dot{\gamma}_{yy}\, \boldsymbol{e}_y + \dot{\gamma}_{yz}\, \boldsymbol{e}_z\right) \cdot \boldsymbol{n}\ \mathrm{d}S, \tag{3.56b}$$

$$F_w^{\text{d}} = \int_{\Gamma^w} \eta(\dot{\gamma}) \left(\dot{\gamma}_{zx}\, \boldsymbol{e}_x + \dot{\gamma}_{zy}\, \boldsymbol{e}_y + \dot{\gamma}_{zz}\, \boldsymbol{e}_z\right) \cdot \boldsymbol{n}\ \mathrm{d}S. \tag{3.56c}$$

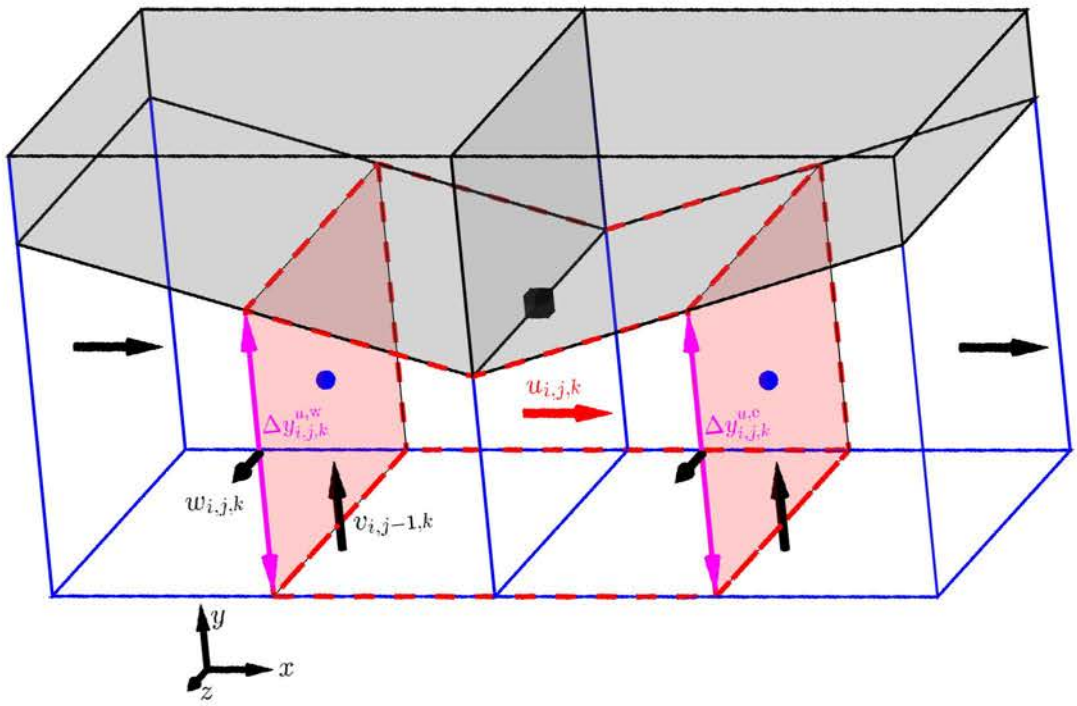


Figure 3.20: Illustration of the control volume $\Omega^u_{i,j,k}$ in the vicinity of two trapezoidal cells (red dashed line (--)). The western and $\Gamma^{u,\mathrm{w}}_{i,j,k}$ ($\Delta y^{u,\mathrm{w}}_{i,j,k} \times \Delta z_k$) and eastern $\Gamma^{u,\mathrm{e}}_{i,j,k}$ ($\Delta y^{u,\mathrm{e}}_{i,j,k} \times \Delta z_k$) faces are shaded in red. The shear stress $\dot{\gamma}_{xy}|_{i,j,k}$ is located at the midpoint of solid edge (■).

The viscous fluxes in Equation (3.56) are split in two different terms: the normal stress fluxes such as $\int_{\Gamma^u} \eta(\dot{\gamma})\, \dot{\gamma}_{xx}\, \boldsymbol{e}_x \cdot \boldsymbol{n}\ \mathrm{d}S$ which include the diagonal elements of velocity gradient tensor and the shear stress fluxes such like $\int_{\Gamma^u} \eta(\dot{\gamma})\, \dot{\gamma}_{xy}\, \boldsymbol{e}_y \cdot \boldsymbol{n}\ \mathrm{d}S$ which include the off-diagonal velocity gradients.

Discretization of normal stress fluxes

In this section, we will focus on the discretization of normal stress terms. A naive approach would be writing those terms as the sum of net fluxes through the faces of control volume, and then discretization of the velocity gradients by using the finite difference approximations such as (3.41). For example, according to the Figure 3.20, $\int_{\Gamma^u_{i,j,k}} \eta(\dot{\gamma})\, \dot{\gamma}_{xx}\, \boldsymbol{e}_x \cdot \boldsymbol{n}\ \mathrm{d}S$ can be expressed as the net fluxes through the eastern $\Gamma^{u,\mathrm{e}}_{i,j,k}$ and western $\Gamma^{u,\mathrm{w}}_{i,j,k}$ faces of control volume $\Omega^u_{i,j,k}$ such that:

$$\begin{aligned}\int_{\Gamma^u_{i,j,k}} \eta(\dot{\gamma})\, \dot{\gamma}_{xx}\, \boldsymbol{e}_x \cdot \boldsymbol{n}\ \mathrm{d}S &= \int_{\Gamma^{u,\mathrm{e}}_{i,j,k}} \eta(\dot{\gamma})\, \dot{\gamma}_{xx}\, \boldsymbol{e}_x \cdot \boldsymbol{n}\ \mathrm{d}S - \int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} \eta(\dot{\gamma})\, \dot{\gamma}_{xx}\, \boldsymbol{e}_x \cdot \boldsymbol{n}\ \mathrm{d}S \\ &\cong 2\,\eta(\dot{\gamma}_{\mathrm{e}})\, \frac{u_{i+1,j,k} - u_{i,j,k}}{\Delta x_{i+1}}\, \Delta z_k\, \Delta y^{u,\mathrm{e}}_{i,j,k} - 2\,\eta(\dot{\gamma}_{\mathrm{w}})\, \frac{u_{i,j,k} - u_{i-1,j,k}}{\Delta x_i}\, \Delta z_k\, \Delta y^{u,\mathrm{w}}_{i,j,k},\end{aligned} \tag{3.57}$$

where the lengths $\Delta y^{u,\mathrm{w}}_{i,j,k}$ and $\Delta y^{u,\mathrm{e}}_{i,j,k}$ are still unknowns and need to be defined. All efforts in this direction leads to failure in terms of numerical accuracy [22] due to the non-orthogonality issue in the cut-cells, discussed in Section 3.4.4. In order to circumvent this issue, the normal stress flux shall be discretized with an expression similar to the pressure gradient (3.24):

$$\int_{\Gamma^u_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xx}\,\boldsymbol{e}_x\cdot\boldsymbol{n}\;\mathrm{d}S \cong \theta^u_{i,j,k}\;\Delta y_j\,\Delta z_k\left(2\,\eta(\dot{\gamma}_\mathrm{e})\,\frac{\partial u}{\partial x}\bigg|_{i+1,j,k} - 2\,\eta(\dot{\gamma}_\mathrm{w})\,\frac{\partial u}{\partial x}\bigg|_{i,j,k}\right), \tag{3.58}$$

where the terms $\partial u/\partial x|_{i+1,j,k}$ and $\partial u/\partial x|_{i,j,k}$ are computed via the generic formula (3.46a) for all types of cells. Note that far away the immersed boundary ($\theta^u = \theta^v = \theta^w = 1$), the standard finite difference formula (3.41) of original MAC method is recovered. Note that similar procedure can be also applied for other normal stress fluxes *i.e.* $\int_{\Gamma^v_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{yy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S$ and $\int_{\Gamma^w_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{zz}\,\boldsymbol{e}_z\cdot\boldsymbol{n}\;\mathrm{d}S$ in control volumes $\Omega^v_{i,j,k}$ and $\Omega^w_{i,j,k}$, respectively.

Discretization of shear stress fluxes

In this section, we will describe only the discretization of shear stress fluxes in Equation (3.56a). Analogous formulas can be also derived for the shear fluxes in y and z directions, (3.56b) and (3.56c). The flux $\int_{\Gamma^u_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S$ is written as the net fluxes through the northern and southern faces of an arbitrary control volume $\Omega^u_{i,j,k}$ such that:

$$\int_{\Gamma^u_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S = \int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S - \int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S. \tag{3.59}$$

If the northern $\Gamma^{u,\mathrm{n}}$ and southern $\Gamma^{u,\mathrm{s}}$ faces of control volume are both fluid *e.g.* $\Omega^u_{i,j,k}$ in Figure 3.15, we can apply the midpoint rule as:

$$\int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S \cong \eta(\dot{\gamma}_\mathrm{n})\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)_{i,j,k}\Delta S^\mathrm{n}, \tag{3.60a}$$

$$\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S \cong \eta(\dot{\gamma}_\mathrm{s})\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)_{i,j-1,k}\Delta S^\mathrm{s}. \tag{3.60b}$$

where ΔS^n and ΔS^s represent respectively the surface areas of northern and southern faces. These areas are determined such that:

$$\Delta S^\mathrm{n} = \frac{1}{2}(\,\theta^v_{i,j,k}\;\Delta x_i + \theta^v_{i+1,j,k}\;\Delta x_{i+1})\,\Delta z_k, \tag{3.61a}$$

$$\Delta S^\mathrm{s} = \frac{1}{2}(\,\theta^v_{i,j-1,k}\;\Delta x_i + \theta^v_{i+1,j-1,k}\;\Delta x_{i+1})\,\Delta z_k, \tag{3.61b}$$

and describe only the fluid parts of faces in order to ensure the local conservation of fluxes across the fluid faces. Note that the velocity gradients are computed based on the formulas given in Section 3.4.4. For the cases in which one of the faces is solid, *e.g.* northern face of control volume $\Omega^u_{i,j,k}$ in Figure 3.20, the shear stress flux through the solid face of $\Gamma^{u,\mathrm{ib}}_{i,j,k}$ reads:

$$\int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} \eta(\dot{\gamma})\,\dot{\gamma}_{xy}\,\boldsymbol{e}_y\cdot\boldsymbol{n}\;\mathrm{d}S \cong \eta(\dot{\gamma}_\mathrm{n})\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)_{i,j,k}\Delta S^\mathrm{ib}, \tag{3.62}$$

where $\Delta S^{\mathrm{ib}} = \frac{1}{2}(\Delta x_i + \Delta x_{i+1})\,\Delta z_k$ describes the projected surface area of solid face along y axis.

Similarly, the term $\int_{\Gamma^u_{i,j,k}} \eta(\dot\gamma)\,\dot\gamma_{xz}\,\boldsymbol{e}_z \cdot \boldsymbol{n}\ \mathrm{d}S$ is expressed as the net flux through top and bottom faces of $\Omega^u_{i,j,k}$ indicated in Figure 3.20 such that:

$$\int_{\Gamma^u_{i,j,k}} \eta(\dot\gamma)\,\dot\gamma_{xz}\,\boldsymbol{e}_z \cdot \boldsymbol{n}\ \mathrm{d}S = \int_{\Gamma^{u,\mathrm{t}}_{i,j,k}} \eta(\dot\gamma)\,\dot\gamma_{xz}\,\boldsymbol{e}_z \cdot \boldsymbol{n}\ \mathrm{d}S - \int_{\Gamma^{u,\mathrm{b}}_{i,j,k}} \eta(\dot\gamma)\,\dot\gamma_{xz}\,\boldsymbol{e}_z \cdot \boldsymbol{n}\ \mathrm{d}S, \tag{3.63}$$

where by applying the midpoint rule, we can get:

$$\int_{\Gamma^{u,\mathrm{t}}_{i,j,k}} \eta(\dot\gamma)\,\dot\gamma_{xz}\,\boldsymbol{e}_z \cdot \boldsymbol{n}\ \mathrm{d}S \cong \eta(\dot\gamma_{\mathrm{t}}) \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)_{i,j,k} \Delta S^{\mathrm{t}}, \tag{3.64a}$$

$$\int_{\Gamma^{u,\mathrm{b}}_{i,j,k}} \eta(\dot\gamma)\,\dot\gamma_{xz}\,\boldsymbol{e}_z \cdot \boldsymbol{n}\ \mathrm{d}S \cong \eta(\dot\gamma_{\mathrm{b}}) \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)_{i,j,k-1} \Delta S^{\mathrm{b}}, \tag{3.64b}$$

in which ΔS^{t} and ΔS^{b} are the integration area of the top and bottom faces of control volume $\Omega^u_{i,j,k}$, respectively. These areas are defined such that:

$$\Delta S^{\mathrm{t}} = \frac{1}{2}(\,\theta^w_{i,j,k}\ \Delta x_i + \theta^w_{i+1,j,k}\ \Delta x_{i+1})\,\Delta y_j, \tag{3.65a}$$

$$\Delta S^{\mathrm{b}} = \frac{1}{2}(\,\theta^w_{i,j,k-1}\ \Delta x_i + \theta^w_{i+1,j,k-1}\ \Delta x_{i+1})\,\Delta y_j. \tag{3.65b}$$

Discrete form of viscous fluxes based on the 7-point stencil scheme

In LS-STAG solver, the viscous terms are discretized in a way that the simplicity of 7-point scheme of the MAC method is preserved for 3D extruded meshes and the property of local conservation of fluxes is verified in the fluid cells. For this purpose, the viscous fluxes are split into the "*implicit*" and "*explicit*" parts such that the implicit part reads:

$$F_u^{\mathrm{d,impl}} = \int_{\Gamma^u_{i,j,k}} \eta(\dot\gamma)\Big[\frac{\partial u}{\partial x}\,\boldsymbol{e}_x + \frac{\partial u}{\partial y}\,\boldsymbol{e}_y + \frac{\partial u}{\partial z}\,\boldsymbol{e}_z\Big] \cdot \boldsymbol{n}\ \mathrm{d}S \tag{3.66a}$$

$$F_v^{\mathrm{d,impl}} = \int_{\Gamma^v_{i,j,k}} \eta(\dot\gamma)\Big[\frac{\partial v}{\partial x}\,\boldsymbol{e}_x + \frac{\partial v}{\partial y}\,\boldsymbol{e}_y + \frac{\partial v}{\partial z}\,\boldsymbol{e}_z\Big] \cdot \boldsymbol{n}\ \mathrm{d}S \tag{3.66b}$$

$$F_w^{\mathrm{d,impl}} = \int_{\Gamma^w_{i,j,k}} \eta(\dot\gamma)\Big[\frac{\partial w}{\partial x}\,\boldsymbol{e}_x + \frac{\partial w}{\partial y}\,\boldsymbol{e}_y + \frac{\partial w}{\partial z}\,\boldsymbol{e}_z\Big] \cdot \boldsymbol{n}\ \mathrm{d}S, \tag{3.66c}$$

and the explicit part reads:

$$F_u^{\mathrm{d,expl}} = \int_{\Gamma^u_{i,j,k}} \eta(\dot\gamma)\Big[\frac{\partial u}{\partial x}\,\boldsymbol{e}_x + \frac{\partial v}{\partial x}\,\boldsymbol{e}_y + \frac{\partial w}{\partial x}\,\boldsymbol{e}_z\Big] \cdot \boldsymbol{n}\ \mathrm{d}S \tag{3.67a}$$

$$F_v^{\mathrm{d,expl}} = \int_{\Gamma^v_{i,j,k}} \eta(\dot\gamma)\Big[\frac{\partial u}{\partial y}\,\boldsymbol{e}_x + \frac{\partial v}{\partial y}\,\boldsymbol{e}_y + \frac{\partial w}{\partial y}\,\boldsymbol{e}_z\Big] \cdot \boldsymbol{n}\ \mathrm{d}S \tag{3.67b}$$

$$F_w^{\mathrm{d,expl}} = \int_{\Gamma^w_{i,j,k}} \eta(\dot\gamma)\Big[\frac{\partial u}{\partial z}\,\boldsymbol{e}_x + \frac{\partial v}{\partial z}\,\boldsymbol{e}_y + \frac{\partial w}{\partial z}\,\boldsymbol{e}_z\Big] \cdot \boldsymbol{n}\ \mathrm{d}S. \tag{3.67c}$$

With such decomposition of viscous fluxes, the computation of velocity unknowns, u, v and w, is completely decoupled. In 3D solver, the explicit parts are computed explicitly at time and

their contribution is added to the RHS of linear system. Note that for the Newtonian fluids ($\eta(\dot{\gamma}) = \eta$), the explicit parts are zero thanks to the divergence theorem and the continuity equation (2.1), $\nabla \cdot (\nabla \boldsymbol{v}^T) = \nabla(\nabla \cdot \boldsymbol{v}) = 0$. We will subsequently consider only the discrete form of the implicit viscous fluxes with a 7-point structure for a control volume far away the immersed boundaries. Accordingly, for the Cartesian control volume $\Omega^u_{i,j,k}$ shown in Figure 3.10, the discretization has a form of:

$$\begin{aligned} F_u^{\mathrm{d,impl}} \cong & \mathcal{K}_{\mathrm{W}}(i,j,k)\, u_{i-1,j,k} \;+\; \mathcal{K}_{\mathrm{E}}(i,j,k)\, u_{i+1,j,k} \\ & + \mathcal{K}_{\mathrm{S}}(i,j,k)\, u_{i,j-1,k} \;+\; \mathcal{K}_{\mathrm{N}}(i,j,k)\, u_{i,j+1,k} \\ & + \mathcal{K}_{\mathrm{B}}(i,j,k)\, u_{i,j,k-1} \;+\; \mathcal{K}_{\mathrm{T}}(i,j,k)\, u_{i,j,k+1} \\ & + \mathcal{K}_{\mathrm{P}}(i,j,k)\, u_{i,j,k} \;+\; S^{\mathrm{ib,v}}_{i,j,k}, \end{aligned} \tag{3.68}$$

with:

$$\mathcal{K}_{\mathrm{W}}(i,j,k) = \eta(\dot{\gamma}_{\mathrm{w}}) \frac{\theta^u_{i,j,k}\, \Delta y_j\, \Delta z_k\, \theta^u_{i-1,j,k}}{V_{i,j,k}/(\Delta y_j\, \Delta z_k)}, \tag{3.69a}$$

$$\mathcal{K}_{\mathrm{E}}(i,j,k) = \eta(\dot{\gamma}_{\mathrm{e}}) \frac{\theta^u_{i,j,k}\, \Delta y_j\, \Delta z_k\, \theta^u_{i+1,j,k}}{V_{i+1,j,k}/(\Delta y_j\, \Delta z_k)}, \tag{3.69b}$$

$$\mathcal{K}_{\mathrm{S}}(i,j,k) = \eta(\dot{\gamma}_{\mathrm{s}}) \frac{(\,\theta^v_{i,j-1,k}\, \Delta x_i \,+\, \theta^v_{i+1,j-1,k}\, \Delta x_{i+1}\,)\, \Delta z_k}{\theta^u_{i,j,k}\, \Delta y_j \,+\, \theta^u_{i,j-1,k}\, \Delta y_{j-1}}, \tag{3.69c}$$

$$\mathcal{K}_{\mathrm{N}}(i,j,k) = \eta(\dot{\gamma}_{\mathrm{n}}) \frac{(\,\theta^v_{i,j,k}\, \Delta x_i \,+\, \theta^v_{i+1,j,k}\, \Delta x_{i+1}\,)\, \Delta z_k}{\theta^u_{i,j,k}\, \Delta y_j \,+\, \theta^u_{i,j+1,k}\, \Delta y_{j+1}}, \tag{3.69d}$$

$$\mathcal{K}_{\mathrm{B}}(i,j,k) = \eta(\dot{\gamma}_{\mathrm{b}}) \frac{(\,\theta^w_{i,j,k-1}\, \Delta x_i \,+\, \theta^w_{i+1,j,k-1}\, \Delta x_{i+1}\,)\, \Delta y_j}{\theta^u_{i,j,k}\, \Delta z_k \,+\, \theta^u_{i,j,k-1}\, \Delta z_{k-1}}, \tag{3.69e}$$

$$\mathcal{K}_{\mathrm{T}}(i,j,k) = \eta(\dot{\gamma}_{\mathrm{t}}) \frac{(\,\theta^w_{i,j,k}\, \Delta x_i \,+\, \theta^w_{i+1,j,k}\, \Delta x_{i+1}\,)\, \Delta y_j}{\theta^u_{i,j,k}\, \Delta z_k \,+\, \theta^u_{i,j,k+1}\, \Delta z_{k+1}}, \tag{3.69f}$$

$$\begin{aligned} \mathcal{K}_{\mathrm{P}}(i,j,k) = & -\eta(\dot{\gamma}_{\mathrm{w}}) \frac{(\,\theta^u_{i,j,k}\,)^2 \Delta y_j\, \Delta z_k}{V_{i,j,k}/(\Delta y_j\, \Delta z_k)} - \eta(\dot{\gamma}_{\mathrm{e}}) \frac{(\,\theta^u_{i,j,k}\,)^2 \Delta y_j\, dz_k}{V_{i+1,j,k}/(\Delta y_j\, \Delta z_k)} \\ & - \mathcal{K}_{\mathrm{S}}(i,j,k) - \mathcal{K}_{\mathrm{N}}(i,j,k) - \mathcal{K}_{\mathrm{B}}(i,j,k) - \mathcal{K}_{\mathrm{T}}(i,j,k), \end{aligned} \tag{3.69g}$$

$$S^{\mathrm{ib,v}}_{i,j,k} = \theta^u_{i,j,k}\; \Delta y_j\, \Delta z_k \left[\eta(\dot{\gamma}_{\mathrm{e}}) \frac{(\,\theta^u_{i,j,k} \,-\, \theta^u_{i+1,j,k}\,)}{V_{i+1,j,k}/\Delta y_j\, \Delta z_k}\, u^{\mathrm{ib}}_{i+1,j,k} - \eta(\dot{\gamma}_{\mathrm{w}}) \frac{(\,\theta^u_{,i-1,j}\, k - \theta^u_{i,j,k}\,)}{V_{i,j,k}/\Delta y_j\, \Delta z_k}\, u^{\mathrm{ib}}_{i,j,k} \right], \tag{3.69h}$$

which recovers the MAC method symmetric discretization, *e.g.* $\mathcal{K}_{\mathrm{W}}(i+1,j,k) = \mathcal{K}_{\mathrm{E}}(i,j,k)$. Next to the immersed boundaries, some coefficients have been determined independently based on the combination of cut-cells that forms the control volume. For example, in the case of the

control volume $\Omega^u_{i,j,k}$ indicated in Figure 3.20, the following modifications:

$$\mathcal{K}_{\rm N}(i,j,k) = 0 \tag{3.70a}$$

$$\begin{aligned}\mathcal{K}_{\rm P}(i,j,k) = &-\eta(\dot\gamma_{\rm w})\frac{(\,\theta^u_{i,j,k}\,)^2\Delta y_j\,\Delta z_k}{V_{i,j,k}/(\Delta y_j\,\Delta z_k)} - \eta(\dot\gamma_{\rm e})\frac{(\,\theta^u_{i,j,k}\,)^2\Delta y_j\,dz_k}{V_{i+1,j,k}/(\Delta y_j\,\Delta z_k)}\\ &- \mathcal{K}_{\rm S}(i,j,k) - \eta(\dot\gamma_{\rm n})\frac{(\Delta x_i + \Delta x_{i+1})\,\Delta z_k}{\theta^u_{i,j,k}\,\Delta y_j} - \mathcal{K}_{\rm B}(i,j,k) - \mathcal{K}_{\rm T}(i,j,k),\end{aligned} \tag{3.70b}$$

$$\begin{aligned}S^{\rm ib,v}_{i,j,k} = \;& \theta^u_{i,j,k}\;\Delta y_j\,\Delta z_k\left[\eta(\dot\gamma_{\rm e})\frac{(\,\theta^u_{i,j,k}\,-\,\theta^u_{i+1,j,k}\,)}{V_{i+1,j,k}/\Delta y_j\,\Delta z_k}\,u^{\rm ib}_{i+1,j,k} - \eta(\dot\gamma_{\rm w})\frac{(\,\theta^u_{,i-1,j}\,k\,-\,\theta^u_{i,j,k}\,)}{V_{i,j,k}/\Delta y_j\,\Delta z_k}\,u^{\rm ib}_{i,j,k}\right]\\ &+\eta(\dot\gamma_{\rm n})\frac{(\Delta x_i + \Delta x_{i+1})\,\Delta z_k}{\theta^u_{i,j,k}\,\Delta y_j}\,u(x_i, y^{\rm ib}_{i,j,k}, z_k),\end{aligned} \tag{3.70c}$$

have been applied to the coefficients due to the presence of solid part at the northern face. This kind of discretization ensures the conservation of total momentum in its discrete form on the LS-STAG mesh, refer to Section 2.2.2.

3.5 Extension of LS-STAG method to non-Newtonian fluids

In Chapter 1, we have introduced the governing equations for the non-Newtonian fluid flows consisting of conservation equation of momentum (1.1) and continuity equation (1.2) which would be completed by the constitutive law based upon the fluid type, see Section 1.3. In the following, we will explain in detail the discretization of non-Newtonian terms in the momentum equations. As observed in the momentum equation (3.2), the non-Newtonian terms are involved only in the viscous terms. The discrete form of viscous fluxes are studied in Section 3.4.5. In non-Newtonian flows, the discretization of these fluxes is completed by computing the shear-rates which are required for evaluating the local viscosities. For this purpose, the shear-rate has to be computed at two distinct locations on LS-STAG mesh, namely at the cell centroids, *e.g.* $\dot\gamma_{\rm w}$ in Equation (3.58), and at the middle of edges, *e.g.* $\dot\gamma_{\rm n}$ in Equation (3.60a).

3.5.1 Interpolation of the shear-rates at the centroids

According to the control volume $\Omega^u_{i,j,k}$ shown in Figure 3.21, the shear-rate $\dot\gamma_{\rm w}$ in Equation (3.58) is located at the centroid of *cut-cell* $\Omega_{i,j,k}$. At this location, $\dot\gamma_{\rm w}$ is discretized as the diagonal elements of tensor $\dot{\boldsymbol{\gamma}}$ at the centroid of cell $\Omega_{i,j,k}$:

$$\dot\gamma_{\rm w} = \sqrt{[\dot\gamma_{xx}]^2_{i,j,k} + [\dot\gamma_{yy}]^2_{i,j,k} + [\dot\gamma_{zz}]^2_{i,j,k} + \left[\overline{\dot\gamma}_{xy}\right]^2_{i,j,k} + [\overline{\dot\gamma}_{xz}]^2_{i,j,k} + \left[\overline{\dot\gamma}_{yz}\right]^2_{i,j,k}}, \tag{3.71}$$

In contrary to diagonal terms which are located at the cells centroids, off-diagonal terms need to be interpolated from data at the surrounding edges owing to their distinct locations compared

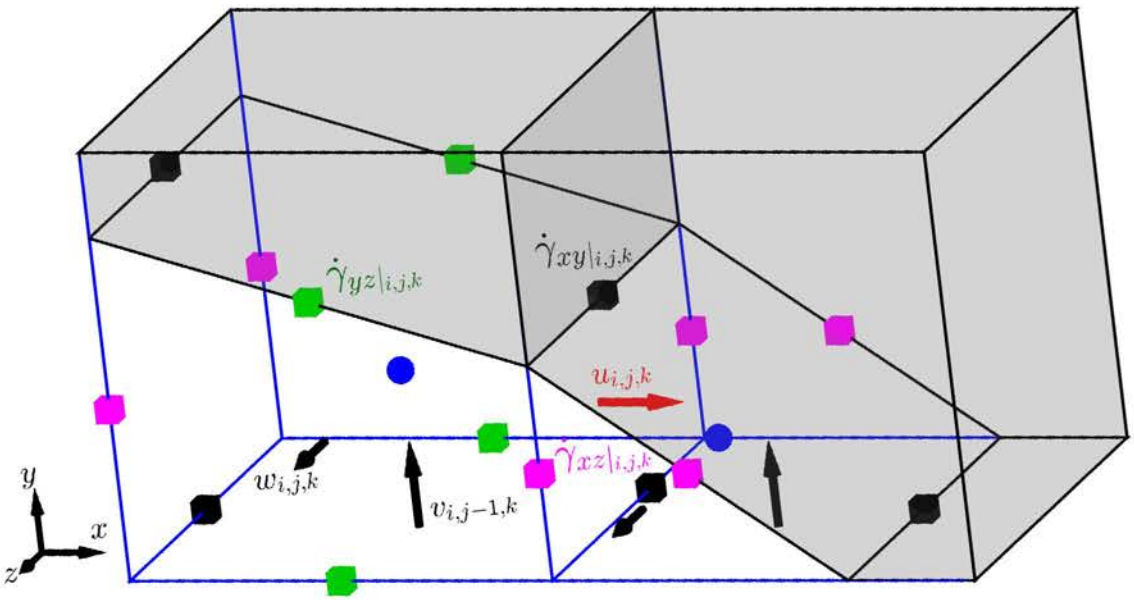


Figure 3.21: Staggered arrangement of the shear-rates on the LS-STAG mesh: the diagonal terms ($\dot{\gamma}_{xx}$, $\dot{\gamma}_{yy}$ and $\dot{\gamma}_{zz}$) at the cell centroids (●) and the off-diagonal ones at midpoints of the edges $\dot{\gamma}_{xy}$ (■), $\dot{\gamma}_{xz}$, (■) and $\dot{\gamma}_{yz}$ (■).

to diagonal ones. This interpolation is carried out with an arithmetic average denoted with $[\cdot]$ based upon the type of cut-cell. For example, the averaged shear-rate $[\dot{\gamma}_{xy}]_{i,j,k}$ in the trapezoidal cell $\Omega_{i,j,k}$ of Figure 3.21, is given by:

$$\left[\dot{\gamma}_{xy}\right]_{i,j,k} = \frac{1}{4}\,\dot{\gamma}_{xy}|_{i,j,k} + \frac{1}{4}\,\dot{\gamma}_{xy}|_{i-1,j,k} + \frac{1}{4}\,\dot{\gamma}_{xy}|_{i-1,j-1,k} + \frac{1}{4}\,\dot{\gamma}_{xy}|_{i,j-1,k}\,. \tag{3.72}$$

The same expression is valid for the Cartesian and pentagonal cells. According to Figure 3.21, in the special case of triangular cut-cells, Equation (3.72) is not valid anymore since only 3 adjacent shear-rates $\dot{\gamma}_{xy}$ terms are involved in $\left[\dot{\gamma}_{xy}\right]_{i+1,j,k}$. Therefore, $\left[\dot{\gamma}_{xy}\right]_{i+1,j,k}$ is determined by a 3-point mean value as:

$$\left[\dot{\gamma}_{xy}\right]_{i+1,j,k} = \frac{1}{3}\,\dot{\gamma}_{xy}|_{i,j,k} + \frac{1}{3}\,\dot{\gamma}_{xy}|_{i,j-1,k} + \frac{1}{3}\,\dot{\gamma}_{xy}|_{i+1,j-1,k}\,. \tag{3.73}$$

It is noteworthy to mention that for the remaining averaged shear-rates in Equation (3.71), *i.e.* $[\dot{\gamma}_{xz}]_{i,j,k}$ and $[\dot{\gamma}_{yz}]_{i,j,k}$, as 4 surrounding shear-rates are always involved, analogous formulas to Equation (3.72), are still valid for all types of cut-cells, even for the triangular ones. For instance in triangular cut-cell $\Omega_{i+1,j,k}$, the mean shear-rate $[\dot{\gamma}_{xz}]_{i+1,j,k}$ is computed such like:

$$[\dot{\gamma}_{xz}]_{i+1,j,k} = \frac{1}{4}\,\dot{\gamma}_{xz}|_{i,j,k} + \frac{1}{4}\,\dot{\gamma}_{xz}|_{i+1,j,k} + \frac{1}{4}\,\dot{\gamma}_{xz}|_{i,j,k-1} + \frac{1}{4}\,\dot{\gamma}_{xz}|_{i+1,j,k-1}\,. \tag{3.74}$$

3.5.2 Interpolation of the shear-rates on edges

We will now determine the shear-rate $\dot{\gamma}_{\mathrm{n}}$ in Equation (3.60a) located on the edge midpoint at face $\Gamma^{u,\mathrm{ib}}$, the same location as $\dot{\gamma}_{xy}|_{i,j,k}$ in Figure 3.21. This term is obtained by using the volume-weighted interpolation the shear-rates computed at the cells centroids (see Section 3.5.1) as:

$$\dot{\gamma}_{\mathrm{n}} = \Big([\beta\, V\, \dot{\gamma}]_{i,j,k} + [\beta\, V\, \dot{\gamma}]_{i+1,j,k} + [\beta\, V\, \dot{\gamma}]_{i,j+1,k} + [\beta\, V\, \dot{\gamma}]_{i+1,j+1,k} \Big) / \widetilde{V}_{i,j,k} \tag{3.75}$$

where $\beta_{i,j,k}$ represents the weight and $\widetilde{V}_{i,j,k}$ is the mean volume as:

$$\widetilde{V}_{i,j,k} = \beta_{i,j,k}\, V_{i,j,k} + \beta_{i+1,j,k}\, V_{i+1,j,k} + \beta_{i,j+1,k}\, V_{i,j+1,k} + \beta_{i+1,j+1,k}\, V_{i+1,j+1,k}. \tag{3.76}$$

The weights are defined based on the type of cell. These weights are summarized in Table 3.2 for different types of cells. Note that these interpolation weights have been defined in [17] for

Table 3.2: The interpolation weights $\beta_{i,j,k}$ for the generic types of cells in 3D extruded geometries.

Cell type	$\beta_{i,j,k}$
Cartesian, Trapezoidal, Pentagonal	1/4
Triangular	1/3
Solid	0

deriving a conservative discretization of the elastic shear stresses, and then successfully employed in [15] for 2D non-Newtonian computations.

By knowing the interpolation weights, we can readily compute $\dot{\gamma}_{\mathrm{n}}$ located at the face $\Gamma^{u,\mathrm{ib}}$ in Figure 3.21. For this case, the contribution of the solid cells *i.e.* $\Omega_{i,j+1,k}$ and $\Omega_{i+1,j+1,k}$, is discarded. Thus $\dot{\gamma}_{\mathrm{n}}$ is written as:

$$\dot{\gamma}_{\mathrm{n}} = \Big(\frac{1}{4} [V\, \dot{\gamma}]_{i,j,k} + \frac{1}{3} [V\, \dot{\gamma}]_{i+1,j,k} + 0 + 0 \Big) / \widetilde{V}_{i,j,k}, \tag{3.77}$$

where the mean volume $\widetilde{V}_{i,j,k}$ is:

$$\widetilde{V}_{i,j,k} = \frac{1}{4} V_{i,j,k} + \frac{1}{3} V_{i+1,j,k} + 0 + 0. \tag{3.78}$$

3.6 LS-STAG time stepping method and solution of linear systems

The time-advancement of the Navier-Stokes equations is carried out by means of the IMEX[1] fractional-step method. The IMEX idea is to combine an implicit scheme for the stiff components with an explicit scheme for the non-stiff ones such that the overall discretization method has the desired stability and accuracy properties without introducing severe time-step restriction. In

[1] IMplicit-EXplicit

this semi-implicit method, the momentum and Poisson equations are solved separately at each time-step. For the momentum equations, the calculation cycle split into two steps: a provisional velocity field ($\widetilde{U}$) calculation step at time $t^{n+1} = (n+1)\,\Delta t$, thereafter follows by a velocity correction step.

In our computations, the unsteady Newtonian flows are performed with the second-order Adams-Bashforth scheme (AB/BDF2) such that the provisional velocity field ($\widetilde{U}$) calculated at time t^{n+1}:

$$\rho\,\mathcal{M}\,\frac{3\widetilde{U} - 4U^n + U^{n-1}}{2\Delta t} + 2\,\rho\,\mathcal{C}[\overline{U}^n]\,U^n - \rho\,\mathcal{C}[\overline{U}^{n-1}]\,U^{n-1} - \mathcal{D}^{\mathrm{T}}\,P^n - \mathcal{K}\,\widetilde{U} = 0, \tag{3.79}$$

is corrected by:

$$\frac{3}{2}\rho\,\mathcal{M}\frac{U^{n+1} - \widetilde{U}}{\Delta t} - \mathcal{D}^{\mathrm{T}}\left(P^{n+1} - P^n\right) = 0, \tag{3.80a}$$

$$\mathcal{D}\,U^{n+1} + \overline{U}^{\mathrm{ib},n+1} = 0. \tag{3.80b}$$

For the steady non-Newtonian flows, the first-order time-stepping with Euler schemes is used until the steady-state solution is obtained. As explained in Section 3.4.5, in non-Newtonian flows, the viscous fluxes are divided into the implicit and explicit parts to fully decouple the computation of provisional velocity components, $\widetilde{u}$, $\widetilde{v}$ and $\widetilde{w}$. For these flows, the provisional velocity field ($\widetilde{U}$) is calculated such that:

$$\rho\,\mathcal{M}\frac{\widetilde{U} - U^n}{\Delta t} + \rho\,\mathcal{C}[\overline{U}^n]\,U^n - \mathcal{D}^{\mathrm{T}}\,P^n - \mathcal{K}\,\widetilde{U} + S^{\mathrm{v},n} = 0, \tag{3.81}$$

in which $S^{\mathrm{v},n}$ represents the explicit part of viscous terms whose contribution is neglected in Newtonian fluids, refer to Section 3.4.5. These fluxes, as well as the shear-rates involved in implicit part of viscous terms ($\mathcal{K}$), are computed explicitly at time t^n. Subsequently, the velocity correction step is applied:

$$\rho\,\mathcal{M}\,\frac{U^{n+1} - \widetilde{U}}{\Delta t} - \mathcal{D}^{\mathrm{T}}\left(P^{n+1} - P^n\right) = 0, \tag{3.82a}$$

$$\mathcal{D}U^{n+1} + \overline{U}^{\mathrm{ib},n+1} = 0. \tag{3.82b}$$

These correction steps (3.80 and 3.82) lead to solving a Poisson equation for the pressure potential $\Phi = 2\,\Delta t\,(P^{n+1} - P^n)/3\,\rho$ in Newtonian fluids and $\Phi = \Delta t\,(P^{n+1} - P^n)/\rho$ in non-Newtonian ones as:

$$\mathcal{A}\,\Phi = \mathcal{D}\,\widetilde{U} + \overline{U}^{\mathrm{ib},n+1}, \qquad \mathcal{A} \equiv -\mathcal{D}\,\mathcal{M}^{-1}\,\mathcal{D}^{\mathrm{T}}, \tag{3.83}$$

which is a symmetric linear system whose 7-point stencil reads:

$$A_{\mathrm{E}}(i,j,k) = \frac{\left(\theta^u_{i,j,k}\Delta y_j \Delta z_k\right)^2}{\frac{1}{2}V_{i,j,k} + \frac{1}{2}V_{i+1,j,k}}, \qquad A_{\mathrm{W}}(i,j,k) = A_{\mathrm{E}}(i-1,j,k), \tag{3.84a}$$

$$A_{\mathrm{N}}(i,j,k) = \frac{\left(\theta^v_{i,j,k}\Delta x_i \Delta z_k\right)^2}{\frac{1}{2}V_{i,j,k} + \frac{1}{2}V_{i,j+1,k}}, \qquad A_{\mathrm{S}}(i,j,k) = A_{\mathrm{N}}(i,j-1,k), \tag{3.84b}$$

$$A_{\mathrm{T}}(i,j,k) = \frac{\left(\theta^w_{i,j,k}\Delta x_i \Delta y_j\right)^2}{\frac{1}{2}V_{i,j,k} + \frac{1}{2}V_{i,j,k+1}}, \qquad A_{\mathrm{B}}(i,j,k) = A_{\mathrm{T}}(i,j,k-1), \tag{3.84c}$$

$$A_{\mathrm{P}}(i,j,k) = -A_{\mathrm{E}}(i,j,k) - A_{\mathrm{W}}(i,j,k) - A_{\mathrm{N}}(i,j,k) - A_{\mathrm{S}}(i,j,k) - A_{\mathrm{T}}(i,j,k) - A_{\mathrm{B}}(i,j,k). \tag{3.84d}$$

The usual pressure equation of the MAC method is recovered in the case of a Cartesian fluid cell. It is noteworthy to mention that in LS-STAG solver, the pressure equation (3.83) is solved in the whole computational domain including the solid cells as for this type of cells, the linear system reads:

$$0 \times \Phi_{i,j,k} = 0, \tag{3.85}$$

which is due to the fact that the pressure is defined up to an additive constant. This indetermination in actual computations is circumvented by adding a small real constant δ in the order of the machine roundoff to the diagonal coefficient $A_{\mathrm{P}}(i,j,k)$. In addition, a standard solver for elliptic equations on Cartesian grids is implemented to solve the Poisson equation (3.83) in the whole computational domain.

In 2D computations simulated in [22], the black-box multigrid/BiCGSTAB solver of Van Kan *et al.* [100] is employed with no modification due to presence of the immersed boundary, and without a significant loss in performance of the solver compared to Cartesian computations. However, In 3D extruded geometries, the LS-STAG code has been developed in an MPI parallel environment and the iterative solution of linear systems, most notably the Poisson pressure equation are solved with the multigrid and Krylov solvers of the HYPRE[1] library [34]. The parallel implementation of LS-STAG solver and the results on its scalability properties will be presented in Chapter 4.

3.7 Conclusion

In this chapter, we have presented the discretization of LS-STAG method for the non-Newtonian fluid flows in the 3D extruded geometries. As in the 2D version, the 3D extruded discretization is designed to strictly fulfill the global conservation laws discussed in Section 2.2.2. Discretization of the continuity equation and the pressure gradients are carried out based on the straightforward extension of their 2D counterparts whereby the generic formulas have been acquired for their

[1] High Performance Preconditioners

discrete forms valid for all cell types. In contrast, we could not obtain the unique expressions for the convective and diffusive terms and their discretization have been constructed in the half control volumes. For the convective fluxes, the skew-symmetric condition has been verified by using the central interpolation with equal weighting. In order to retain the 7-point structure of stencil in 3D, the diffusive fluxes have been split into implicit and explicit parts. Another benefit of such decomposition is that the computation of the provisional velocities are totally decoupled in the fractional-step time scheme. In discretization of diffusive terms, the major issue occurs for the off-diagonal elements $\partial w/\partial x$ and $\partial w/\partial y$ in velocity gradient tensor as the orthogonality property of the MAC mesh is lost. In this manuscript, we have employed the approach proposed by Van der Heiden [99]. Although, discretization in this manner can preserve the 7-point stencil, but we will observe later that it is not yet as mature as the discretization of other terms in the Navier-Stokes equations. In fact, using a standard two-point finite-difference formula would be inaccurate unless a non-orthogonal correction is taken into account [70]. The extension of LS-STAG for the non-Newtonian flows has been achieved by a volume-weighted interpolation of the shear-rates at the edges. The time-advancement of the Navier-Stokes equations have been treated based on the IMEX fractional-step method. For the steady computations reported in the this manuscript, first-order time stepping with Euler schemes is used until the solution reaches steady-state. Unsteady computations of Newtonian flows have been performed with second-order method of BDF2 time-stepping with Adams-Bashforth extrapolations.

Chapter 4

Numerical Validation of LS-STAG method

Contents

4.1 Parallelization of the LS-STAG solver 83
4.1.1 Domain decomposition for Cartesian grids 83
4.1.2 Scalability of LS-STAG solver 87
4.2 Accuracy study of the LS-STAG method 90
4.3 Computation of flows around a circular cylinder 95
4.3.1 Benchmark computations of laminar flow past a circular cylinder in a confined square duct 96
4.3.2 Weakly turbulent flow at $Re_D = 300$ 108

4.1 Parallelization of the LS-STAG solver

The LS-STAG flow solver is a *MPI*[1] code basically written in FORTRAN-90. The MPI is a library specification for message-passing which offers standardization, portability, performance, and rich functionality in a number of high quality implementations. It is a standard *API*[2] that can be used to create parallel applications. We will now focus on domain decomposition strategies, solving linear systems in parallel version of LS-STAG solver and evaluation of this solver in terms of the speed-up and efficiency in parallel computations.

4.1.1 Domain decomposition for Cartesian grids

The first step in designing a parallel algorithm is decomposition of the physical domain into the sets of sub-domains so that each set of sub-domain will be handled by a different processor. Then, the sub-domains are mapped to different processors such that each processor works

[1] Message Passing Interface
[2] Application Programming Interface

only on the sub-domain that is assigned to it. The processes may also need to communicate in order to exchange data. The way that these processes are connected to each other is often called *topology* of the computer. A physical topology represents that connections between the cores, chips, and nodes in the hardware whereas a virtual topology represents the way that MPI processes communicate. The description of virtual topology depends only on the application, and is machine-independent. It can be also exploited by the system in assigning of processes to physical processors. Essentially, MPI provides the users two types of topologies; those based on a *Cartesian* grid, and *graph*-based topologies. The Cartesian grid-based of LS-STAG allows us to implement the Cartesian topology into this method. For this purpose, we first create a new communicator with the Cartesian topology in an existing MPI environment. Knowing the layout of the Cartesian grid and the relationship between the coordinates and their corresponding process ranks is a key element for communication between the processors. Thus, a grid of processes is subsequently described with a Cartesian topology such that each process can be identified by Cartesian coordinates. In the next step, the domain is partitioned which is an important aspect in the domain decomposition approach. In a Cartesian topology, the partitioning strategy can vary from one-dimensional to three-dimensional. In the one-dimensional (also known as slab) decomposition, the message exchange proceeds only in one direction. For the two (pencil) and three-dimensional (cube) types of decomposition, the algorithm of data exchange occurs in two and three coordinate axis, respectively. In LS-STAG method, the domain decomposition is performed by partitioning the domain in all three coordinate directions based on the domain size and number of processors. However, users can choose to partition the domain only in one or two coordinates based on the physics of the model. Note that for three-dimensional decomposition, the load balancing among the processor cores is automatically chosen by MPI.

A 2D Cartesian decomposition of a domain for non-periodic boundary condition is schematically indicated in Figure 4.1. In this figure, the domain is decomposed along x and y directions resulting in 12 sub-domains. The position of each sub-domain is labeled by a coordinate tuple. The rank of processor (ID number) increases first along the x direction, and then the y direction. Note that the periodic boundary condition in each direction allows the first and last sub-domains to participate in the communication. For example in a domain with periodic boundary condition in x direction, the extra communications between the sub-domains are given such that:

$$(0,0,0) \longleftrightarrow (3,0,0), \quad (0,1,0) \longleftrightarrow (3,1,0), \quad (0,2,0) \longleftrightarrow (3,2,0).$$

Each sub-domain in Figure 4.1 is surrounded by special boundary cells subsequently called the *ghost cells*. These additional overlapping cells contain the data that are duplicated from subset of neighbor cells. By allocating these series of ghost cells around the borders of sub-domains, the 7-points stencil scheme can be achieved. Each layer is updated at each time-step by using border exchange communication MPI routines dedicated to virtual Cartesian topologies. In LS-STAG solver, the velocity components, $\boldsymbol{v} = (u, v, w)$, and pressure p in the border of

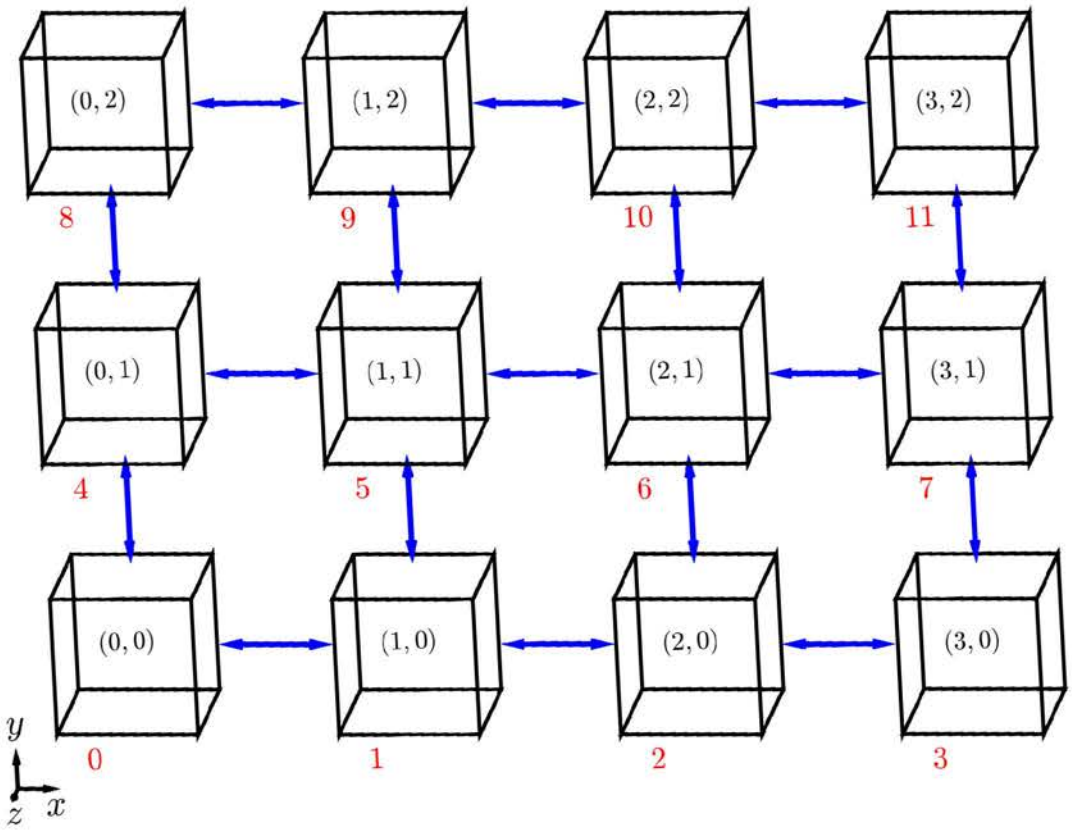


Figure 4.1: A 2D Cartesian decomposition for a non-periodic domain with the array indexing of sub-domains. The sub-domains are handled by the processors with distinct ranks starting from 0.

each sub-domain are the data that must be exchanged by each sub-domain. To illustrate the procedure of data exchange, let the velocity gradient $\partial v/\partial x|_{i,j,k}$ (■) locate at the midpoint of edge on the border of a sub-domain in processor 1, see Figure 4.2. To compute the gradient in processor 1, the value of velocity component $v_{i+1,j,k}$ from its adjacent processors (2) is required. The velocity vector $v_{i+1,j,k}$ is calculated on processor 2 and sent to the ghost cell $\Omega'_{i+1,j,k}$ in processor 1. On the other hand, the variables computed in cell $\Omega_{i,j,k}$ (processor 1) are transfered to processor 2 and stored in $\Omega'_{i,j,k}$. The data exchange process is done by employing the "point-to-point" communication which is basically composed of send and receive operations. Note that communication between the velocity components are performed twice per time iteration, once in calculation of provisional velocity fields and thereafter in correction step. For non-Newtonian flows, only the averaged shear-rates at the centroids (see Section 3.5.1) of each cell need to be exchanged. In the case of mean shear-rates on the edges, the volume-weighted interpolation discussed in Section 3.5.2, can be employed separately in each processors.

Solving the linear systems, especially the discrete Poisson pressure equation, is relied on

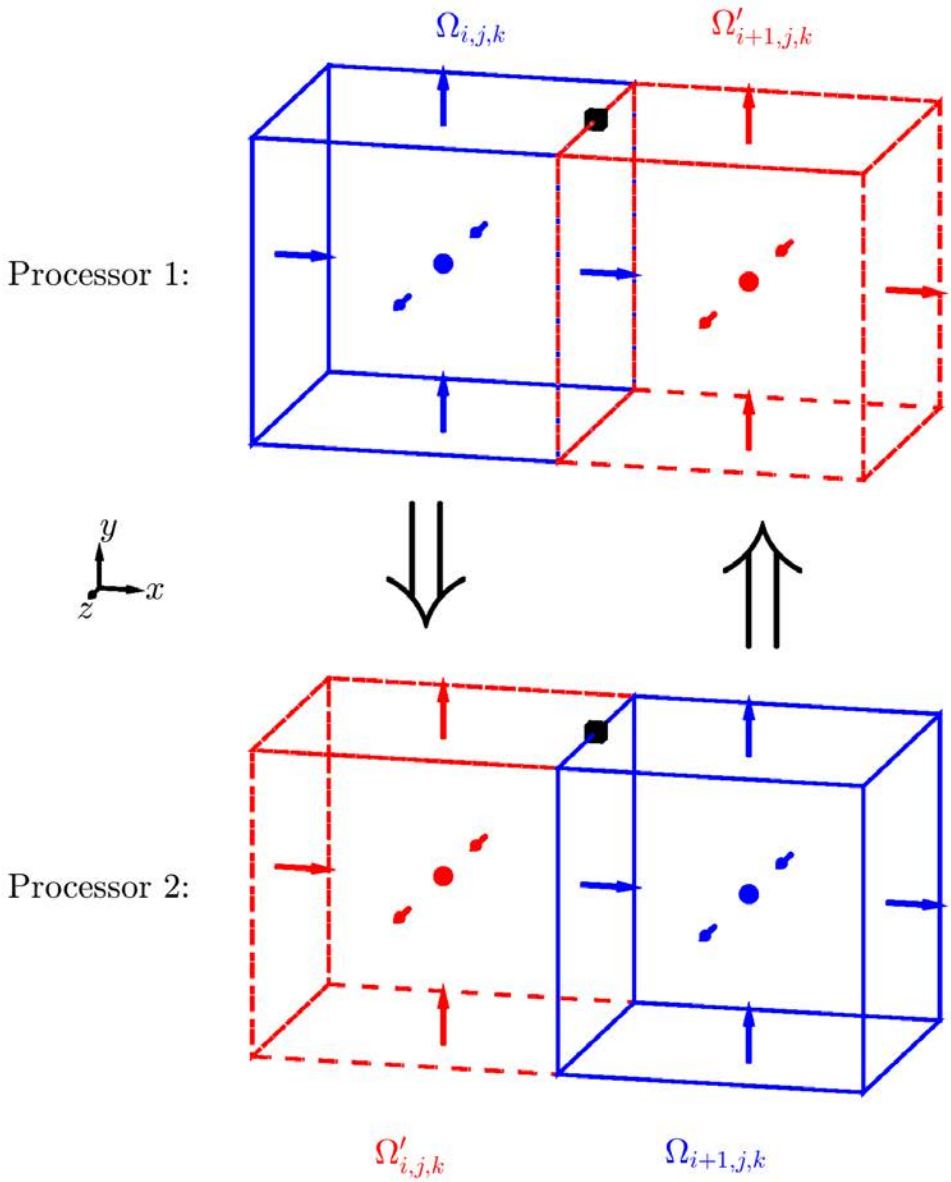


Figure 4.2: The communication between two successive processors. The velocity components in cells $\Omega_{i,j,k}$ and $\Omega_{i+1,j,k}$ related to two different processors (1 and 2), are respectively stored in overlapping cells $\Omega'_{i,j,k}$ and $\Omega'_{i+1,j,k}$ via MPI "point-to-point" communication routines. The data in blue cells are computed on current processor while those in red (overlapping) cells are retrieved from the neighbor processors.

HYPRE which is a library of high performance preconditioners and solvers featuring multigrid methods for the solution of large, sparse linear systems of equations on massively parallel computers. HYPRE builds the sparse matrix structures through various conceptual interfaces such as the structured-grid and semi-structured-grid interfaces [33]. The structured-grid (Struct)

interface is appropriate for scalar applications on structured grids with a fixed stencil pattern (7-point in 3D extruded LS-STAG) of non-zeros at each grid point. The most specific interface that matches our application is Struct interface which allows us to use specialized and more efficient solvers and preconditioners, *i.e.* SMG and PFMG [33], without losing access to more general solvers.

4.1.2 Scalability of LS-STAG solver

The parallel performance of a solver is evaluated by applying the *parallel scaling*. Parallel scaling measures the reduction in wall-clock time as the number of processors used for the calculation, increases. Two common measurements of parallel scaling are:

- "Weak scaling" where for a given computational work per processor, the time is measured as the problem size increases.
- "Strong scaling" in which the time is measured for a fixed problem size as the number of processors increases.

The scalability of LS-STAG solver is assessed by simulation of the 3D periodic and transitional Taylor-Green vortex flow which is an unsteady flow of a decaying vortex. The initial analytical flow field is also given by:

$$u = U_{\text{ref}}\, sin(\frac{x\,\pi}{D_{\text{ref}}})\, cos(\frac{y\,\pi}{D_{\text{ref}}})\, cos(\frac{z\,\pi}{D_{\text{ref}}}), \tag{4.1a}$$

$$v = -\,U_{\text{ref}}\, cos(\frac{x\,\pi}{D_{\text{ref}}})\, sin(\frac{y\,\pi}{D_{\text{ref}}})\, cos(\frac{z\,\pi}{D_{\text{ref}}}), \tag{4.1b}$$

$$w = 0, \tag{4.1c}$$

where $U_{\text{ref}} = 1.6\,m/s$ and $D_{\text{ref}} = 1\,m$ are the initial reference velocity and the domain reference length, respectively. The flow is computed within a cubic domain such that $-1 \leq x, y, z \leq 1$ with the periodic boundary conditions applied to all boundary surfaces. The cubic domain contains a smooth initial distribution of vorticity, see Figure 4.3. As we advance in time, the vortices roll-up, stretch and interact, eventually this flows transitions to turbulence with the creation of small scales. Both scaling measurements are conducted on the cluster of *Institut Jean Lamour* (Université de Lorraine) with an infiniband network and the nodes with 12 processors (Intel®-Xeon E5 X5650-2.66 Ghz). In addition, the linear systems are efficiently solved with BiCGSTAG solver and a constant time-step $\Delta t = 0.01$ is chosen for all simulations.

Figure 4.4 shows the efficiency of LS-STAG solver based on the weak scaling measurement. In this case, the local grid consists $64 \times 64 \times 64$ cells per each processor. The efficiency is described as the simulation wall-time with 1 processor ($WT(1)$) to the wall-time with n processors ($WT(n)$):

$$\frac{WT(1)}{WT(n)} \times 100.$$

Note that for the non-Newtonian cases, a fluid with the power-law index $n = 1$ is used in

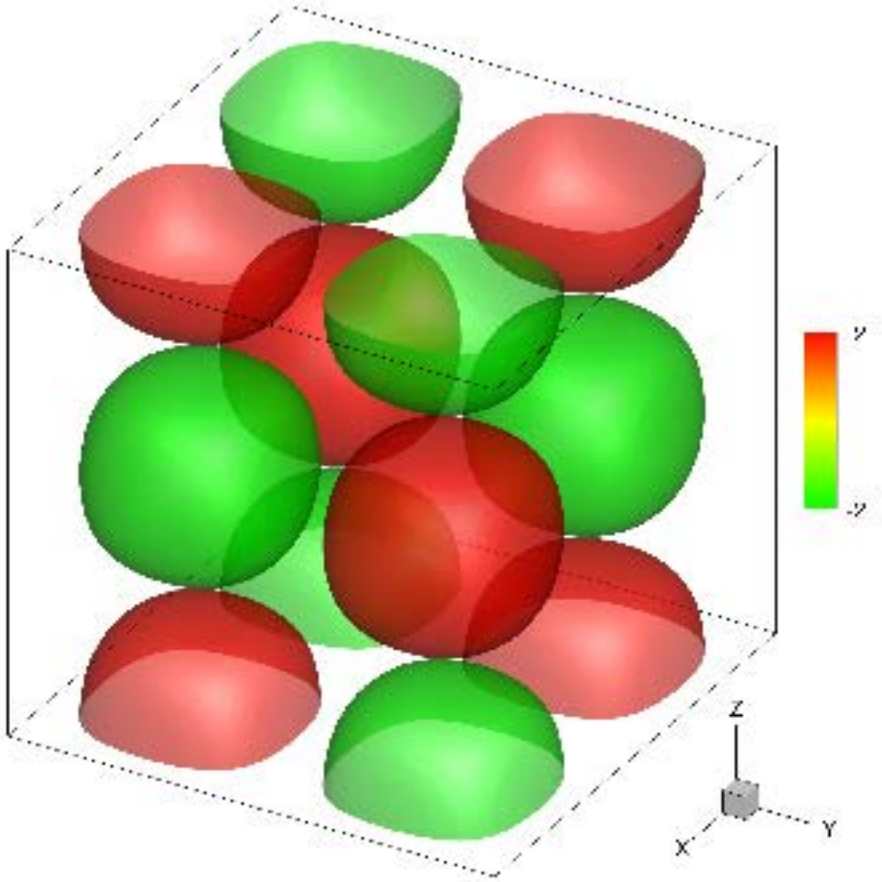


Figure 4.3: A cubic domain with dimensions $2\,m \times 2\,m \times 2\,m$ in which the Taylor-Green vortex flow is simulated. The iso-surfaces of the z component of vorticity, $\omega_z = \pm 2$, are also illustrated at $t = 0$.

order to compare the effects of non-Newtonian terms in solver efficiency. For the ideal behavior, computational time remains constant by increasing the number of processors, so Efficiency=100%. According to Figure 4.4, a continuous decay in efficiency of LS-STAG solver is observed with rising the number of processes np for both Newtonian and non-Newtonian cases since the communication between processors increases which generally degrades parallel performance. For $np > 12$, the communication of processors in different nodes decelerate the computations and subsequently reduces the efficiency. Similar trend is also reported in Baker *et al.* [6] for simulations with few number of processors. They investigated the scaling properties of several popular multigrid solvers in HYPRE for over 100000 cores. Although, the simulations of non-Newtonian cases are more time-consuming than the corresponding Newtonian ones, the efficiency of non-Newtonian cases are considerably higher than their Newtonian counterparts since only one extra variable ($\dot{\gamma}$) is added up to the communication step. Note that the efficiency and performance of the LS-STAG solver may be enhanced by improving its optimization level.

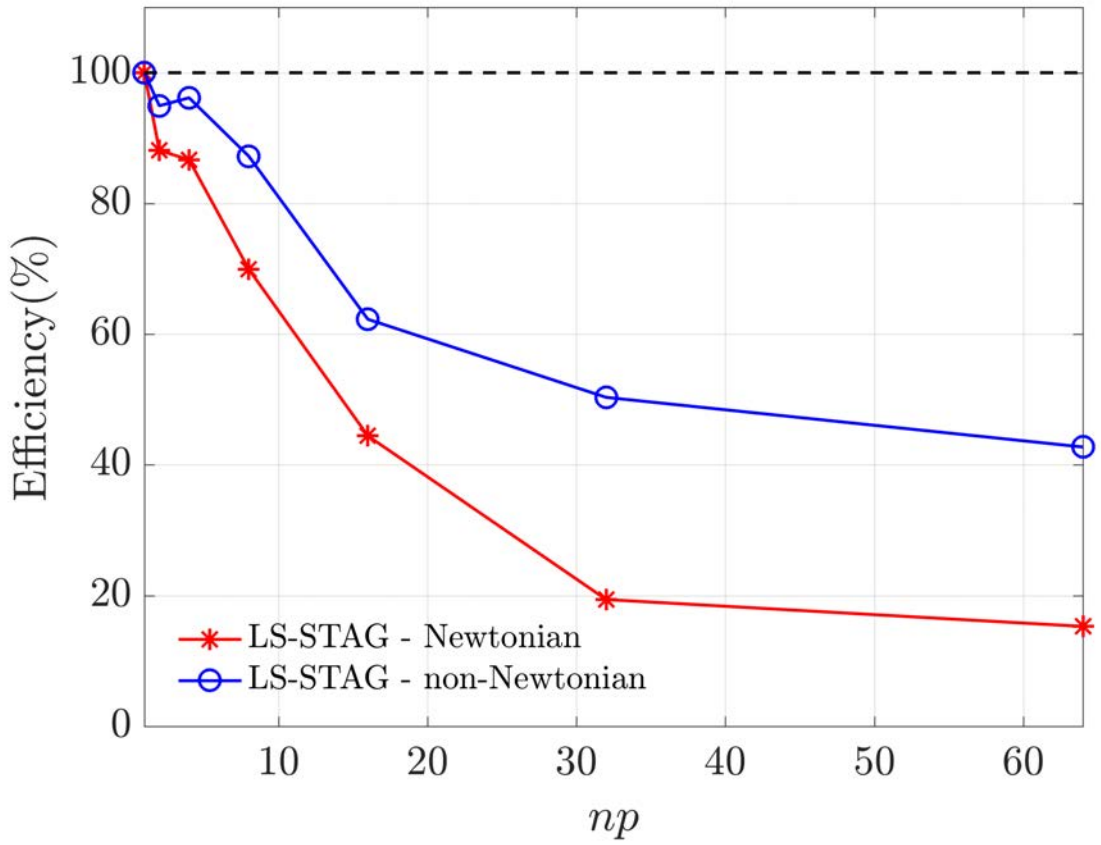


Figure 4.4: The efficiency of LS-STAG solver versus the number of processors (np) in the weak scaling measurement with the local grid $64 \times 64 \times 64$ cells per each processor. The dashed line (--) shows the ideal behavior.

Figure 4.5 shows the speedup of LS-STAG solver based on the strong scaling measurement both for Newtonian and non-Newtonian cases. In this measurement, a fixed global grid with $256 \times 256 \times 256$ cells is used for all computations. The speedup is measured as the ratio of simulation wall-time with 1 processor ($WT(1)$) to the wall-time with n processors ($WT(n)$):

$$\frac{WT(1)}{WT(n)}.$$

For the ideal behavior, the speedup is proportional to the number of processors. However, as the number of processors increases, the computational work per processor decreases, and the communication time typically increases and deteriorate the parallel speedup. Thus, we observe the deviation from the ideal behavior (dashed line (--)) for both cases (Newtonian and non-Newtonian) by increasing the number of processors. Similar behavior is also reported in speedup results of DG[1] method reported by Bosnyakov *et al.* [14] in computation of the Taylor-Green vortex flow for a mesh with $64 \times 64 \times 64$ elements. In addition, the communication overhead

[1]Discontinuous Galerkin

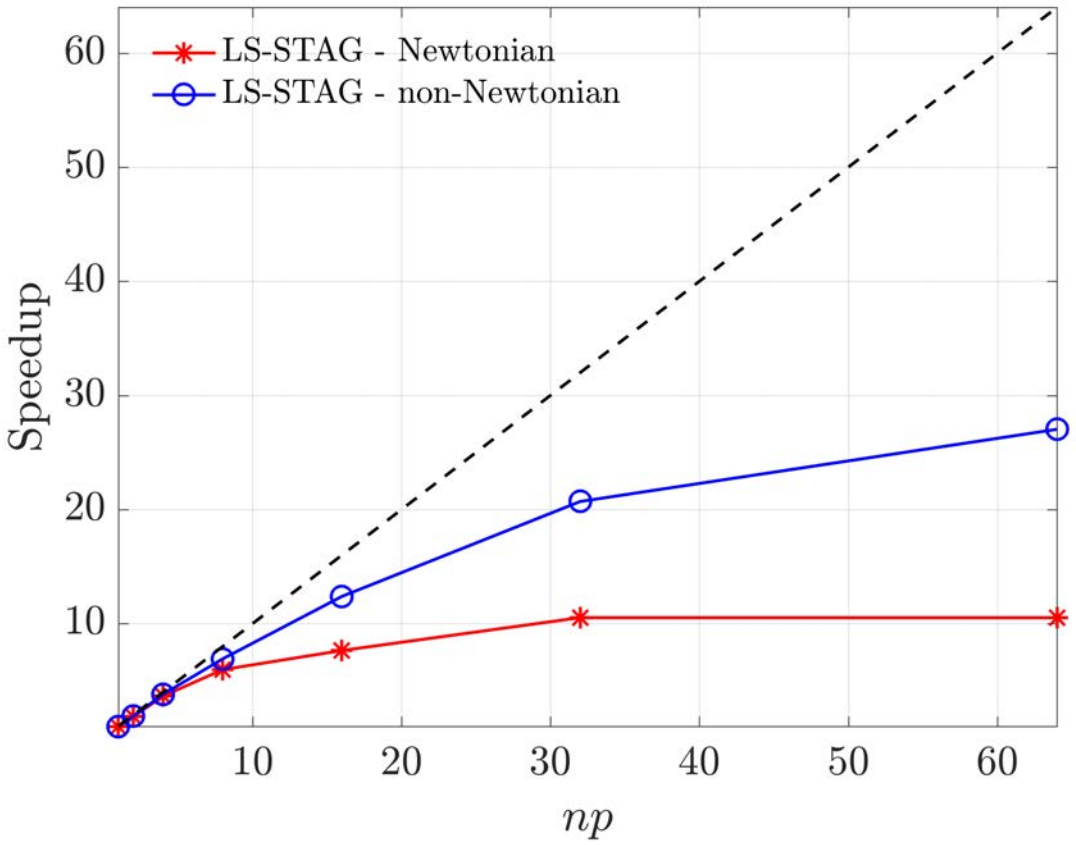


Figure 4.5: The speedup of LS-STAG solver versus the number of processors (np) in the strong scaling measurement. The dashed line (- -) shows the ideal speedup.

is significantly increases for $np > 12$ since the computations are conducted on different nodes. Such like the weak scaling measurement, the non-Newtonian cases shows better performance than the Newtonian ones.

We will subsequently evaluate the properties of LS-STAG method on various benchmark flows of Newtonian and non-Newtonian fluids in 3D extruded geometries. The spatial accuracy of LS-STAG method is assessed on Hagen-Poiseuille flow in Section 4.2. The flow past circular cylinder at different flow regimes will be studied in details in Section 4.3.

4.2 Accuracy study of the LS-STAG method

The spatial accuracy of the LS-STAG method for 2D non-Newtonian fluids has thoroughly been studied on Taylor-Couette flows in [15], where second-order accuracy has been observed in Cartesian cells away from the complex geometry and super-linear convergence in the cut-cells (measured orders are 1.4 for Newtonian fluids and 1.2 for a shear-thinning fluid with $n = 0.5$). Although a Taylor-Couette cell is an extruded geometry, the independence of exact solution

in the axial direction makes impossible to evaluate the accuracy of LS-STAG discretization in that direction. Hence, a more suitable test-case will be the Hagen-Poiseuille flow where the viscous fluid flow driven by a constant pressure gradient established between the two ends of a circular cylinder of radius R with no-slip boundaries, see Figure 4.6. For the power-law fluids,

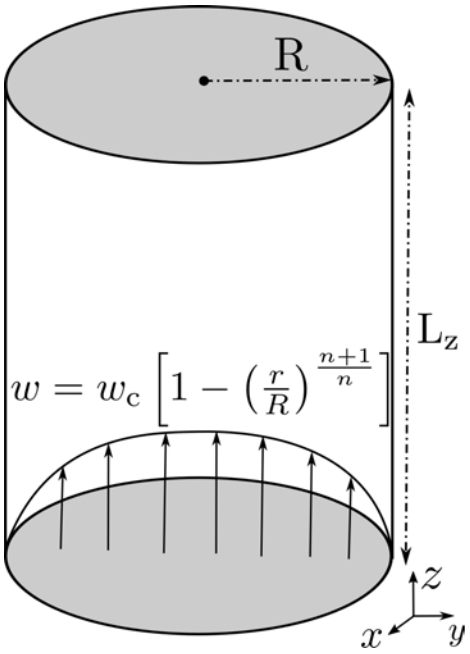


Figure 4.6: Geometry of Hagen-Poiseuille flow.

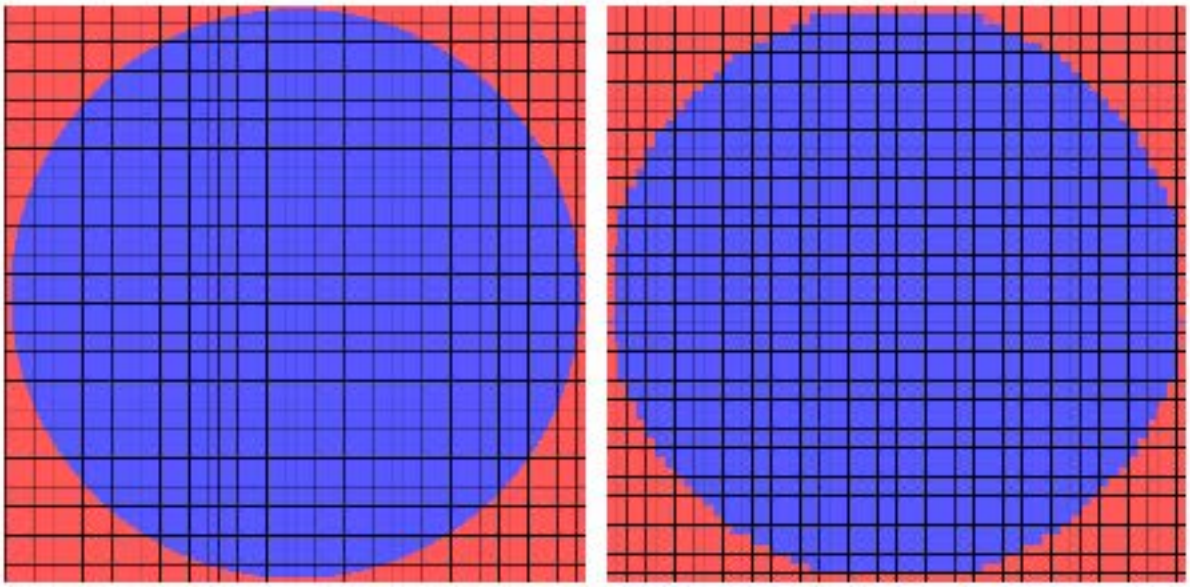
the analytical solution reads in polar coordinates:

$$w(r) = w_c \left[1 - \left(\frac{r}{R}\right)^{\frac{n+1}{n}}\right], \tag{4.2}$$

where w_c is the streamwise velocity occurs at the centerline ($r = 0$). For the shear-thinning fluids with the power-law index n, this velocity is computed as:

$$w_c = \frac{3n+1}{n+1}\,\overline{w},$$

in which $\overline{w}$ describes the mean velocity and can be obtained based on the pressure drop Δp across the circular cylinder such that $\overline{w} = \dfrac{\Delta p}{L_z}\dfrac{R^2}{8\mu}$. The constant pressure drop is modeled as a source term in the Navier-Stokes equations. Computations are performed for a Newtonian fluid and a shear-thinning fluid with $n = 0.5$ at $Re_D = 2000$, where $D = 2R$ and w_c are the reference length and velocity, respectively. The periodic boundary conditions are also imposed at the two ends of pipe in z direction.



(a) (b)

Figure 4.7: Comparison of the 2D cross section of the coarsest mesh ($h = 1.65 \times 10^{-2}$) for (a) LS-STAG and (b) staircase methods for Hagen-Poiseuille flow.

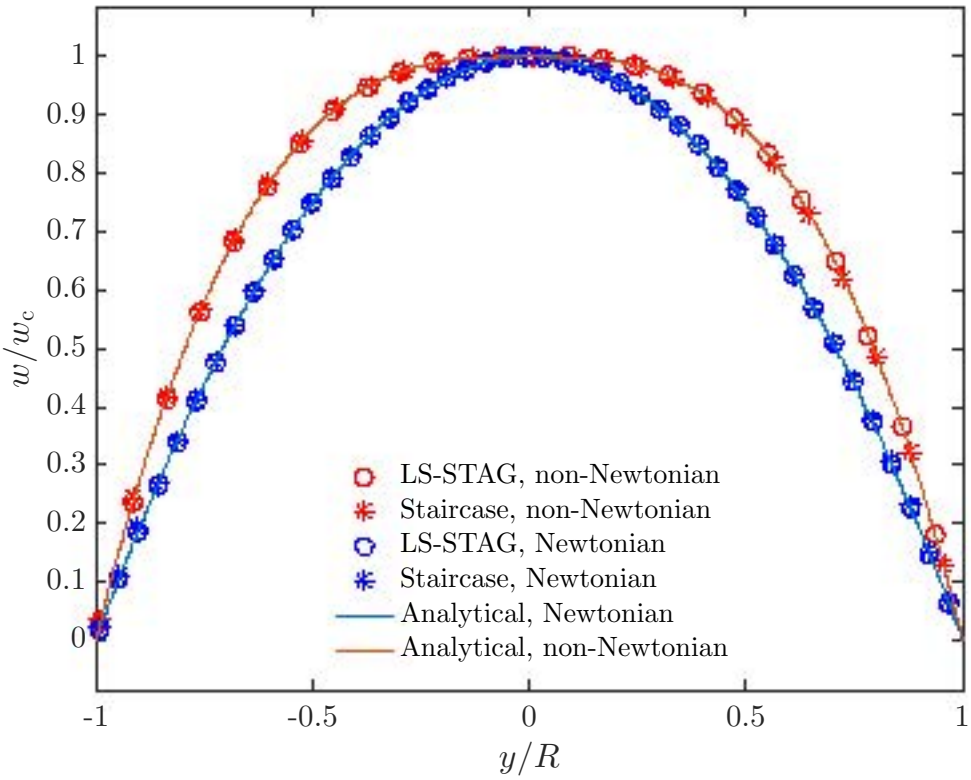


Figure 4.8: Axial velocity profile for Newtonian and shear-thinning Hagen-Poiseuille flows at cross section $z = L_z/2$ for $h = 5.5 \times 10^{-3}$.

The numerical simulation proceeds in time until we arrive at a solution that is close to the exact solution within the residual tolerance for w velocity. The residual tolerance is determined as the maximum relative difference of w between two successive time iterations such that:

$$\epsilon_w = \frac{\max \mid w^{n+1} - w^n \mid}{\mid w^{n+1} \mid}.$$

For this flow, the solution is converged when the residual tolerance is $\epsilon_w < 10^{-5}$.

This test-case has already been investigated by Van der Heiden *et al.* [99, 98] for a Newtonian fluid. Following their recommendations, the pipe dimensions are $2R \times L_z = 1 \times 2$. The simulations are carried out on a series of refined meshes with square cells of size h. In order to evaluate the cut-cell treatment in LS-STAG method, we have also done the same computations for the so-called staircase method which corresponds to a stepwise approximation of the irregular boundaries with Cartesian cells. It is implemented in LS-STAG code by imposing all cell-face fraction ratios (θ) equal to 1 in the cut-cells which transforms any cut-cell to the Cartesian cell. The staircase treatment only affects the discretization in cut-cells, while the discretization in Cartesian cells remains unchanged. According to [98], the computational grid has a constant number of cells in z direction and the uniform mesh refinement is applied in x and y directions. The cross section of the grid distribution for LS-STAG and staircase methods for the coarsest mesh ($h = 1.65 \times 10^{-2}$) is displayed in Figure 4.7. The sharp representation of the geometry in LS-STAG method is clearly observed in Figure 4.7(a) compared to the crude approximation of boundaries in staircase method shown in Figure 4.7(b). We will subsequently see that although a small portion of computational cells include the cut-cells, this transformation will greatly degrade the accuracy.

According to Equation (4.2), the axial velocity of the Hagen-Poiseuille flow is dependent on the value of the power-law index n. For Newtonian flow with $n = 1$, the velocity profile is quadratic ($w(r) \propto r^2$), while it is cubic ($w(r) \propto r^3$) for shear-thinning fluids with $n = 0.5$. In addition, the velocity profile is flatter near the centerline and decays faster towards the wall for the shear-thinning fluids. This behavior is also observed in Figure 4.8 that displays Newtonian and shear-thinning axial velocity profiles at section $z = L_z/2$ and compares them with the analytical solution. For both types of fluids, the velocity profiles are in an excellent agreement with the theoretical solutions even for staircase method.

For assessing the accuracy of LS-STAG method, we have computed the error on the streamwise velocity w in the maximum norm L_∞:

$$E_h(w) = \max_{\text{CVs } \Omega^w_{i,j,k}} |w_{i,j,k} - w^{\text{Theo}}(x^c_i, y^c_j, z_k)|,$$

where $w^{\text{Theo}}(x^c_i, y^c_j, z_k)$ is the theoretical profile Equation (4.2) computed at the grid points, including the cut-cell. Figure 4.9 compares the errors obtained on the whole computational domain and in Cartesian cells away from the irregular boundary for both cases, namely, Newtonian ($n = 1$) and shear-thinning with $n = 0.5$. As in the 2D cases [15], the LS-STAG errors are more

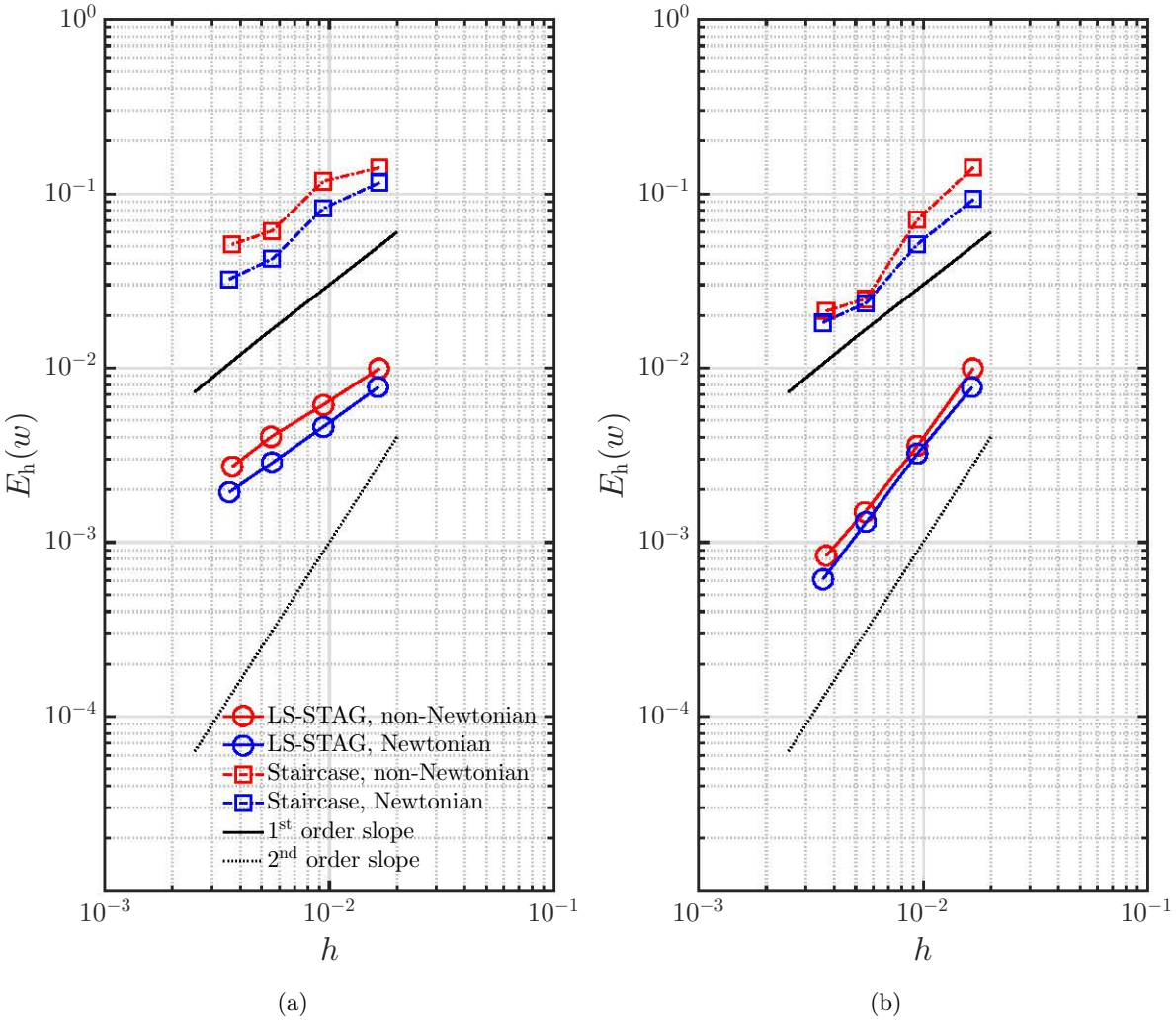


Figure 4.9: Maximal error on streamwise velocity versus grid size h for the Hagen-Poiseuille flow (a) on the whole fluid domain and (b) on 90% of the fluid domain.

than a decade lower than those of the first-order staircase method, which highlights the benefits of our cut-cell treatment. Second-order accuracy is obtained in Cartesian cells away from the irregular boundary for both fluid types. The magnitude of errors are slightly higher for the power-law fluids due to additional errors introduced by computation of non-Newtonian viscosity. However, compared to the 2D case, the error on the whole domain which actually measures the error in cut-cells, reduces from super-linear to linear only. This behavior, also observed in the cut-cell method of Van der Heiden [99], is certainly caused by the ad hoc computations of the velocity gradients $\partial w/\partial x$ and $\partial w/\partial y$ which were absent from the 2D version.

Further insight on the structure of discretization error is given in Figure 4.10. In this figure, the pointwise error on mesh points along the cross section $z = L_z/2$ is displayed for $h = 5.5 \times 10^{-3}$. As expected, the largest errors are localized near the complex boundary. For the LS-STAG method, the errors drop to several orders of magnitude away from the cut-cells.

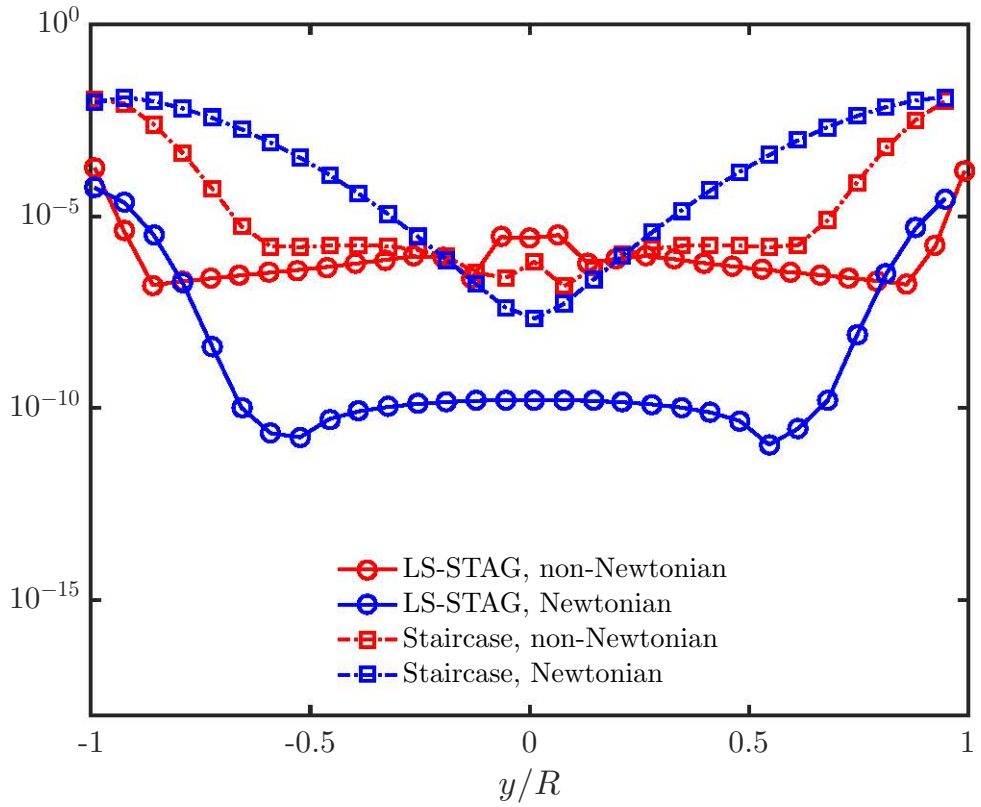


Figure 4.10: Pointwise error profiles on cross section $z = L_z/2$ for a mesh with $h = 5.5 \times 10^{-3}$ in Hagen-Poiseuille flow.

In contrast, the crude treatment of the staircase method pollutes the solution in the whole fluid domain. For non-Newtonian fluids, the difference between the staircase and LS-STAG method is less pronounced. Similar trend was also reported in the 2D version of LS-STAG for Taylor-Couette flows, refer to [15], which is attributed to the additional errors introduced by evaluation of the non-Newtonian viscosity in the viscous fluxes.

4.3 Computation of flows around a circular cylinder

In this section, the performances of the LS-STAG code will be valuated for benchmark cylinder flows at various regimes (a) steady (b) unsteady laminar and (c) weakly turbulent. The first 2 flows are set in a confined square duct, for which accurate solutions have been obtained by the group of Turek [10, 90] with various unstructured flow solvers. These benchmark results will assess the performances of the LS-STAG methods in terms of accuracy and computational resources against the commercial package ANSYS-FLUENT©.

The third test case concerns the turbulent flow past an unconfined cylinder at $Re_D = 300$, for which accurate results have been reported with a variety of numerical methods (*e.g.* [66, 59, 83]), and in particular to us the immersed boundary method of Balaras [7]. By performing this flow

computation on the same Cartesian grids, we will be able to firmly evaluate the performance of cut-cell methods compared to immersed boundary methods.

All these flows have been performed in the geometry depicted in Figure 4.11. The domain dimensions and conditions at the boundaries will be given subsequently for each configuration.

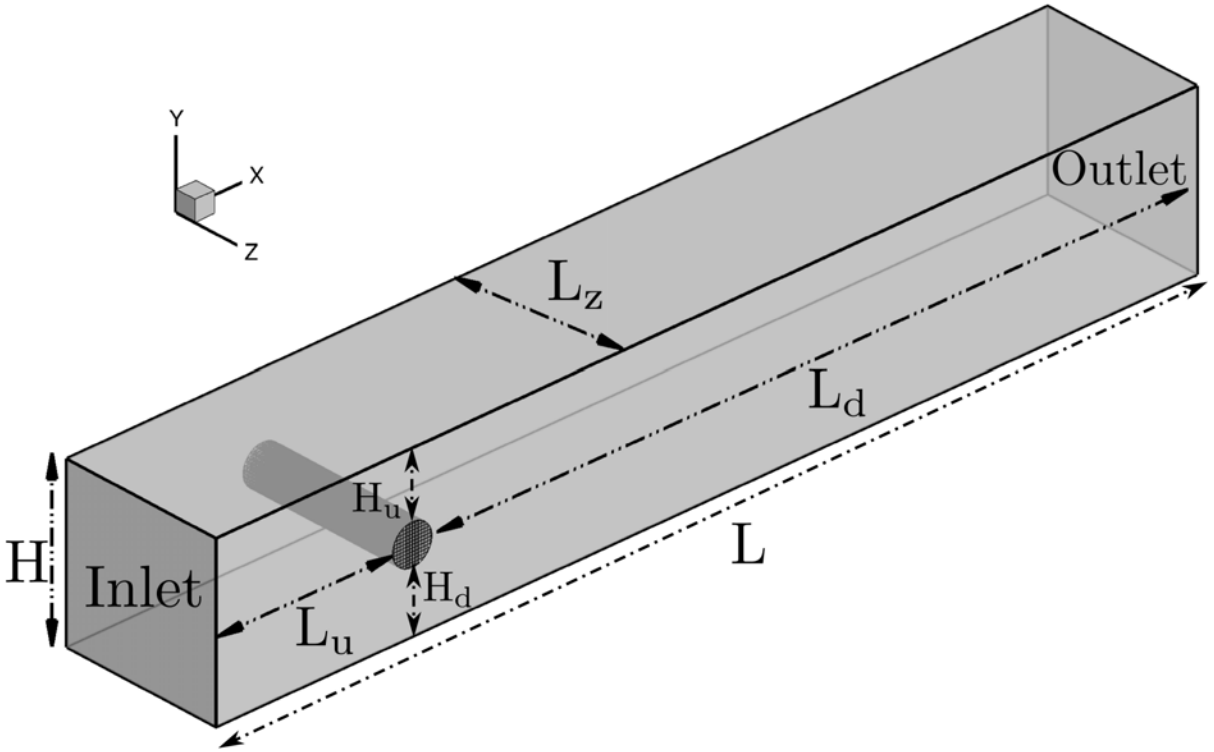


Figure 4.11: Computational domain for the cylinder flows of Section 4.3.

4.3.1 Benchmark computations of laminar flow past a circular cylinder in a confined square duct

A series of benchmark problems have been defined by Schäfer *et al.* [90] within the "DFG Priority Research Program" for 2D and 3D laminar flows around a cylinder in a confined square duct. The first 3D circular cylinder problem concerns the flow around a cylinder at $Re_D = 20$ (Benchmark 3D-1Z). For this low Reynolds number, accurate results have been reported by John [54] and Braack and Richter [19], that will be used as reference for our computations.

The second interesting benchmark problem is an unsteady laminar flow with a time-varying Reynolds number past a circular cylinder which corresponds to Benchmark 3D-3Z in [90]. This flow has been studied by John [55] in order to evaluate the efficiency of the numerical approaches to the solution of incompressible Navier-Stokes equations. However, a comprehensive benchmark case is established by Bayraktar *et al.* [10] who conducted a thorough comparison of the per-

formances of 3 unstructured CFD packages (OpenFOAM, Ansys-CFX and FeatFlow) on a sequence of body-fitted curvilinear grids and provided highly accurate benchmark results. These body-fitted grids have been communicated to us by the authors, so we have been able to use them for performing a series of computations with the commercial CFD package FLUENT© (V14.57) on the same cluster used for the LS-STAG computations.

The following computational domain and boundary conditions have been used for both steady and unsteady cases. In Figure 4.11, the size of the rectangular duct is $H \times L_z = 0.41\ m \times 0.41\ m$, the cylinder diameter is $D = 0.1\ m$, and the upstream and downstream distances from the cylinder are $L_u = 0.45\ m$ and $L = 1.95\ m$, respectively. In the cross-stream direction (y), there is an offset of 1 cm ($H_d = 0.15\ m$ and $H_u = 0.16\ m$) in order to generate a non-zero lift force on cylinder. No-slip boundary conditions are applied at the spanwise and cross-stream boundaries and a convective outflow condition [74] is applied at the outlet.

For steady cases, a parabolic stream-wise velocity profile is set at the inlet as:

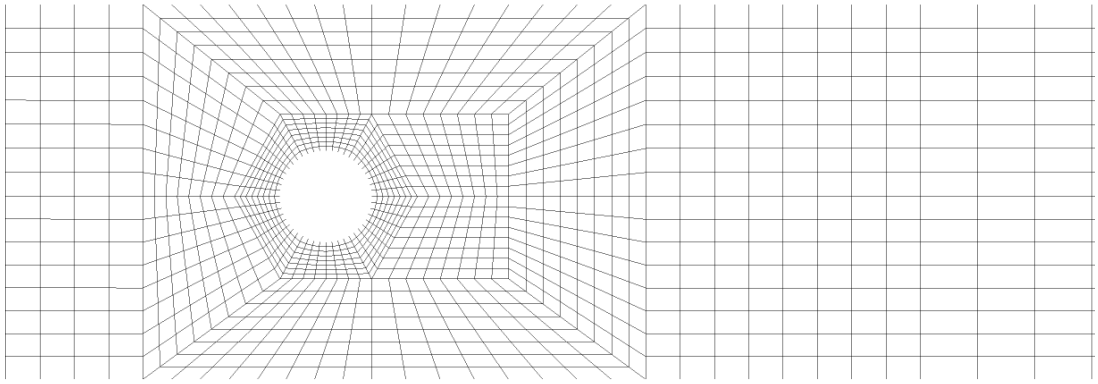
$$u_{\text{inlet}} = 16\, U_{\max}\, y\, z \frac{(H-y)(H-z)}{H^4}, \quad U_{\max} = 0.45\ m/s. \tag{4.3}$$

The fluid viscosity of $\nu = 10^{-3}\ m^2/s$ yields the Reynolds number $Re_D = 20$ based on the cylinder diameter $D_{\text{ref}} = D$ and the mean flow velocity $U_{\text{ref}} = 4U_{\max}/9$ at the inlet. For unsteady flows, a sinusoidal inflow profile is prescribed such that:

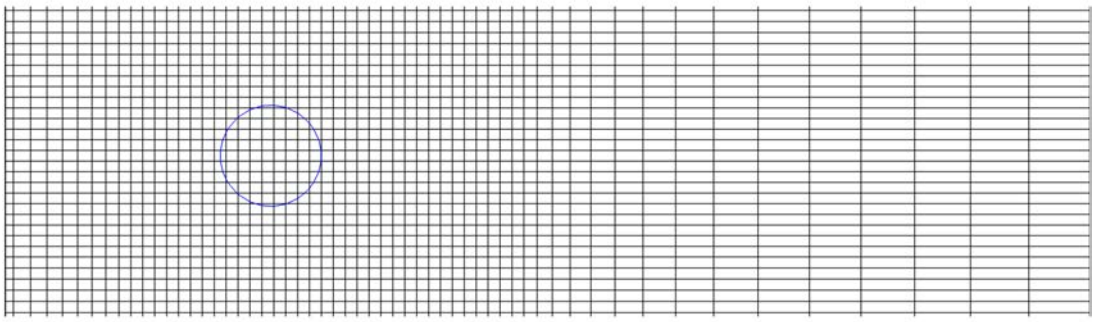
$$u_{\text{inlet}} = 16\, U_{\max}\, y\, z \sin(\frac{\pi\, t}{8}) \frac{(H-y)(H-z)}{H^4}, \quad U_{\max} = 2.25\ m/s, \tag{4.4}$$

which yields a time-varying Reynolds number $Re_D(t) \in [0, 100]$.

Following Bayraktar *et al.* [10], 3 different grids, labeled L2-L4, have been used in our FLUENT© computations. A 2D xy view of the coarsest mesh (L2) is shown in Figure 4.12(a). The refined grid layer around the cylinder surface consists in near-uniform hexahedral cells of size $\Delta_2/D = \sqrt{h_x^2 + h_y^2 + h_z^2}/D \simeq 0.31$ based on the main diagonal at the cylinder surface. The 3D mesh is obtained by extrusion in the z direction with a uniform distribution of cells, which gives a number of "49152" 3D cells. Finer meshes of cell sizes $\Delta_2/2$ and $\Delta_2/4$ are obtained by successive refinement, yielding meshes L3 and L4 with "393216" and "3145728" cells, respectively. For the LS-STAG computations, corresponding block-structured Cartesian grids M2–M4 have been constructed such that the total number of fluid cells and the grid resolution near the complex boundary are as close as possible to the curvilinear meshes L2-L4 in spite of the inferior flexibility of Cartesian grids. The xy view of grid M2 is indicated in Figure 4.12(b). The grid clustering around the cylinder is obtained by defining a rectangular block of $L_x/D \times L_y/D = 4.0 \times 2.0$ with uniform cells of size $h_x/D \times h_y/D = 0.118 \times 0.105$, while the grid size is relaxed upstream and downstream of this block. A uniform cell distribution of size $h_z/D = 0.256$ is used in the spanwise direction. The resulting 3D mesh has "49136" fluid cells of average size $\Delta_2/D \simeq 0.30$ near the cylinder body, which is similar to the L2 mesh. The main characteristics of the M2–M4 meshes are summarized in Table 4.1, along with the time-step used for the steady and unsteady cases. Note that the time-steps in steady flows result in a CFL number around 0.35 for all grids.



(a)



(b)

Figure 4.12: 2D view of the coarsest meshes for the circular cylinder benchmark: (a) L2 mesh of FLUENT© against and (b) M2 mesh LS-STAG method.

Table 4.1: Salient properties of the LS-STAG meshes used for the circular cylinder benchmark. Note that the time-steps are used for both LS-STAG and FLUENT© computations in unsteady cases.

Mesh	$\Delta t U_{\text{ref}}/D$		$N_x \times N_y \times N_z$	Cell Types			$h_x/D \times h_y/D \times h_z/D$
	Steady	Unsteady		Cartesian	Cut-cell	Solid	
M2	0.060	0.016	$83 \times 38 \times 16$	97.4%	1.1%	1.5%	$0.118 \times 0.105 \times 0.256$
M3	0.030	0.008	$159 \times 77 \times 32$	97.6%	0.6%	1.8%	$0.060 \times 0.051 \times 0.128$
M4	0.015	0.004	$317 \times 153 \times 64$	97.8%	0.3%	1.9%	$0.030 \times 0.026 \times 0.064$

The FLUENT© computations in unsteady cases have been performed with the pressure-based solver, with accurate discretization methods [37]: QUICK scheme for convection discretization, Green-Gauss node-based interpolation for gradient calculation, and PRESTO! for pressure interpolation. Second-order implicit time-stepping (BDF2) has been selected. For pressure-velocity coupling, 2 variants of the segregated PISO time-advancement have been evaluated: the iterative (default) and non-iterative (NITA) versions. The default PISO algorithm performs several inner iterations per time-step to solve the momentum and pressure equations

while the NITA variant solves these equations only once per time-step, yielding thus a time-advancement which resembles the most to our AB/BDF2 fractional step method. All other numerical parameters have been set to their default settings.

Steady flow at $Re_D = 20$

In this part, we will study a steady flow around a confined circular cylinder at $Re_D = 20$. In this flow, a pair of vortices is generated within the wake of the cylinder and at the sidewalls, see for example the snapshot of contour plot of spanwise vorticity, ω_z, in the center plane $z = 0.205\,m$ indicated in Figure 4.13. Note that the LS-STAG simulations of the flow at $Re_D = 20$ on meshes M2–M4 have been carried out starting from rest at $t = 0$ and proceeding in time until the steady state flow is reached. The steady state has been considered to be reached when the residual tolerances for velocity components $(\epsilon_u, \epsilon_v, \epsilon_w)$ are less than the convergence criterion 10^{-6}.

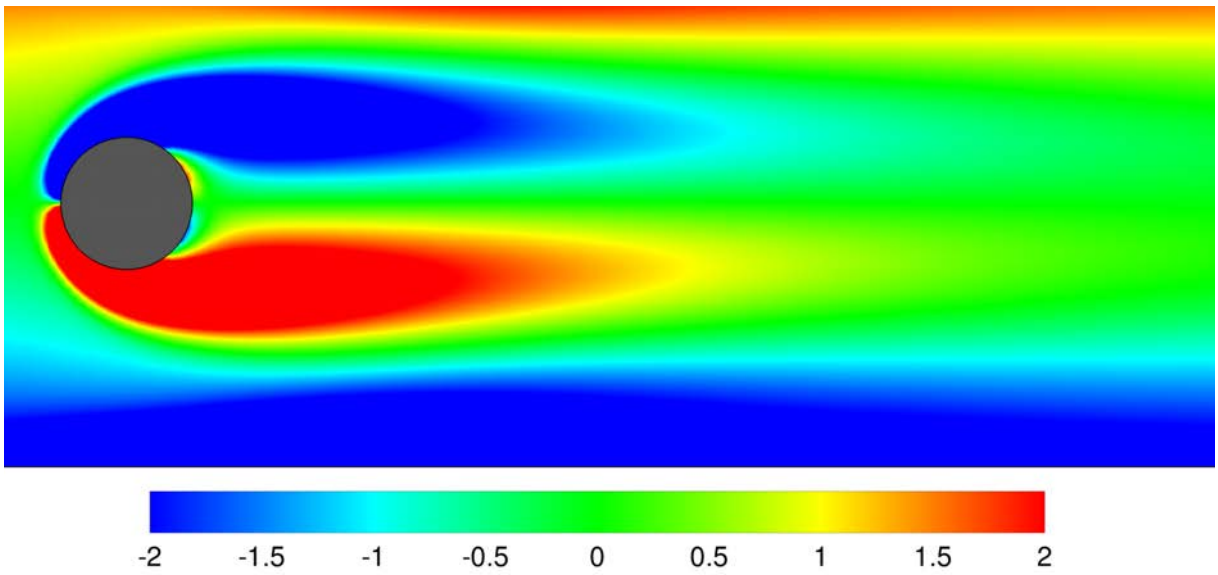


Figure 4.13: Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, in the center plane $z = 0.205\,m$ at $Re_D = 20$.

Figure 4.14 displays the streamwise (u) and cross-flow (v) velocities plotted along the cross-stream (y) direction at cross sections $x = 0.35\,m$, $0.5\ m$ and $0.65\ m$. It can be observed that the mesh M2 shows inaccurate behavior compared to the finer meshes M3 and M4 in all cross sections. These discrepancies are more drastic at cross section $x = 0.50\,m$, most notably in the vicinity of cylinder due to the inadequate grid resolution around the cylinder. In fact, since only a few grid points are distributed around the cylinder in this coarse mesh, the velocities are not estimated properly in this region. Thus, by increasing the number of grid points around the cylinder, we are able to capture the velocity profiles less sensitive to the grids, so we expect to obtain enable reasonable accuracy on these meshes. Note that the velocity profiles are not symmetric as there is an offset of $1\,cm$ in y direction.

Subsequently, in order to assess the accuracy of our method at this low Reynolds number, we will compare the global parameters of interest computed by LS-STAG with the reference values reported in [54, 19]. According to [54], the reference quantities to compute are the drag and lift coefficients and the pressure difference between the points at the front and rear of cylinder. These quantities are determined as:

$$C_{\mathrm{D}} = \frac{2F_{\mathrm{D}}}{\rho\, U_{\mathrm{ref}}^2\, D\, H}, \tag{4.5a}$$

$$C_{\mathrm{L}} = \frac{2F_{\mathrm{L}}}{\rho\, U_{\mathrm{ref}}^2\, D\, H}, \tag{4.5b}$$

$$\Delta P = P_{\mathrm{stag}} - P_{\mathrm{end}}, \tag{4.5c}$$

in which P_{stag} and P_{end} are measured respectively at the points $\boldsymbol{x}_{\mathrm{stag}} = (0.45, 0.2, 0.205)$ and $\boldsymbol{x}_{\mathrm{end}} = (0.55, 0.2, 0.205)$.

The results obtained by LS-STAG and staircase methods and their relative errors with respect to the reference data of [54, 19] are summarized in Table 4.2. For comparison purpose, the very precise values of CFD software package FeatFlow of Bayraktar *et al.* [10] for meshes L2–L4 and the lower and upper bounds of results reported by Schafer and Turek [90] are also given. The FeatFlow computations are in excellent agreement with the reference values as the curvilinear grids have greater geometrical flexibility than the Cartesian one, thus even the coarse curvilinear grids are able to capture flow details efficiently.

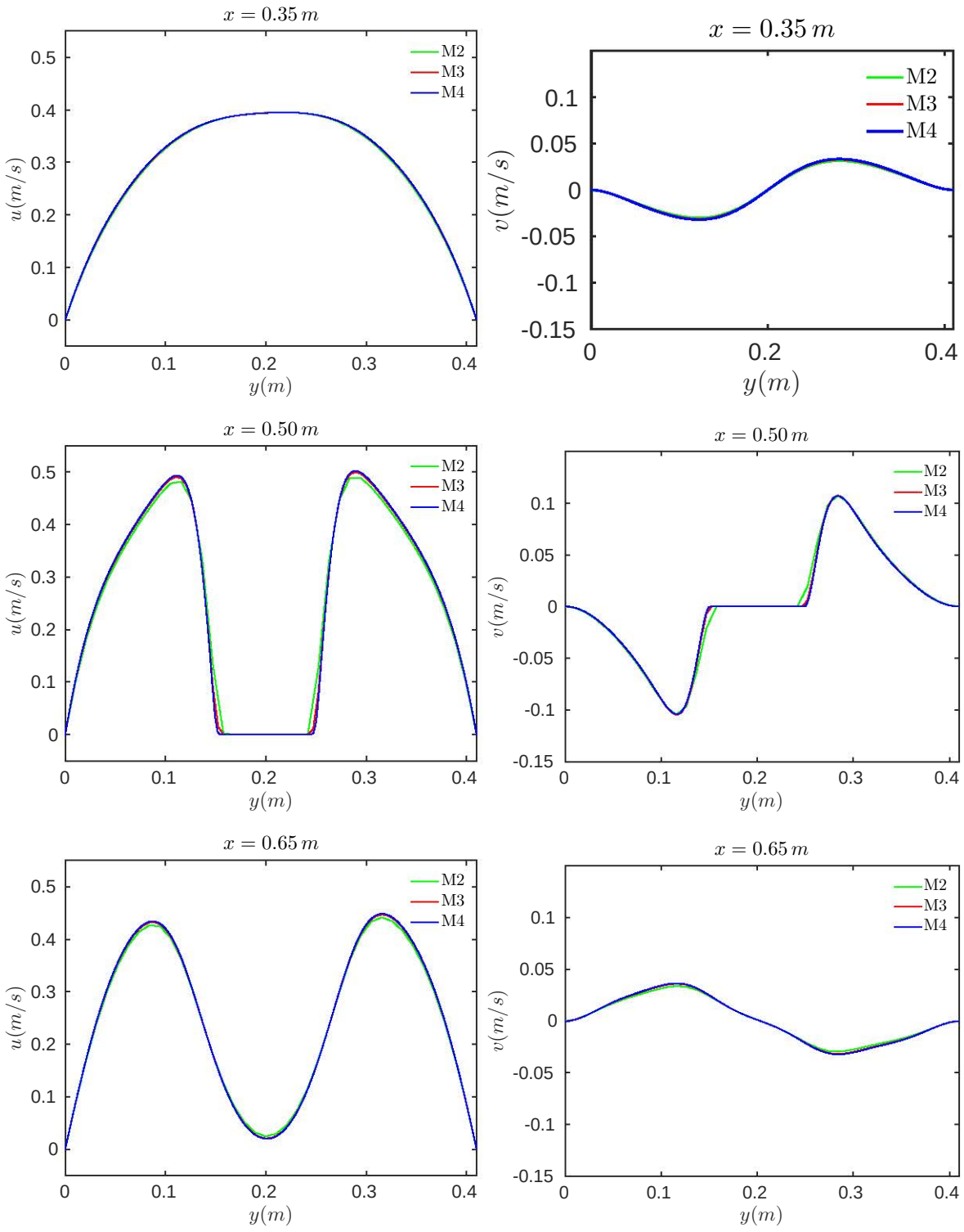


Figure 4.14: u and v velocity profiles for grid levels M2–M4 in y direction for 3 different locations $x = 0.35\,m$, 0.5 m and 0.65 m along the square duct.

Table 4.2: Comparison of global parameters obtained by LS-STAG and staircase methods with the reference values reported in [19, 54] ($C_{\mathrm{D}} = 6.1853$, $C_{\mathrm{L}} = 0.009401$ and $\Delta P = 0.171$) on meshes M2–M4 in cylinder flow at $Re_D = 20$. The FeatFlow computations on meshes L2–L4 reported in [10] and the lower and upper bounds of results in [90] are also given.

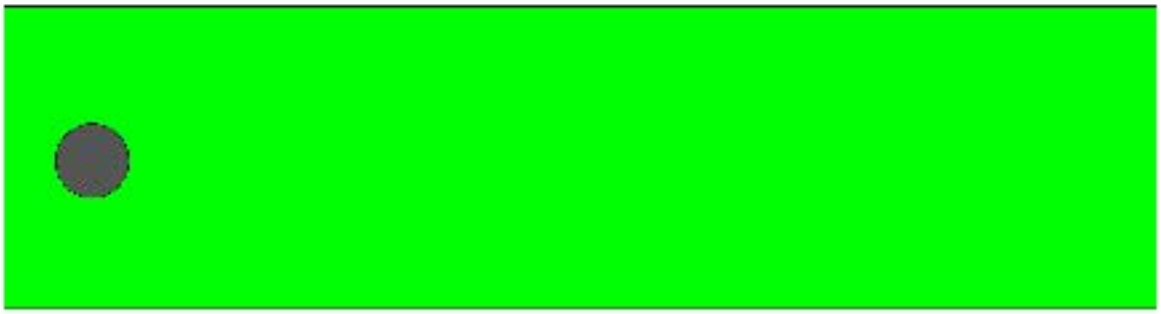
Case	C_{D}	% Err	C_{L}	% Err	ΔP	% Err
LS-STAG, M2	6.37	3.0	0.0063	33	0.139	19
Staircase, M2	4.67	24.5	0.0008	92	0.25	43.6
FeatFlow [10], L2	6.17	0.18	0.0094	0.21	-	-
LS-STAG, M3	6.42	3.8	0.0082	12.8	0.166	3.2
Staircase, M3	4.72	23.66	0.0057	39.7	0.162	5.4
FeatFlow [10], L3	6.18	0.04	0.0094	0.15	-	-
LS-STAG, M4	6.41	3.6	0.0088	6	0.170	0.5
Staircase, M4	4.64	25	0.0093	1.26	0.166	3
FeatFlow [10], L4	6.18	0.01	0.0094	0.05	-	-
Schafer & Turek [90]	6.05-6.25	-	0.008-0.010	-	0.165-0.175	-

Moreover, we can see that although LS-STAG method provides admissible results at $Re_D = 20$, they are not as accurate as those of the FeatFlow which is certainly due to the ad hoc computations of the $\partial w/\partial x$ and $\partial w/\partial y$ in w velocity. As a matter of fact, since the viscous terms are dominant at this low Reynolds number, the drag and lift forces have not been computed precisely even for the finest mesh (M4). But, the pressure difference have been very well estimated for the meshes M3 and M4. We will see later in Sections 4.3.1 and 4.3.2 that the accuracy will be improved at higher Reynolds as the viscous terms are less pronounced. Note that the values of pressure differences are not reported in benchmark computations of [10].

On mesh M2, even though the drag coefficient is incidentally better than finer meshes, the lift coefficient and the pressure difference are considerably far from the reference values since the grid is not sufficiently fine near the cylinder which yields to large numerical errors. As the number of grid points increases especially next to the cylinder, the Cartesian grids are mapped more efficiently to the complex geometry. Therefore, the magnitude of error decreases significantly for meshes M3 and M4. In addition, it is observed that for almost all meshes, LS-STAG achieved better prediction of global parameters than the so-called staircase method.

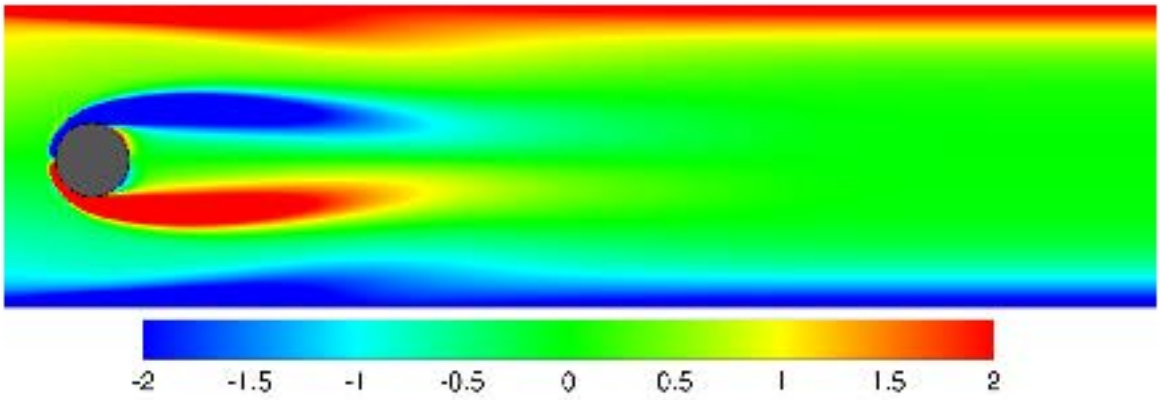
Comparison with unstructured body-fitted solver Ansys-FLUENT©

In this part, we aim to address 2 crucial questions regarding flow computations in complex geometries which, to the best of our knowledge, have seldomly been addressed in the literature: how accurate is the solution given by a cut-cell/IBM code compared to a body-fitted unstructured solver of same formal order of accuracy? How do the computational resources of the cut-cell/IBM code (memory usage, CPU time) compare with those of the body-fitted solver?



(a) $t = 0\,s$

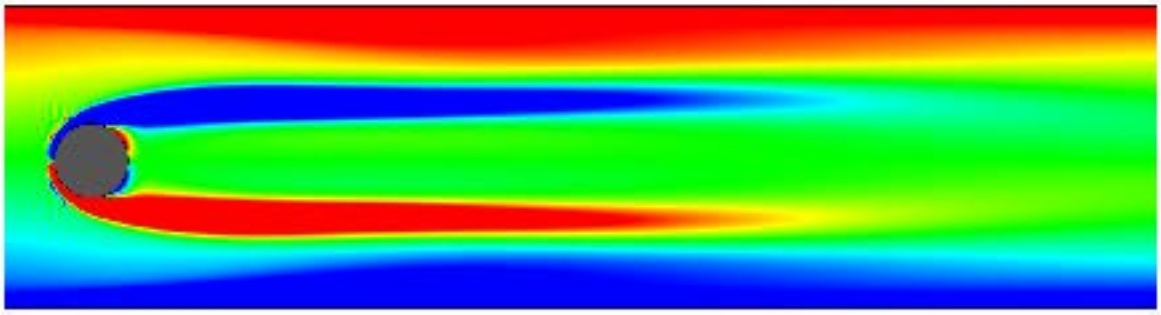
(b) $t = 1\,s$



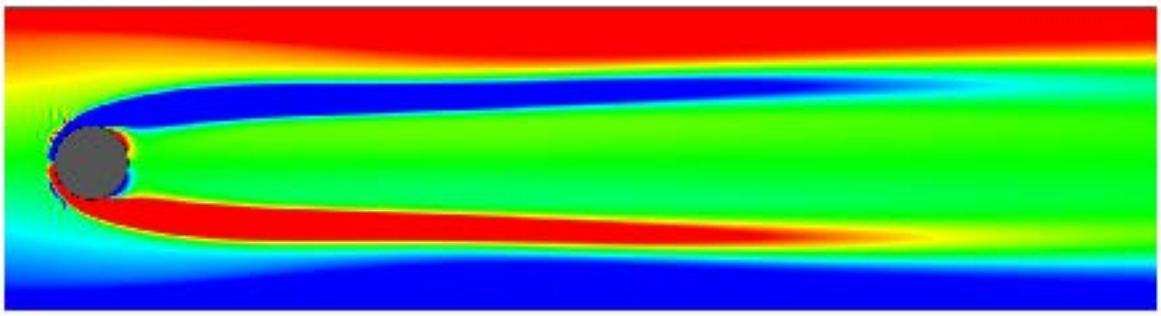
(c) $t = 2\,s$

Figure 4.15: Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, at the center plane $z = 0.205\,m$ for $t = 0,\ 1,\ 2\,s$.

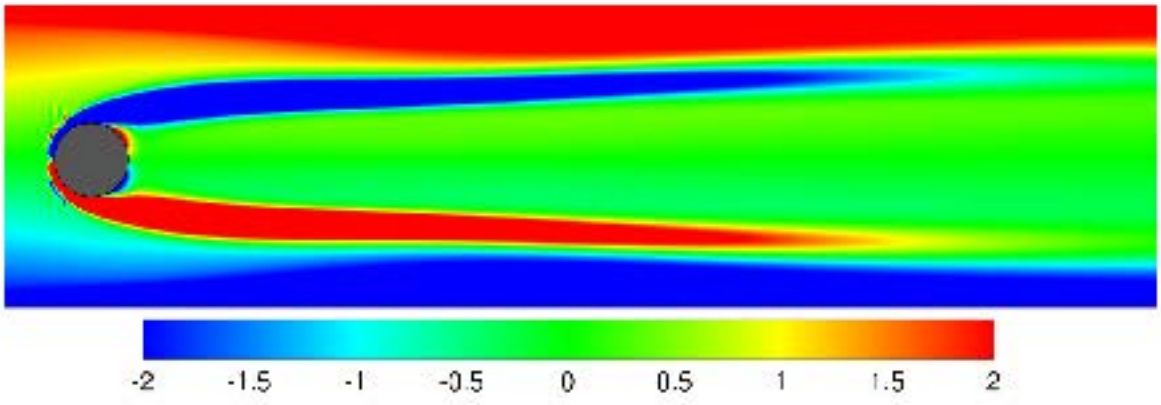
As in [10, 90], the LS-STAG and FLUENT© simulations are conducted within the time interval $[0, 8]\,s$ corresponding to a half period of the inflow condition (4.4). All computations are evaluated against a highly accurate FeatFlow solution obtained on the L5 mesh [9]. Each simulation begins from the rest (zero inflow and initial conditions) at $t = 0$ and terminates once



(a) $t = 3\,s$



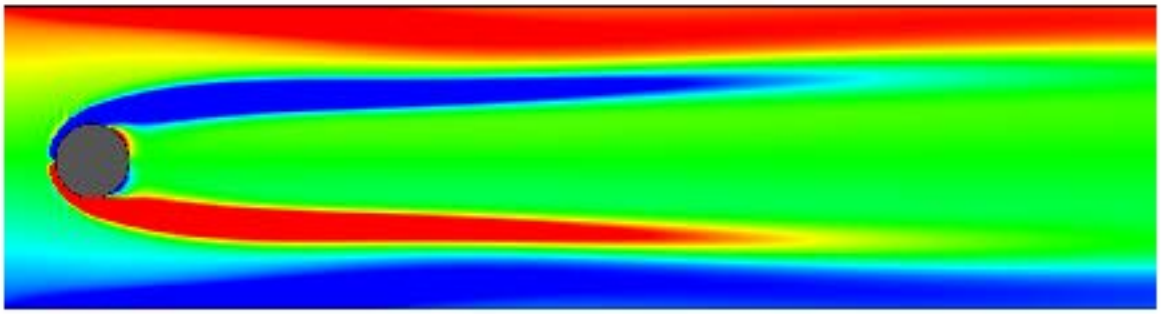
(b) $t = 4\,s$



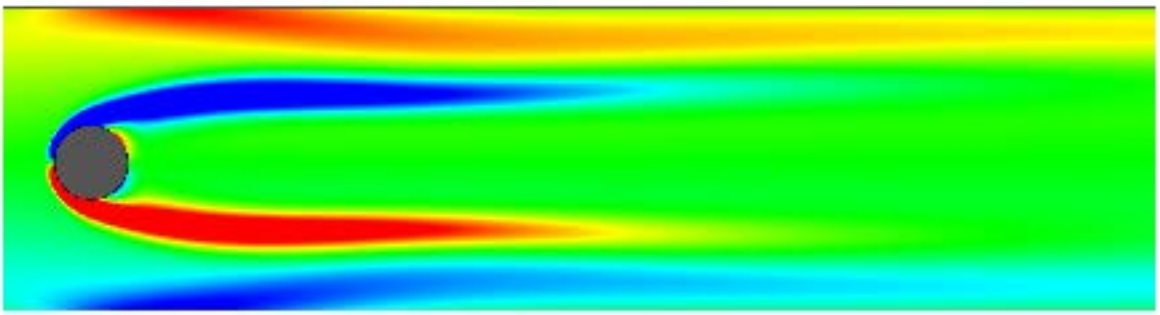
(c) $t = 5\,s$

Figure 4.16: Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, at the center plane $z = 0.205\,m$ for $t = 3,\ 4,\ 5\,s$.

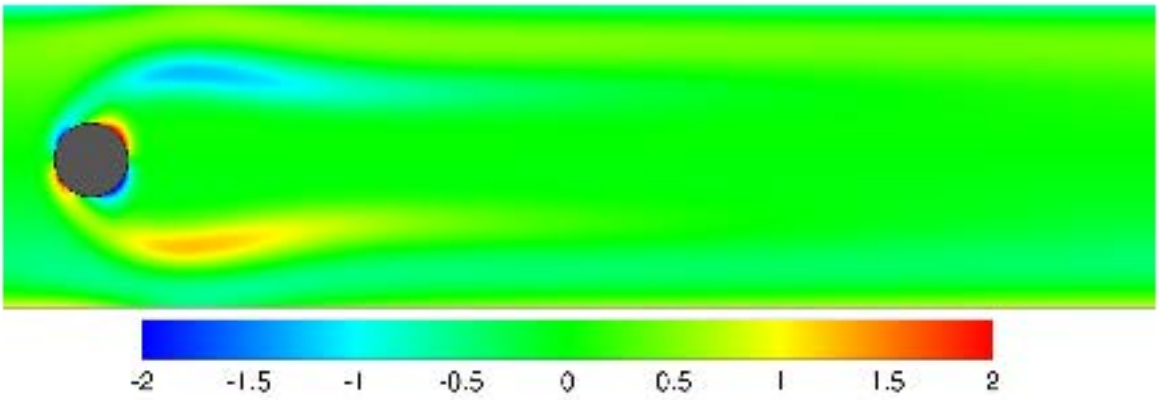
again at zero inflow at $t = 8$. Figures 4.15 to 4.17 indicate the snapshot of contour plots of spanwise vorticity, ω_z, at the center plane $z = 0.205\,m$ in different times. It is seen that the size and strength of vortices behind the wake of cylinder and at the sidewalls increase by the time up to the $t = 4\,s$ in which the time-varying Reynolds number $Re_D(t) = 100$ has its highest



(a) $t = 6\,s$



(b) $t = 7\,s$



(c) $t = 8\,s$

Figure 4.17: Contour plots of spanwise vorticity, $-2 \leq \omega_z \leq 2$, at the center plane $z = 0.205\,m$ for $t = 6,\ 7,\ 8\,s$.

value. Subsequently, the vortices becomes smaller (weaker) as the Reynolds number decreases by time within the interval $t \in [4, 8]\,s$.

Figure 4.18 displays the time evolution of the lift and drag coefficients obtained on various meshes. Note that the accuracy of LS-STAG method is evaluated mostly with the results of

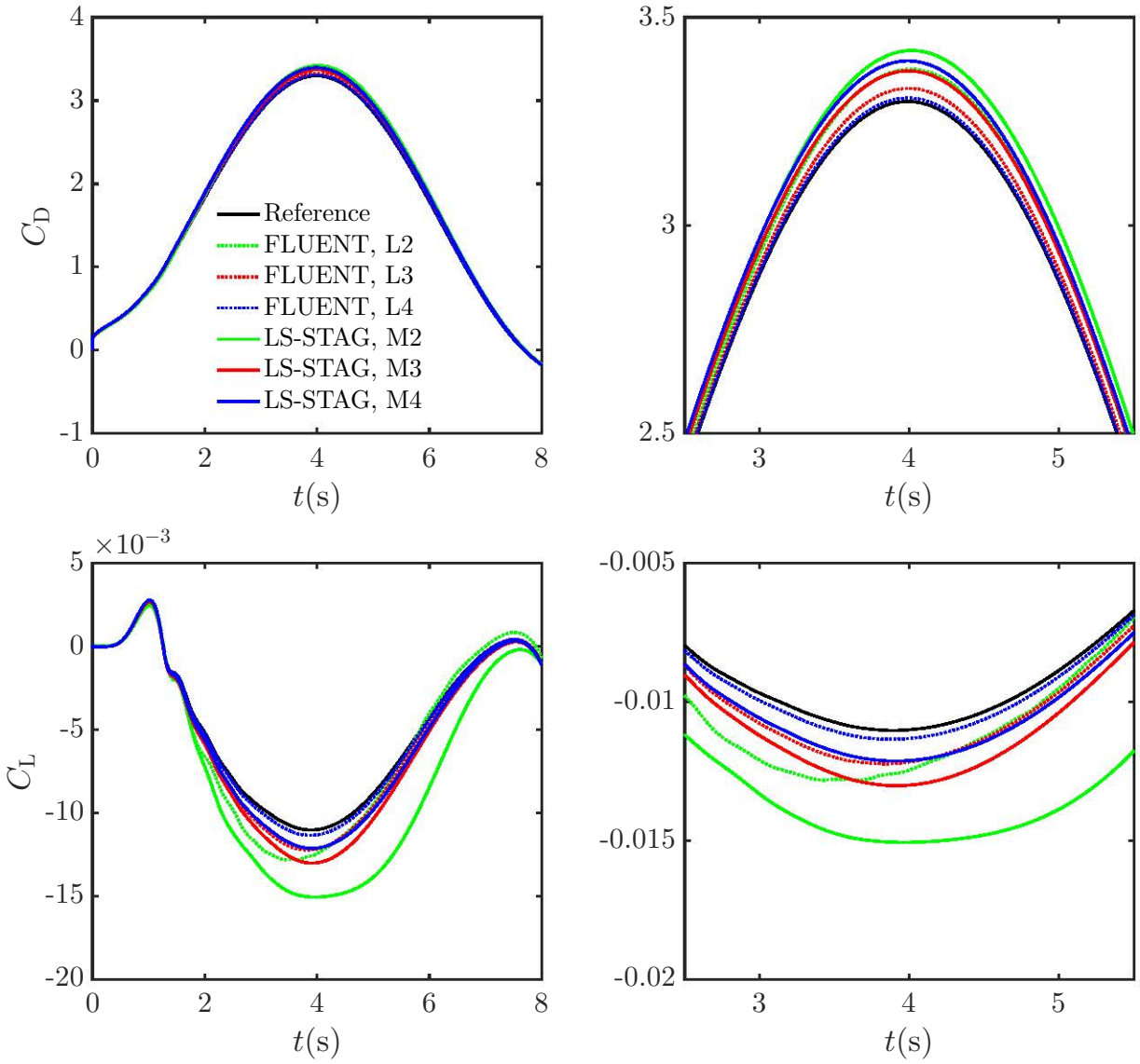


Figure 4.18: Time evolution of the force coefficients computed by LS-STAG, FLUENT© and the reference solution [9] at the cylinder. At left: on the total time interval; at right: close up around $t = 4\,s$.

lift coefficient as it is a more sensitive case than the drag coefficient. The LS-STAG solvers fails to capture the flow dynamics on the coarse M2 mesh due to insufficient grid resolution as in the steady flow at $Re_D = 20$. Unlike the case at $Re_D = 20$, the FLUENT© solver is not capable of predicting the flow details for the mesh L2. Accurate results are obtained on two finer meshes although the LS-STAG computation appears to be slightly inferior to FLUENT©, see the close-ups of Figure 4.18. The expected agreement of LS-STAG calculations with the reference data of [9] is clearly seen in this figure. However, relying only on these results is not sufficient at all to elucidate the accuracy of method. Therefore, we will also assess the global parameters of flow such as the drag and lift coefficients. As suggested in [10], a quantitative

comparison is provided in Table 4.3 where the errors on the extrema of force coefficients and L_2 and L_∞ norms of errors for $t \in [0, 8]\, s$ are compared. The L_2 and L_∞ norms of drag and lift coefficients are determined such that:

$$\|C_{\mathrm{D}}\|_2 = \sqrt{\frac{\sum_{t=0}^{8}(C_{\mathrm{D_{ref}}}(t) - C_{\mathrm{D_{ls}}}(t))^2}{\sum_{t=0}^{8}(C_{\mathrm{D_{ref}}}(t))^2}}, \tag{4.6a}$$

$$\|C_{\mathrm{L}}\|_2 = \sqrt{\frac{\sum_{t=0}^{8}(C_{\mathrm{L_{ref}}}(t) - C_{\mathrm{L_{ls}}}(t))^2}{\sum_{t=0}^{8}(C_{\mathrm{L_{ref}}}(t))^2}}, \tag{4.6b}$$

$$\|C_{\mathrm{D}}\|_\infty = \max_{0\le t\le 8}\left(|C_{\mathrm{D_{ref}}}(t) - C_{\mathrm{D_{ls}}}(t)|\right), \tag{4.6c}$$

$$\|C_{\mathrm{L}}\|_\infty = \max_{0\le t\le 8}\left(|C_{\mathrm{L_{ref}}}(t) - C_{\mathrm{L_{ls}}}(t)|\right), \tag{4.6d}$$

in which t is the equidistant discrete time-step and the notations of "ref" and "ls" stand for the results obtained from the reference benchmark [9] and LS-STAG method, respectively.

As expected, the LS-STAG results are more accurate since the effect of viscous terms are less dominant in this flow with higher Reynolds number than that at $Re_D = 20$. It can be observed that the quantities related to the lift coefficient, *i.e.* $C_{\mathrm{L_{min}}}$, $C_{\mathrm{L_{max}}}$ and C_{L} display a monotonic convergence. The errors are comparable to FLUENT© on the first 2 meshes, by slightly inferior to FLUENT© on M4/L4. In addition, results on the drag coefficient, $C_{\mathrm{D_{max}}}$ and C_{D}, although satisfactory, do not display a grid convergence as smooth as FLUENT©. This behavior is certainly caused by the discretization of velocity gradients $\partial w/\partial x$ and $\partial w/\partial y$ in the cut-cells which is not as mature as the other terms in the Navier-Stokes equations, refer to Section 3.4.4. This type of discretization is still inferior to standard finite-volume discretization in curvilinear grids.

Table 4.3: Relative errors (in percent) of the force coefficients with respect to the reference solution [9] ($C_{D_{\max}} = 3.2978$, $C_{L_{\max}} = 0.00276$ and $C_{L_{\min}} = -0.0110$) along with L_2 and L_∞ norms of errors within time interval $t \in [0, 8]\, s$.

Solver	Mesh	% Error			% L_2 Error		% L_∞ Error	
		$C_{\mathrm{D_{max}}}$	$C_{\mathrm{L_{max}}}$	$C_{\mathrm{L_{min}}}$	C_{D}	C_{L}	C_{D}	C_{L}
LS-STAG	M2	3.72	11.05	36.61	3.74	48.06	13.13	0.51
	M3	2.23	2.96	18.07	2.42	16.94	7.43	0.2
	M4	2.96	0.49	10.02	3.05	10.22	9.76	0.1
FLUENT©	L2	2.35	4.40	16.29	2.11	17.03	8.09	0.25
	L3	0.96	2.16	10.96	0.81	10.01	3.19	0.13
	L4	0.25	0.08	2.89	0.21	2.67	0.83	0.03

Finally, Table 4.4 presents the run-time and memory usage for the LS-STAG and FLUENT© computations. Due to the limited access of FLUENT© licenses in our institution, the compar-

isons have been performed on 2 processors for the coarse mesh computation (*i.e.* M2 and L2) and 4 processors for the two finer meshes. The FLUENT© computations with the default PISO algorithm proved to be exceedingly time-consuming (more than 22 times slower than LS-STAG) due to the cost of inner iterations and have not been pursued on the finest mesh. Compared to the NITA variant of the PISO algorithm, the LS-STAG code is still substantially faster (approximately 3 to 4 times) than FLUENT©. The memory saving in LS-STAG method is also

Table 4.4: Clock time and memory usage of the LS-STAG and FLUENT© computations on the IJL-LEMTA parallel machine. Note that *np* refers to the number of processors.

Mesh	*np*	Wall Time (min)			Memory Usage (mb)		
		LS-STAG	FLUENT©		LS-STAG	FLUENT©	
			PISO	NITA		PISO	NITA
M2/L2	2	16	363	55	90	406	400
M3/L3	4	132	2943	488	411	1368	1388
M4/L4	4	2164	-	7841	2417	-	7927

substantial: 4.5 times less memory needed than FLUENT© on M2/L2 and superior to 3 times for the finer meshes due to the on-the-fly computation of cell geometry parameters of the LS-STAG code. Finally, we mention that these comparisons are made with the current implementation of LS-STAG code that uses the HYPRE library for solving the linear systems. Since these linear systems typically take up to 60% of the computational cost of a simulation, the optimization of LS-STAG code will always be limited by the performances of its algebraic solver package.

As a conclusion to this section, the results of this benchmark flow are very promising for the cut-cell methods. The results do not exhibit a convergence as smooth as for a finite-volume on curvilinear grids, since discretization in cut-cells has not yet reached the level of maturity of curvilinear grid discretization. However, the computations are performed at a fraction of the cost of an unstructured solver, with a reasonable accuracy.

4.3.2 Weakly turbulent flow at $Re_D = 300$

The purpose of this section is to assess the ability of LS-STAG method to perform accurate direct numerical simulation of turbulent flows in complex geometries. The selected test-case is the weakly turbulent flow at $Re_D = 300$ around a circular cylinder with diameter D. At this Reynolds number, the flow is at the beginning of the wake turbulent shedding regime [105] and an accurate treatment of the fine scale structures in the near-cylinder wake is required to properly capture the flow dynamics. It has been chosen since a set of experimental results such as [73], and accurate body-conforming DNS, *e.g.* the spectral simulations of Mittal and Balachandar [66] and the zonal B-splines computations of Kravchenko *et al.* [59], are available in the literature. The aforementioned studies will serve as the reference test-cases for present work. In addition, this flow has also been simulated by a state-of-the-art immersed boundary method, refer to Balaras

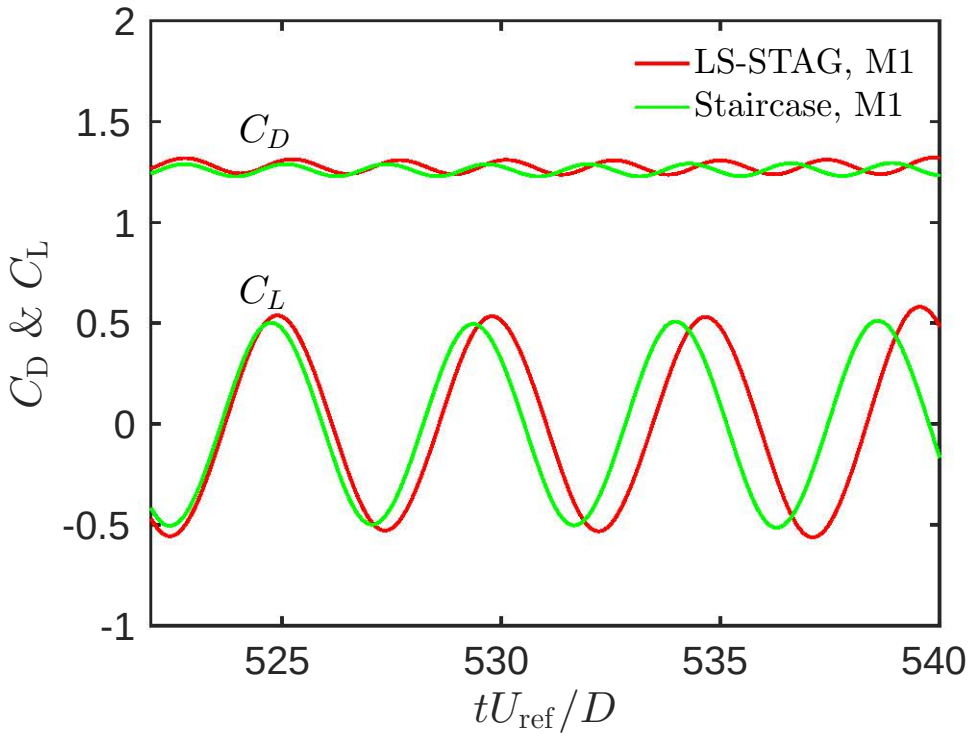


Figure 4.19: Time evolution of the force coefficients for the cylinder flow at $Re_D = 300$ on the M1 mesh.

[7]. In order to address the comparison of LS-STAG method with this IBM computation, identical computational domain and grid points as [7] have been used in our studies. The computational domain is a rectangular box with dimensions $L/D \times H/D \times L_z/D = 50 \times 30 \times 6$. Two Cartesian grids *i.e.* M1 and M2 are also constructed which corresponds to the Case 1 and Case 2 meshes in Balaras [7] computations, respectively. The coarser mesh M1 has $243 \times 133 \times 64$ cells with a grid distribution similar to the mesh displayed in Figure 4.12(b): the grid distribution is uniform in the spanwise (z) direction, and in the xy plane a uniform block of size $4D \times 4D$ is used in the vicinity of circular cylinder with grid resolution $h/D = 3.88 \times 10^{-2}$. A finer mesh M2 of $422 \times 226 \times 64$ cells with the grid resolution of $h/D = 1.94 \times 10^{-2}$ around the cylinder has also been considered in our simulations.

The following conditions are imposed to the boundaries: uniform velocity $u = U_{\text{ref}}$ at the inlet boundary, periodicity at the spanwise boundaries, traction-free conditions at cross-stream boundaries, and convective outflow condition at the outlet boundary. The time-steps are set as $\Delta t\, U_{\text{ref}}/D = 9 \times 10^{-3}$ and 4.5×10^{-3} on meshes M1 and M2, respectively, yielding a maximal cell CFL number of 0.41 during one shedding period. The initial condition is defined as the rest with random perturbations in the spanwise and crosswise direction to trigger the flow instability. The simulations have been advanced in time up to a statistically steady flow pattern. Then, statistics have been calculated by averaging in space in the spanwise direction and in time over 15 shedding cycles ($T \sim 80\, D/U_{\text{ref}}$). In addition to LS-STAG computations on M1 and M2

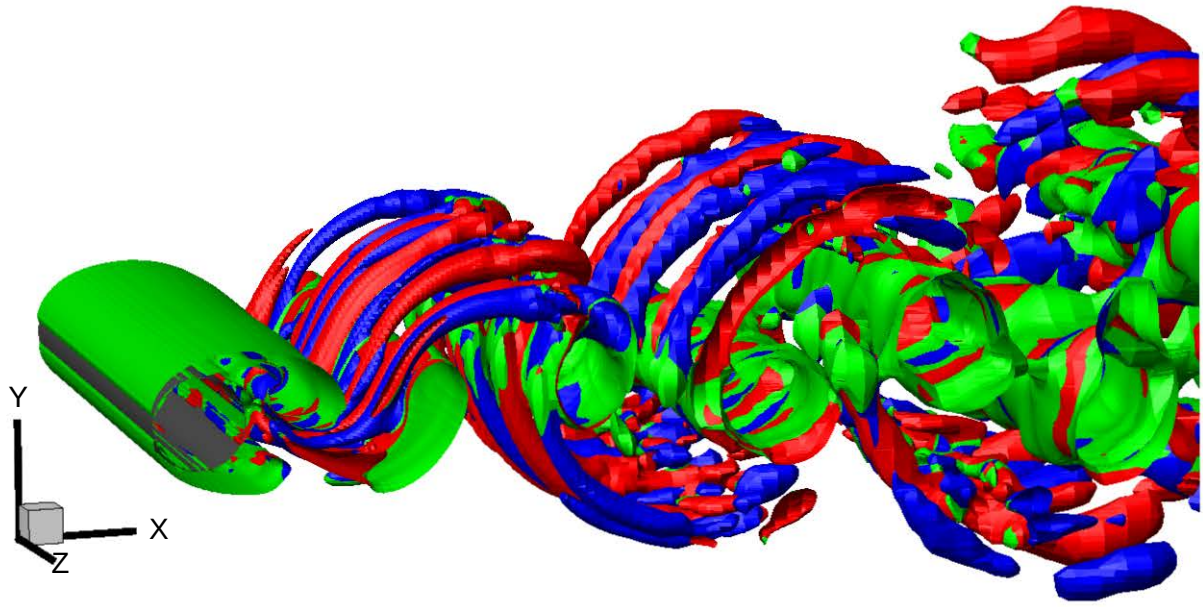


Figure 4.20: Vortex visualization of the cylinder flow at $Re_D = 300$ by the Q-criterion on mesh M2. The isosurfaces of $Q = 0.1$ (green) are displayed at an instant in time. The isosurfaces are colored with streamwise vorticity values: red is $\omega_x\, D/U_{\text{ref}} = 0.25$ and blue is $\omega_x\, D/U_{\text{ref}} = -0.25$.

meshes, a staircase simulation (see Section 4.2) on M1 mesh has also been performed in order to evaluate the effect of our cut-cell treatment. For both methods, the time history of the drag and lift coefficients is displayed in Figure 4.19. Although both methods give force signals with comparable amplitudes, the crude staircase treatment of the complex geometry introduces a spurious lag in the force signals which yields incorrect prediction of the shedding frequency, as we shall see next.

The flow at $Re_D = 300$ is characterized by the presence of streamwise-oriented coherent structures of spanwise length order of D in the cylinder wake [105]. These large vortical structures are best visualized by the Q-criterion [50], which identifies vortical flow regions to positive values of the second invariant of the velocity gradient tensor $\nabla \boldsymbol{v}$. By decomposing this tensor into a symmetric $S_{ij} = \left(\partial u_i/\partial x_j + \partial u_j/\partial x_i\right)/2$ and skew-symmetric $W_{ij} = \left(\partial u_i/\partial x_j - \partial u_j/\partial x_i\right)/2$ components, the Q-criterion is given such that:

$$Q = \frac{1}{2}(S_{ij}\, S_{ij} - W_{ij}\, W_{ij}).$$

Figure 4.20 displays isosurfaces of Q in the near-wake of circular cylinder at an instantaneous time. It is clearly shown the presence of large counter-rotating streamwise vortices in the near cylinder wake, whose number decreases further downstream. A similar qualitative observation of the wake structure has been performed in Kravchenko [59] by visualizing the instantaneous streamwise vorticity.

Table 4.5 compares the main flow characteristics, namely the mean drag coefficient, $\overline{C}_{\mathrm{D}}$, root mean square average of the lift coefficient, $\overline{C}_{\mathrm{L_{rms}}}$, and the Strouhal number $St = f\,D/U_{\mathrm{ref}}$, where f is the shedding frequency, with reference simulations [66, 59] and experimental measurements [104, 73], IBM computations [7] and the recent DNS of Rajani *et al.* [83] with a finite-volume solver on a multiblock polar grid. It is clearly seen that the shedding frequency is the easiest quantity to predict: only the IBM and staircase method on the coarsest mesh M1 and Rajani [83] misestimate its value. For the drag coefficient, accurate reference computations bracket its value in the range $[1.26, 1.28]$, for which the LS-STAG computations on both coarse and fine meshes give a very close prediction, $\overline{C}_{\mathrm{D}} = 1.29$. The rms lift coefficient proves to be the most sensitive value to predict, with reference values spread in the range $[0.380, 0.435]$, with an upper value given by the experiments [73]. Both coarse and fine meshes in LS-STAG simulation fall in the range, and in particular the M2 prediction is the closest computational result to the experimental value.

Table 4.5: Mean flow parameters for the cylinder flow at $Re_D = 300$.

Case	$\overline{C}_{\mathrm{D}}$	$\overline{C}_{\mathrm{L_{rms}}}$	St
Measurement [104]	1.22	–	–
Measurement [73]	–	0.435	0.203
Spectral simulation [66]	1.26	0.380	0.203
B-spline simulation [59], Case 1	1.24	0.330	0.202
B-spline simulation [59], Case 2	1.28	0.40	0.203
LS-STAG, M1	1.29	0.420	0.203
LS-STAG, M2	1.29	0.437	0.203
Staircase, M1	1.24	0.346	0.217
IBM Case 1 [7]	1.41	0.44	0.20
IBM Case 2 [7]	1.27	0.42	0.21
Body-fitted finite-volume [83]	1.28	0.499	0.195

The mean streamwise $\overline{u}/U_{\mathrm{ref}}$ and cross-stream $\overline{v}/U_{\mathrm{ref}}$ velocity profiles at three different downstream locations in the wake of circular cylinder are indicated in Figures 4.21 and 4.22, respectively. In addition, the time-averaged velocity fluctuations for the streamwise $\overline{u'\,u'}/U_{\mathrm{ref}}^2$ and cross-stream $\overline{v'\,v'}/U_{\mathrm{ref}}^2$ components and also the Reynolds shear stress $\overline{u'\,v'}/U_{\mathrm{ref}}^2$ are shown in Figures 4.23 to 4.25, respectively. In these figures, the LS-STAG and staircase computations are compared to IBM simulations [7] and reference spectral results [66]. Note that the B-spline solution of Kravchenko [59] are virtually indistinguishable from the spectral data and not shown in figures. Firstly, it is observed that the LS-STAG results on both coarse and fine meshes show excellent agreement with the reference results, which indicates that the M1 grid resolution is sufficiently fine to capture the wake dynamics at this flow regime. For this grid resolution, the IBM computation exhibits significant discrepancies most notably on the cross-flow velocities

(Figures 4.21 and 4.22) and velocity fluctuations (Figures 4.23 and 4.24) in the cylinder wake. It was conjectured that the discrepancies were caused by insufficient grid resolution, see [7]. However, on the light of LS-STAG computations, these discrepancies are more likely due to the inferior treatment of complex geometries given by the IBM compared to a cut-cell method of formally same order of accuracy. It is also interesting to observe that the staircase, which is formally a first-order method according to Section 4.2, gives superior results to the IBM method at similar grid resolution, see Table 4.5 and Figures 4.21 to 4.24.

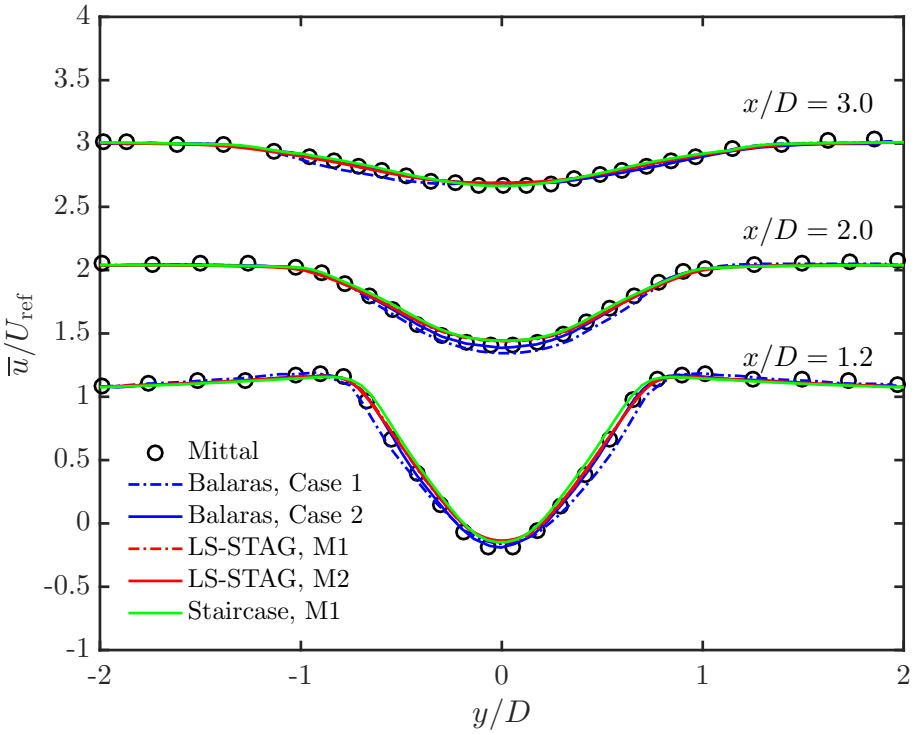


Figure 4.21: Mean streamwise velocity profiles at different locations in the wake of circular cylinder at $Re_D = 300$.

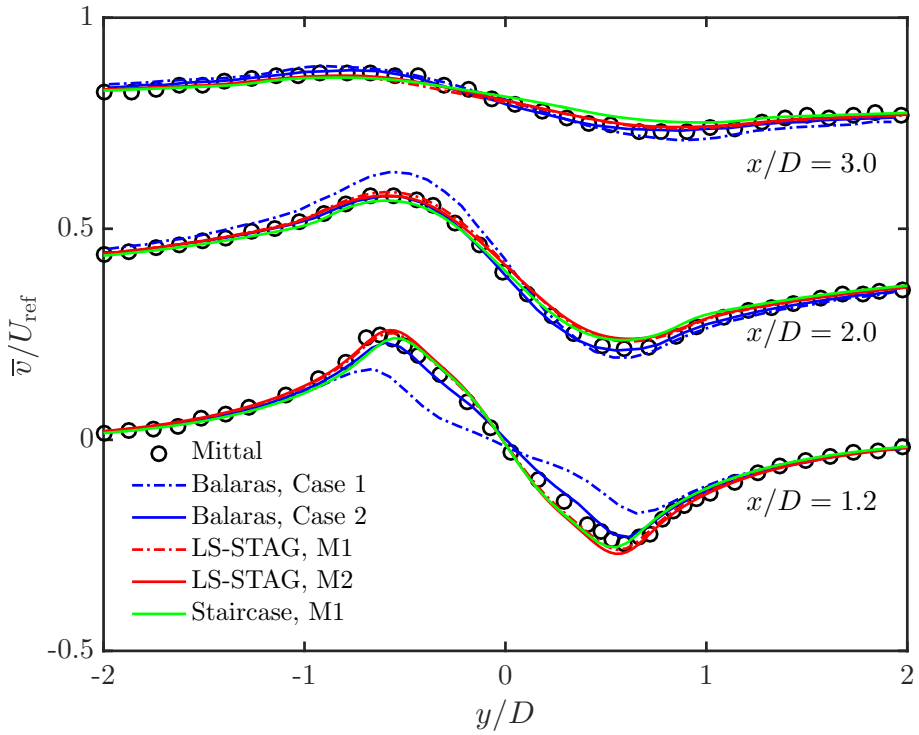


Figure 4.22: Mean cross-flow velocity profiles at different locations in the wake of circular cylinder at $Re_D = 300$.

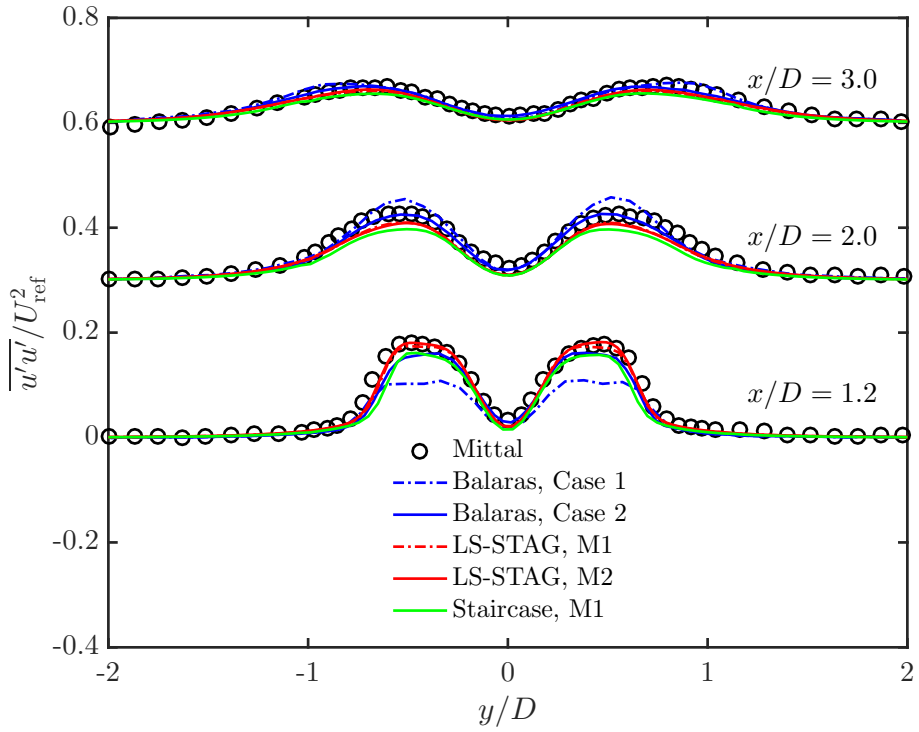


Figure 4.23: time-averaged streamwise velocity fluctuations at different locations in the wake of circular cylinder at $Re_D = 300$.

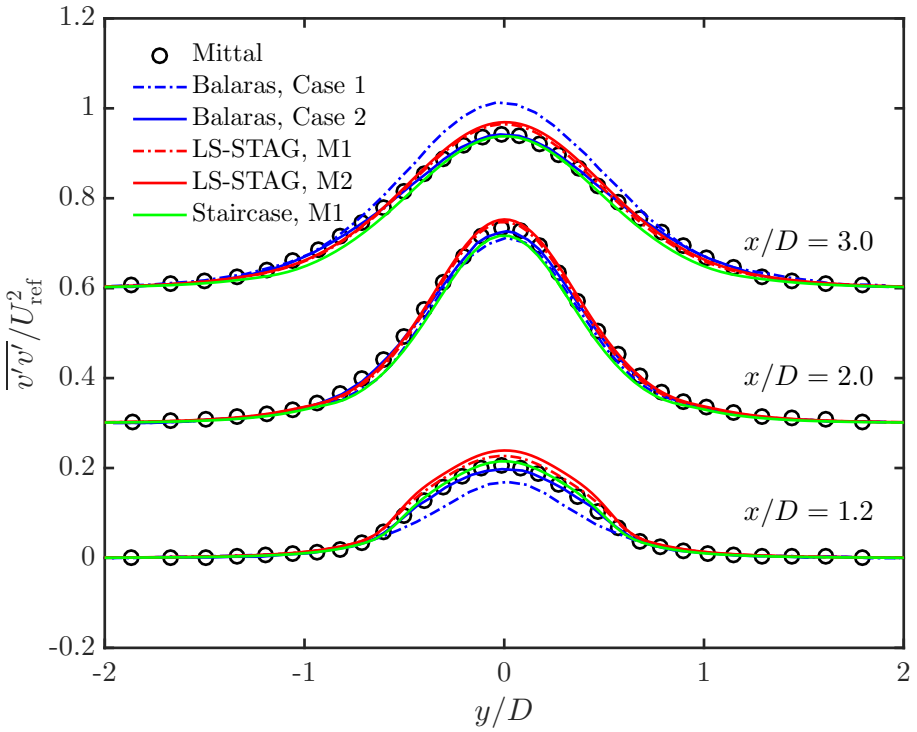


Figure 4.24: time-averaged cross-flow velocity fluctuations at different locations in the wake of circular cylinder at $Re_D = 300$.

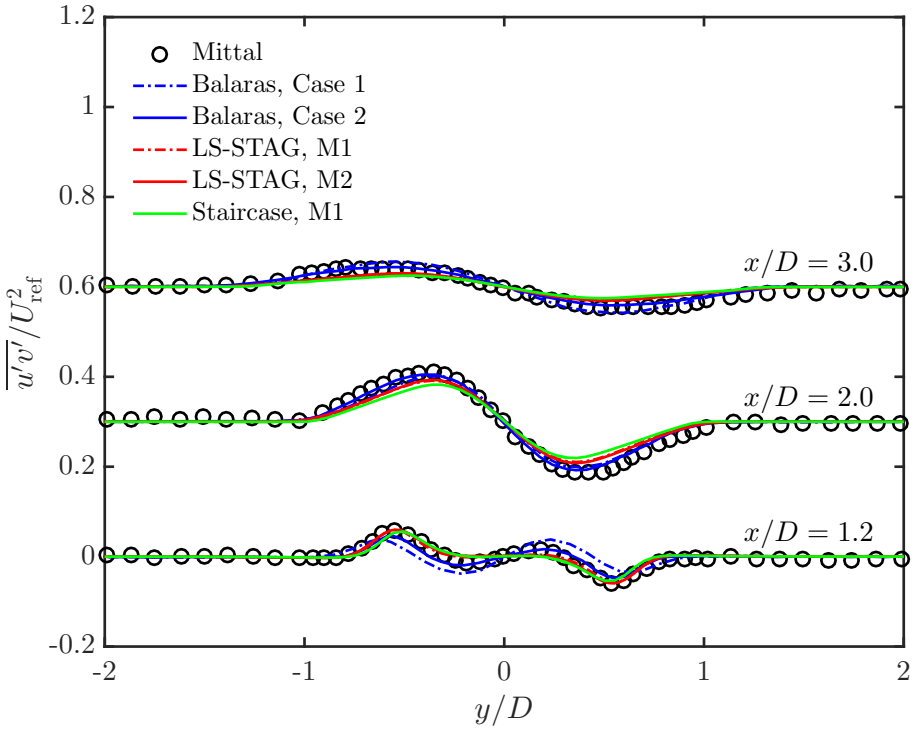


Figure 4.25: time-averaged Reynolds shear stress at different locations in the wake of circular cylinder at $Re_D = 300$.

Chapter 5

Validation of LS-STAG results with experimental data for non-Newtonian fluid flows

Contents

5.1 Flow of shear-thinning liquids in a 1:2 axisymmetric sudden expansion 115

5.1.1 Grid convergence study in expansion flow 119

5.1.2 Comparison of global parameters in expansion flow 122

5.1.3 Description of velocity profiles in different sections upstream and downstream of the abrupt expansion 127

5.2 Numerical simulation of granular flows in a Taylor-Couette cell . . 133

5.2.1 Study of convergence criterion in granular flows 133

5.2.2 Comparison of LS-STAG results with experimental data of granular flows in Taylor-Couette cell 143

5.1 Flow of shear-thinning liquids in a 1:2 axisymmetric sudden expansion

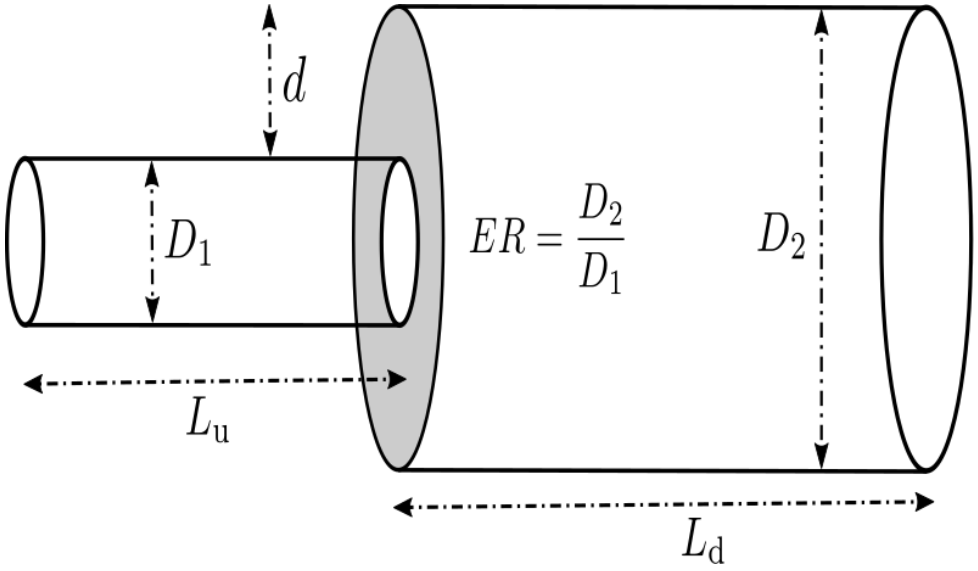
In this section, the LS-STAG method is applied to the flow of Newtonian and shear-thinning fluids in an axisymmetric sudden expansion geometry of ratio $ER = 2$, for which rheological characterization, experimental flow measurements through NMR[1] and FLUENT© simulations have been recently performed by Rigal [86] in LEMTA laboratory. Following this work, the geometry sketched in Figure 5.1(a) is composed by two pipes of diameter $D_1 = D_2/2 = 10\ mm$, with upstream ($L_\mathrm{u} = 150\ mm$) and downstream ($L_\mathrm{d} = 300\ mm$) lengths set sufficiently large

[1] Nuclear Magnetic Resonance

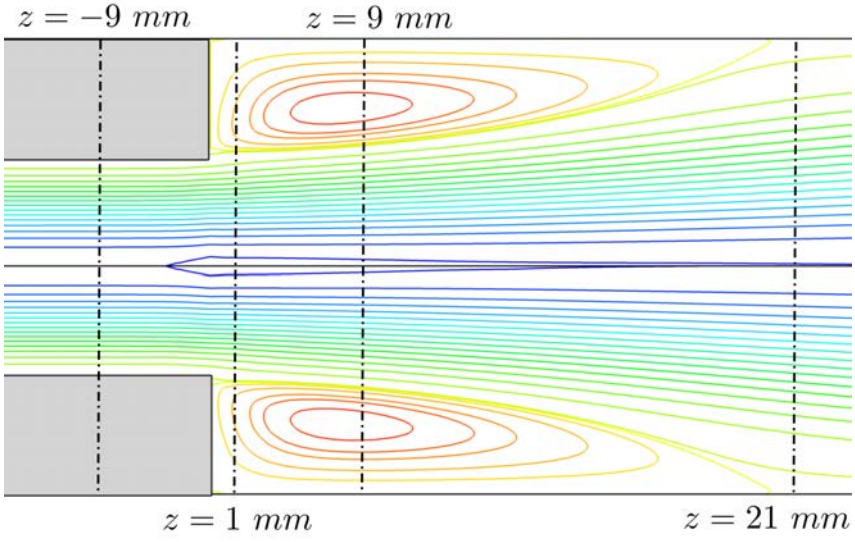
to ensure a fully developed flow and neglect the boundary and singularity effects. At the pipe inlet, an Ostwald velocity profile is imposed such that:

$$w(r) = \frac{3n+1}{n+1} U_{\text{ref}} \left[1 - \left(\frac{2r}{D_1} \right)^{\frac{n+1}{n}} \right], \tag{5.1}$$

in which U_{ref} is the inlet mean velocity.



(a)



(b)

Figure 5.1: Sketch of a circular duct with sudden expansion: (a) isometric view and (b) 2D view in the vicinity of singularity. The sudden expansion plane is placed at $z = 0$.

Following to Rigal's [86] work, three different fluid types, *i.e.* water and Xanthan solutions with 0.1% and 0.3% mass concentrations, are used in our numerical simulations. The Cross model is also employed in computation of the shear-thinning viscosity. The values of power-law index n required in inlet velocity and the parameters of Cross model are extracted from [86], see Table 1.1.

The level-set function $\phi(x, y, z)$ is built by using the CSG[1] method [35] whereby we can construct the complex domains by using the boolean operators (*e.g.* intersection, union) to combine the basic geometries (*e.g.* circles, hyperplanes, spheres) with sufficiently simple analytical level-set functions [35, 48]. The expansion geometry consists of circles and planes whose level-set functions are:

$$\phi_1(x, y, z) = \sqrt{x^2 + y^2} - R_1, \tag{5.2a}$$

$$\phi_2(x, y, z) = \sqrt{x^2 + y^2} - R_2, \tag{5.2b}$$

$$\phi_3(x, y, z) = z + \epsilon, \tag{5.2c}$$

$$\phi_4(x, y, z) = -z, \tag{5.2d}$$

where ϵ is an arbitrary small positive value. Accordingly, the upstream and downstream pipes are created from the intersection of regions whose level-set functions are $\phi_{\mathrm{u}}(x, y, z) = \max(\phi_1, \phi_3)$ and $\phi_{\mathrm{d}}(x, y, z) = \max(\phi_2, -\phi_4)$, respectively. Finally, the whole fluid domain is determined from the union of upstream and downstream pipes such that its level-set function is $\phi(x, y, z) = \min(\phi_{\mathrm{u}}, \phi_{\mathrm{d}})$. Due to the expansion plane, 25 new cut-cells are introduced in its location at cross-section $z = 0$ which are in fact the different arrangement of:

- basic types of cut-cells with one solid face in z direction, or
- Cartesian cells with one cut face in z direction.

All possible arrangement of these new cut-cells are illustrated in Figures 5.2 and 5.3. It is important to note that all discretization performed in Chapter 3 are still applicable to these new cut-cells. We only need to determine the surface area, involved in viscous fluxes, for each new cut-cell.

[1] Constructive Solid Geometry

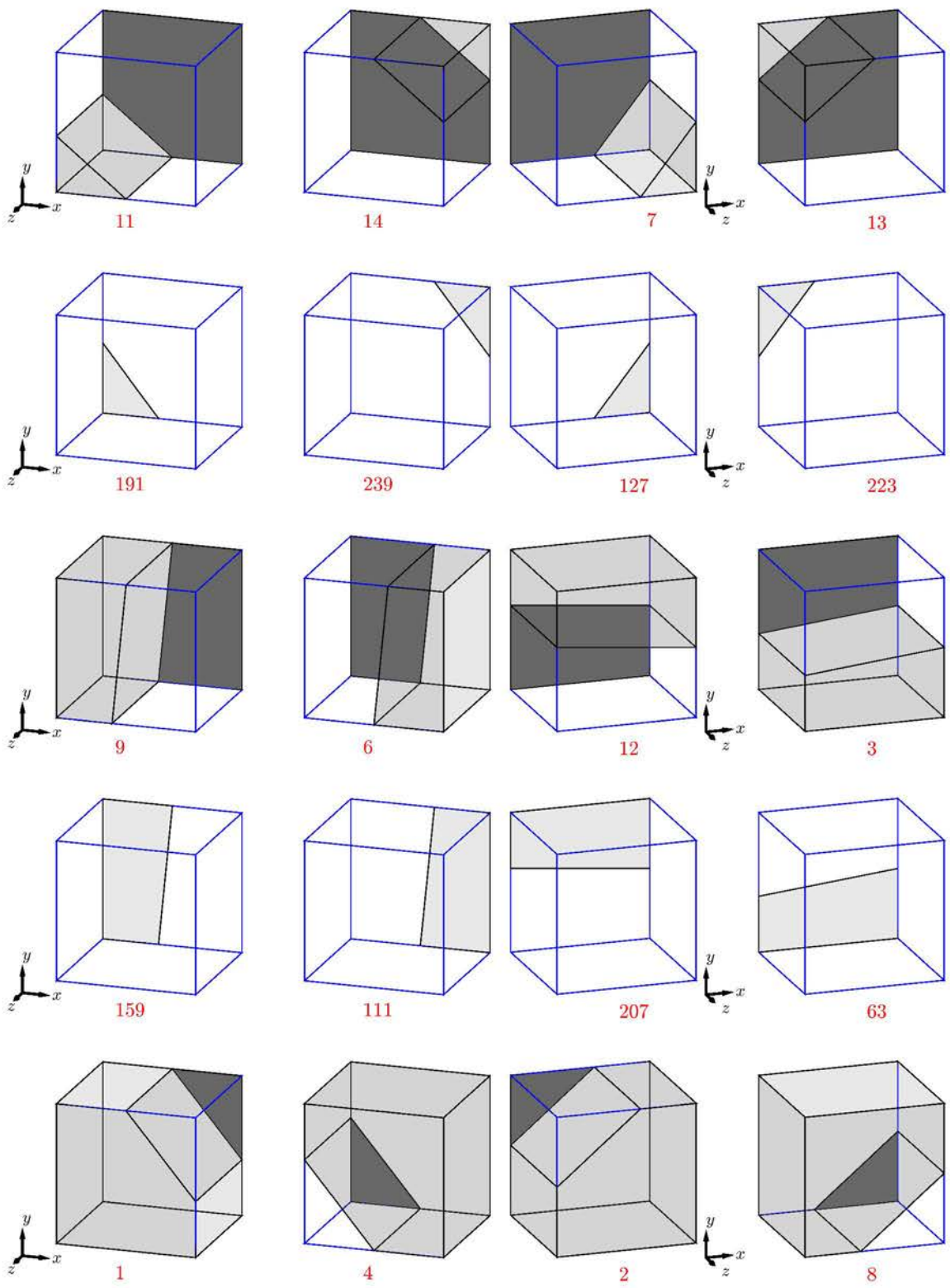


Figure 5.2: 25 distinct cells introduced in axisymmetric expansion flow.

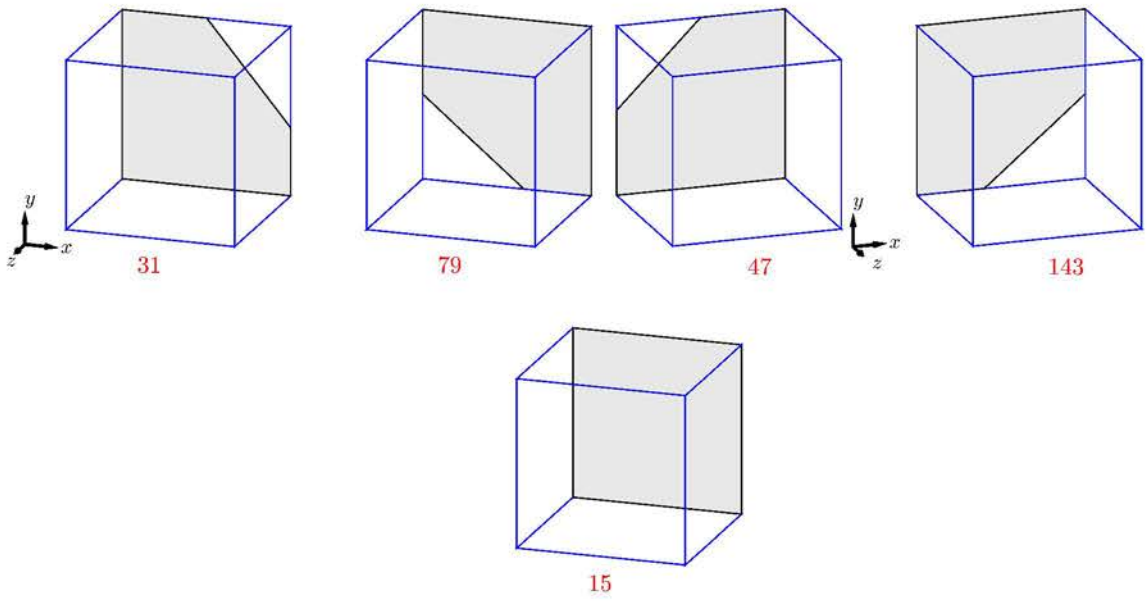


Figure 5.3: 25 distinct cells introduced in axisymmetric expansion flow (continued).

5.1.1 Grid convergence study in expansion flow

A grid refinement study is conducted on 4 refined meshes whose properties are reported in Table 5.1. In the axial direction (z), the domain is decomposed into 3 blocks. The grid size h_z is kept constant in the block with $z \in [-5\ mm, 15\ mm]$ next to singular expansion and is relaxed in the blocks at the upstream and downstream of singularity. A uniform grid clustering with the cell size of h is applied in the x and y directions.

Table 5.1: Main properties of the grids for axisymmetric expansion flow.

Mesh	$N_x \times N_y \times N_z$	Cell Types			h/D_1	h_z/D_1
		Cartesian	Cut-cell	Solid		
M1	$50 \times 50 \times 185$	254844	32336	175320	4.08×10^{-2}	16.7×10^{-2}
M2	$100 \times 100 \times 370$	2147480	130200	1422320	2.02×10^{-2}	8.33×10^{-2}
M3	$150 \times 150 \times 555$	7321320	294780	4871400	1.35×10^{-2}	5.56×10^{-2}
M4	$200 \times 200 \times 740$	17428564	523916	11647520	1.01×10^{-2}	4.17×10^{-2}

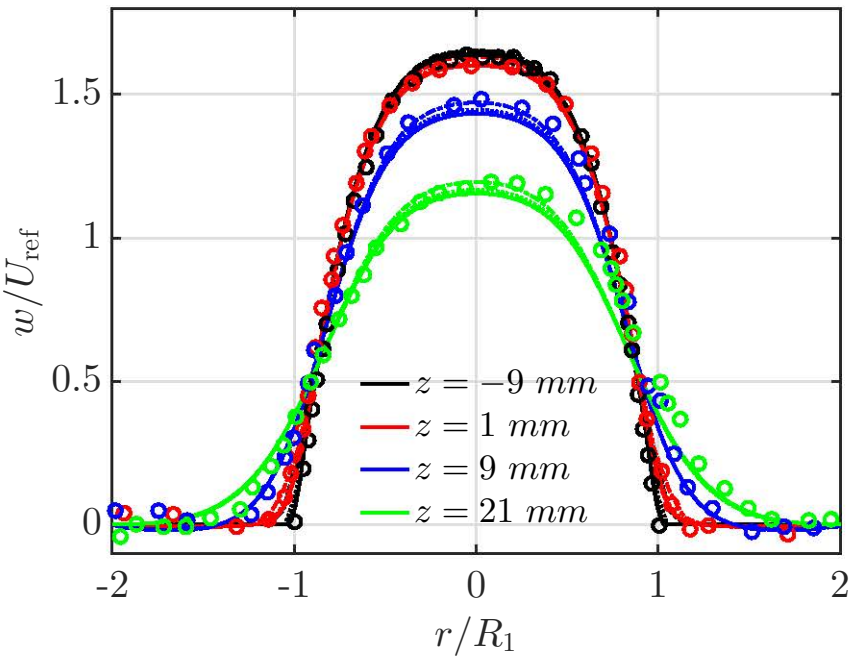
Figure 5.4 displays the LS-STAG axial velocity profiles for the Xanthan 0.1% and 0.3% flows at $Re_{D_1} = 56$. These profiles are demonstrated at various cross-sections before and after the singularity, see Figure 5.1(b). The experimental data reported by Rigal [86] are also given in the case of Xanthan 0.1%, see Figure 5.4(a). For both fluids, the negative velocities at $z = 1\ mm$ and $z = 9\ mm$ indicate the presence of a recirculation zone in these cross-sections which vanishes downstream at $z = 21\ mm$. Close inspection of the profiles at $z = 1\ mm$ next to

the abrupt expansion indicates insufficient grid resolutions for the 2 coarsest meshes (*i.e.* M1 and M2) especially in the vicinity of abrupt expansion. However, for the finer meshes M3 and M4, the velocity profiles are independent of the grid size. In addition, it can be seen that for Xanthan 0.1% flow, LS-STAG results shows an excellent agreement with the experimental data measured by NMR velocimetry [86]. Note that no experimental data is reported in Rigal [86] for Xanthan 0.3% at this Reynolds number. In the following, a detailed grid study is investigated by comparing the main flow parameters such as the vortex length and intensity, $\cdots$.

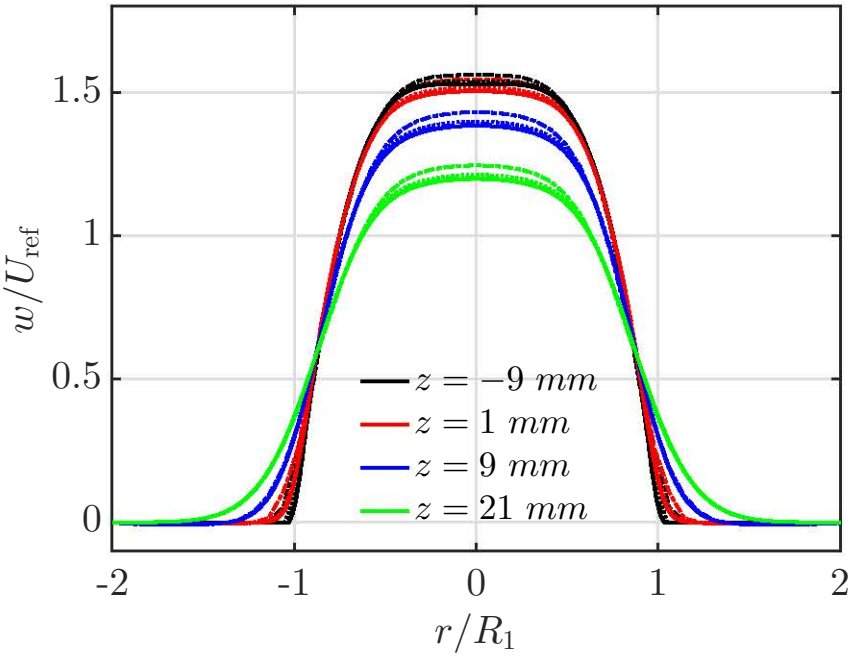
The main feature of expansion flows is the presence of a recirculating vortex caused by the separation of fluid flow at the singularity and its reattachment at the wall further downstream. For non-Newtonian fluids, the vortex characteristics in axisymmetric geometries have been studied experimentally in [44, 76] and numerically in [45, 71]. The quantities of interest are the vortex length L_v, its intensity $\Delta\psi$, the establishment length L_e and the Couette correction C. The size of recirculation zone L_v is equal to the length of area bounded by the streamline which separates the positive velocities of the main flow from the negative velocity of the secondary flow. The intensity of vortex $\Delta\psi$ is characterized by the maximal value of the stream function ψ in the recirculation zone which is the difference of the stream function values at the center of vortex and on the pipe wall, $\Delta\psi = \psi_\mathrm{v} - \psi_\mathrm{wall}$. The flow establishment length (L_e) is the downstream distance to the expansion plane from which the axial velocity becomes constant. The establishment length is defined as the location where difference between the axial and outlet velocities at the centerline become less than to 2%. All these quantities are normalized by the step-height $d = (D_2 - D_1)/2$. In addition, the Couette correction factor C represents the pressure drop across the expansion due to the singularity and is determined such that:

$$C = \frac{\Delta P - \Delta P_\mathrm{fd}}{2\tau_\mathrm{wall}}, \tag{5.3}$$

where ΔP is the total pressure drop computed in the fluid domain, ΔP_fd corresponds to the fully-developed pressure difference assuming the flow is fully developed in the whole of each channel, and τ_wall is the wall-shear stress. The values of these parameters are summarized in Table 5.2 for various meshes (M1–M4) at $Re_{D_1} = 56$.



(a)



(b)

Figure 5.4: Grid convergence of axial velocity profiles at different cross-sections of the duct for Xanthan 0.1% flow at $Re_{D_1} = 56$. The circle (○) denotes the experimental results of Rigal [86] while the dash-dotted (-·-), dotted (··), dashed (--) and solid (–) lines represent LS-STAG solutions on meshes M1 to M4 respectively.

Table 5.2: Grid convergence study of the dimensionless vortex length (L_v/d), vortex intensity $\Delta\psi^* = \Delta\psi/(\pi \mathrm{d}^2 U_\mathrm{ref})$ and establishment length (L_e/d) and Couette correction factor (C) at $Re_{D_1} = 56$.

Fluid type	Mesh	L_v/d	$\Delta\psi^* \times 10^3$	L_e/d	C
Water	M1	5.05	15.86	14.33	-2.60
	M2	5.00	15.32	14.17	-1.92
	M3	4.97	15.23	14.11	-1.76
	M4	4.96	15.11	14.08	-1.66
Xanthan 0.1%	M1	4.25	6.24	15.33	-1.19
	M2	4.10	5.68	15.33	-0.98
	M3	4.04	5.55	15.56	-0.90
	M4	4.00	5.43	15.58	-0.89
Xanthan 0.3%	M1	4.37	2.98	21.00	-6.69
	M2	4.16	2.44	21.00	-6.52
	M3	4.08	2.27	21.22	-6.44
	M4	4.03	2.16	21.33	-6.44

It is tempting to conclude nearly grid independent results are obtained on grid M3 with discrepancies below 1% for vortex length L_v, establishment length L_e and Couette correction factor C, and below 3% for the vortex intensity $\Delta\psi$ compared to the M4 results. It can be also observed that the higher is the shear-thinning index, the weaker is the intensity of vortex. In effect, since the drop in velocity gradients is more pronounced in the recirculation zone for high shear-thinning fluids, the bulk flow is able to drive the secondary flow at a lower rate. Furthermore, we observe that the shear-thinning behavior retards the flow establishment after the sudden enlargement. In other words, the establishment length increases with the shear-thinning index, thus the shortest length is related to water (Newtonian fluids) and Xanthan 0.3% (highly shear-thinning fluid) owns the longest one. We will see later that the flow establishment is directly related to the size of recirculation zone, which is not affected by the shear-thinning properties, since it is the same for water and Xanthan solutions. A similar trend has also been observed in the numerical computations of Halmos *et al.* [45] for power-law fluids and the experiments of Pak *et al.* [76] for inelastic fluids in the laminar regime. However, the vortex length and Couette correction factor do not obey a special trend based on the power-law index.

5.1.2 Comparison of global parameters in expansion flow

More insight in the vortex characteristics is given by varying the Reynolds number, with computations performed at $Re_{D_1} = 9$, 22, 56, 77.5, 100, 160 and 217 on the M3 mesh. Figure 5.5 shows the length of recirculation zone predicted by the LS-STAG code and the axisymmetric FLUENT© computations of Rigal [86] with respect to the generalized Reynolds number, Re_gen.

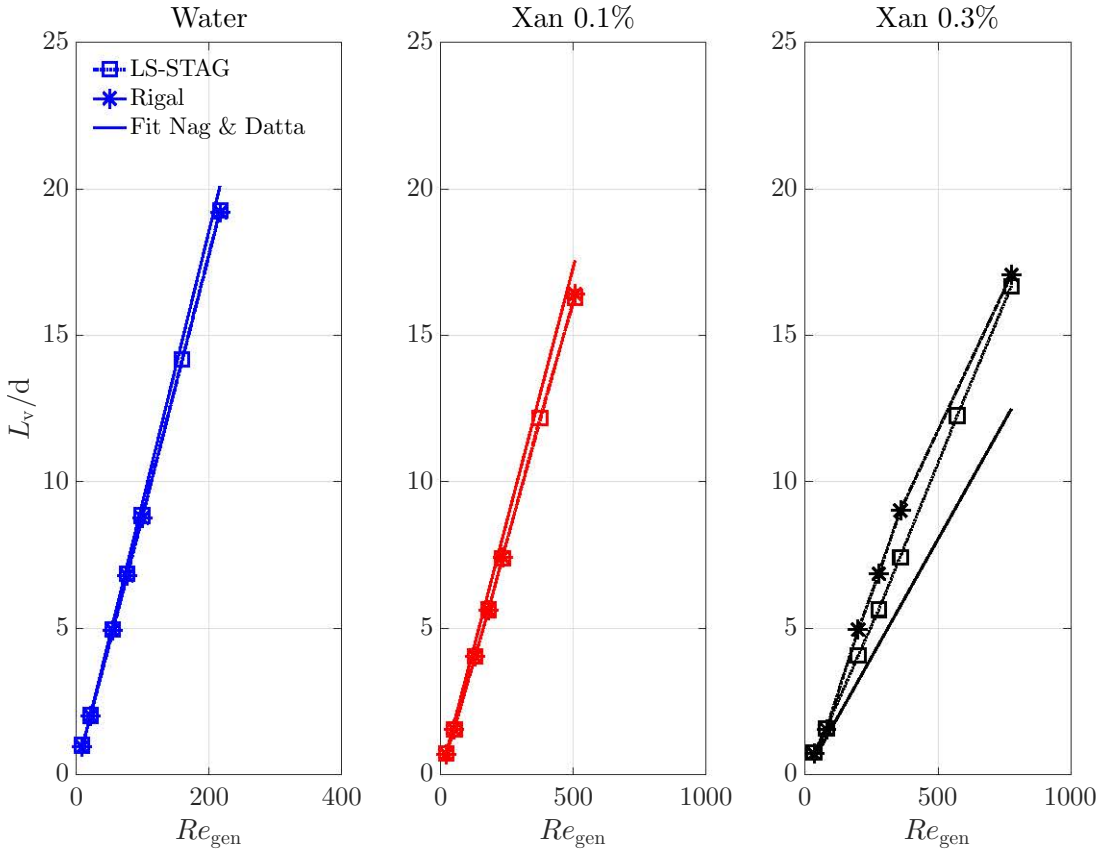


Figure 5.5: Variation of recirculation zone length against generalized Reynolds number.

According to the nag and Datta [71], Re_{gen} is determined such that:

$$Re_{\text{gen}} = 2^{3-n} \left(\frac{n}{3n+1}\right)^n Re_{D_1},$$

where n is the power-law index given in Table 1.1. For a given shear-thinning index, a monotonic increment in reattachment length L_v is observed as the Reynolds number increases. Similar behavior is also reported in the literature *e.g.* [45, 71, 61, 5]. According to Halmos *et al.* [45, 44], the length of recirculation zone is a linear function of Re_{gen} for all fluid types. Deviation from linearity for the FLUENT© computations of the Xanthan 0.3% flows may be attributed to the grid under-resolution [86]. In addition, the empirical curve fitting of Nag and Datta [71] has been added to this figure. This mathematical correlation predicts a linear growth of the recirculation zone for the expansion ratio $ER = 2$ as:

$$\frac{L_v}{d} = \left(0.0297\, n + 0.0629\, n^2\right) Re_{\text{gen}}. \tag{5.4}$$

Nag and Datta [71] stated that their fit is valid only for the range of $Re_{\text{gen}} \in [50, 200]$. It explains why our computational data are in an excellent agreement with the reported fit at lower Reynolds numbers, while the results are deviated from the fit as the Reynolds number increases. This trend is more perceptible for Xanthan 0.3% computations at $Re_{D_1} \geq 56$ which

corresponds to $Re_{\text{gen}} \gtrsim 200.1$. In addition, since inertia becomes important in higher Reynolds number, the results may deviate from linearity. For non-Newtonian cases, this nonlinearity becomes more distinctive as the index n decreases from unity. According to Nag and Datta [71], the non-dimensional recirculation length has a linear form of:

$$\alpha_{\text{v}} = \frac{L_{\text{v}}/\text{d}}{Re_{\text{gen}}}$$

at all expansion ratio (ER) where α_{v} is the function of ER and n. The prediction of α_{v} for LS-STAG method compared to references [86, 71] is given in Table 5.3 both for Newtonian and non-Newtonian cases at $ER = 2$. For Newtonian fluids, Macango *et al.* [61] and Badekas *et al.* [5] well fitted their experimental data by a linear curve as $\alpha_{\text{v}} = 0.0887$ and 0.0912, respectively. As seen, LS-STAG provides satisfactory results compared to their reference data. It can be also observed that the values of α_{v} decreases as the power-law index decreases. Note that for Xanthan 0.3% flow, the discrepancy between LS-STAG and reference [71] results can be interpreted by the restriction condition of correlation (5.4).

Table 5.3: Comparison of LS-STAG results and reported data of [86, 71] for water and Xanthan solutions with 0.1% and 0.3% mass concentration.

Source	α		
	Water	Xanthan 0.1%	Xanthan 0.3%
Macagno [61]	0.0887	–	–
Badekas [5]	0.0912	–	–
Nag and Datta [71]	0.0926	0.0347	0.0161
FLUENT© [86]	0.0879	0.0325	0.0222
LS-STAG	0.0881	0.0325	0.0216

Figure 5.6 displays the evolution of secondary vortex intensity $\Delta\psi$ with respect to the Reynolds number for various fluids. As discussed for $Re_{D_1} = 56$ in Table 5.2, the shear-thinning property of fluids tends to decrease the vortex intensity for a given Reynolds number. In fact, the drop of velocity gradient in recirculation zone results in an increase in viscosity and reduction in stresses. For the fluids with higher shear-thinning character, the increase of viscosity gradient in this region will be greater and therefore the vortex intensity decreases. Secondly, for a given shear-thinning index, a direct relationship is observed between the vortex intensity and Reynolds number such that the vortex intensity increases as the Reynolds number increases. However, this is not an unbounded augmentation; $\Delta\psi$ increases rapidly with the Reynolds number until reaching an asymptote at high Reynolds number. This behavior is attributed to the increase in rate-of-momentum transfer from the bulk flow to the recirculation zone, hence the secondary flow becomes more intense, refer to Halmos *et al.* [45]. Similar trend is also observed in [5, 93] for Newtonian flows with various expansion ratios. For highly shear-thinning case, Xanthan 0.3%, there is a disparity between LS-STAG and Rigal [86]. It is expected that LS-STAG method

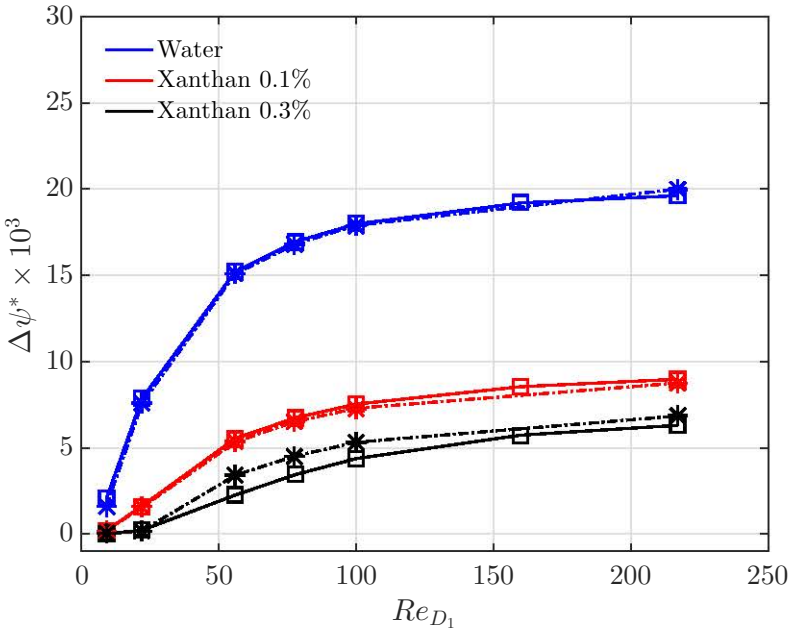


Figure 5.6: Evolution of vortex intensity with the Reynolds number: LS-STAG (-□-) versus FLUENT© (-·∗-·) [86] for three different fluids.

estimates more admissible results according to the previous comparison of reattachment length with the reference fit.

The evolution of non-dimensional establishment length L_e/d against the Reynolds number Re_{D_1} for the different fluid flows is demonstrated in Figure 5.7. It is observed that the establishment length increases with the shear-thinning index at a given Reynolds number. The monotonic growth of L_e with the Reynolds number is directly related to the size of vortex. In fact, bigger recirculation zone at higher Reynolds number allows the flow to decelerate more gradually, hence retarding the flow establishment is observed downstream of the singularity as the Reynold number increases. Additionally, the curves also reflect increasing the establishment length with decreasing the power-law index at a constant Reynolds number. In other words, as the shear-thinning character diminishes, the bulk flow reaches faster to a fully-developed state.

The effects of Reynolds number and power-law index on the Couette correction factor are demonstrated in Figure 5.8. For a given fluid, this factor decreases as the Reynolds number increases. Furthermore, at high Reynolds number, the Couette correction factor is a decreasing function of power-law index which may be related to the fact that the flow is becoming dominated by inertia. The positive Couette coefficients are obtained at low Reynolds numbers $Re_{D_1} = 9, 22$ for Xanthan 0.1% unlike all other cases in which this factor is always negative. The negative value of C indicates that the pressure loss through the expansion is less than the estimated for fully developed flow since the pressure recovery exceeds the viscous losses when the fluid goes through the expansion. In the meantime, for water, all results are fitted very well with a linear curve of the form $C = -0.029\, Re_{D_1}$ in which the maximum difference is less than 12%.

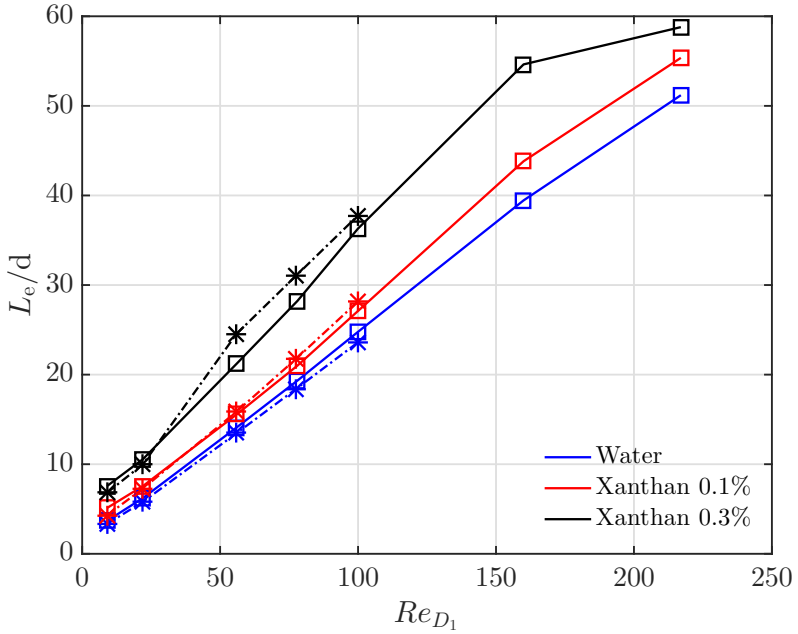


Figure 5.7: Evolution of establishment length with the Reynolds number: LS-STAG (-□-) versus FLUENT© (-·∗-·) [86] for three different fluids.

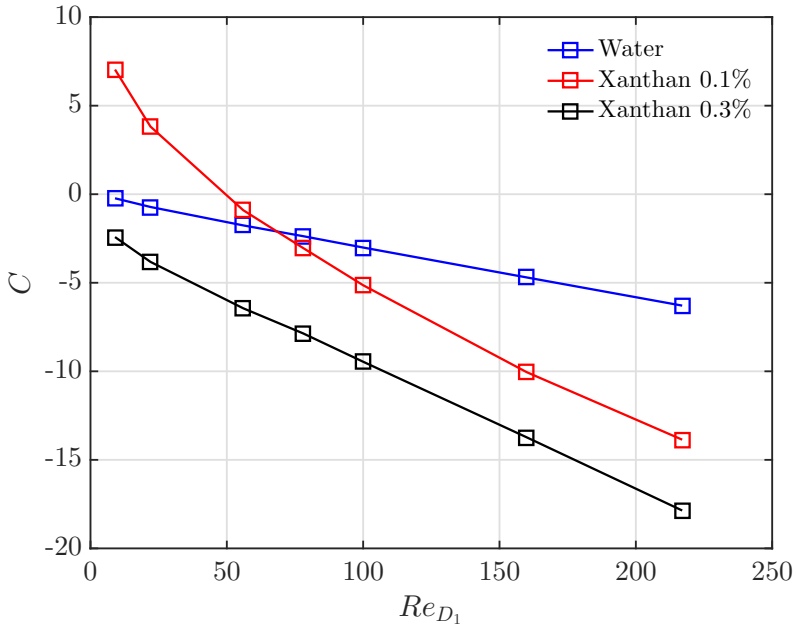


Figure 5.8: Variation of Couette correction factor against Reynolds number for three different fluids.

5.1.3 Description of velocity profiles in different sections upstream and downstream of the abrupt expansion

In this section, the axial velocity profiles predicted by LS-STAG method will be compared with experimental data of Rigal [86] at different sections. Figures 5.9 to 5.11 represent respectively the axial velocity profiles for water, Xanthan 0.1% and 0.3% solutions in the cross-section $z = -9\ mm$ prior to expansion. First, it is seen the LS-STAG prediction are in an excellent agreement with the experimental data extracted from Rigal [86] particularly for the shear-thinning cases. Note that the experimental data are not available for all Reynolds numbers. In addition, the effect of shear-thinning behavior on the shape of velocity profiles is manifested by their flattening such that the axial velocity profiles become more monotonic in the center and steeper near the wall as the power-law index increases. We have also determined the ratio of w_c/U_{ref}, that expresses the proportion of axial velocity at centerline w_c to the inlet bulk velocity for different fluids. For water, this ratio is almost constant, $w_c/U_{ref} = 1.99$, at different Reynolds numbers which shows only 0.5% error compared to the theoretical value of $w_c/U_{ref} = 2$. For Xanthan 0.1%, the ratio of $w_c/U_{ref} \simeq 1.64$ is obtained at different Reynolds numbers which is close to the analytical value of 1.7 (the maximum error is 3.7%). The most discrepancy is observed for the highly shear-thinning case, *i.e.* Xanthan 0.3% at $Re_{D_1} = 217$ with 5% deviation from the analytical value ($w_c/U_{ref} = 1.5$). Note that similar FLUENT© results are reported in [86].

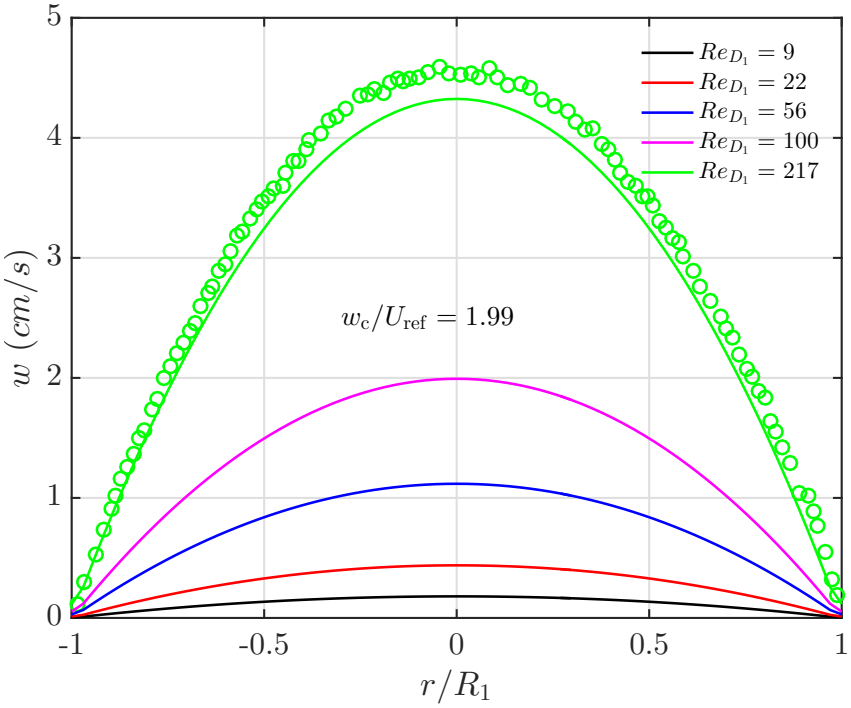


Figure 5.9: Axial velocity profiles of water at $z = -9\ mm$ for different Reynolds numbers: circles ($\circ$) for experimental data [86] and solid lines (–) for LS-STAG computation. For all Reynolds number the ratio of w/U_{ref} is equal to 1.99.

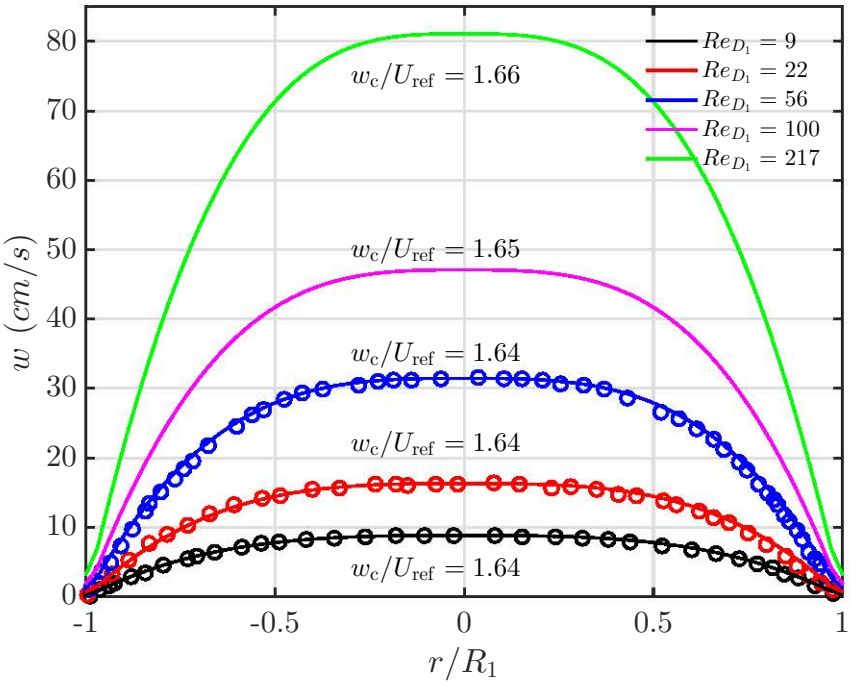


Figure 5.10: Axial velocity profiles of Xanthan 0.1% at $z = -9\ mm$ for different Reynolds numbers: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation.

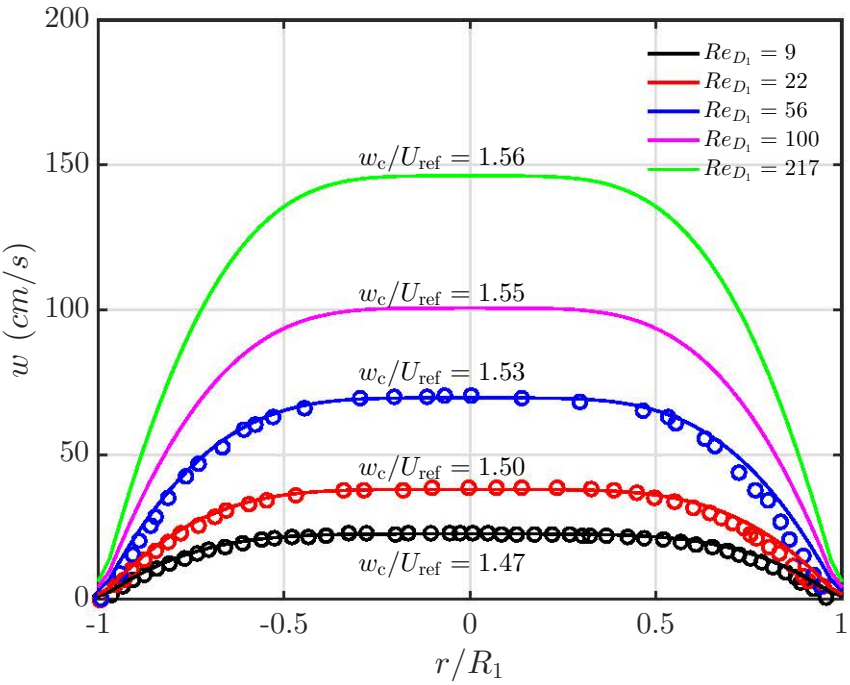


Figure 5.11: Axial velocity profiles of Xanthan 0.3% at $z = -9\ mm$ for different Reynolds numbers: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation.

We will subsequently inspect the axial velocity profiles for different cross-sections, $z = 1\ mm$, $9\ mm$ and $21\ mm$, downstream the pipe at different Reynolds numbers. Figures 5.12 to 5.17 display the axial velocity profiles for the water, Xanthan 0.1% and 0.3% flows at $Re_{D_1} = 9$ and 100. The LS-STAG predictions show excellent agreement with the experimental measurements [86] at all cross-sections which confirms the accuracy and robustness of our method. As discussed in Section 5.1.1, the negative velocities indicate the presence of recirculation zone at a cross-section. Thus, we can see an inflection point on the velocity profiles in the vicinity of recirculation zones. For all types of fluids at $Re_{D_1} = 9$, the recirculation zone is found only at the section $z = 1\ mm$ and the flow is fully established at $z \geq 9\ mm$. In the meantime, the vortex length is in direct relationship with the Reynolds number, so this region is longer at higher Reynolds number. As an example, we can clearly see that the recirculation zone extends up to $z = 9\ mm$ and $z = 21\ mm$ at $Re_{D_1} = 56$ (Figures 5.4(a) and 5.4(b)) at $Re_{D_1} = 100$ (Figures 5.15 to 5.17), respectively.

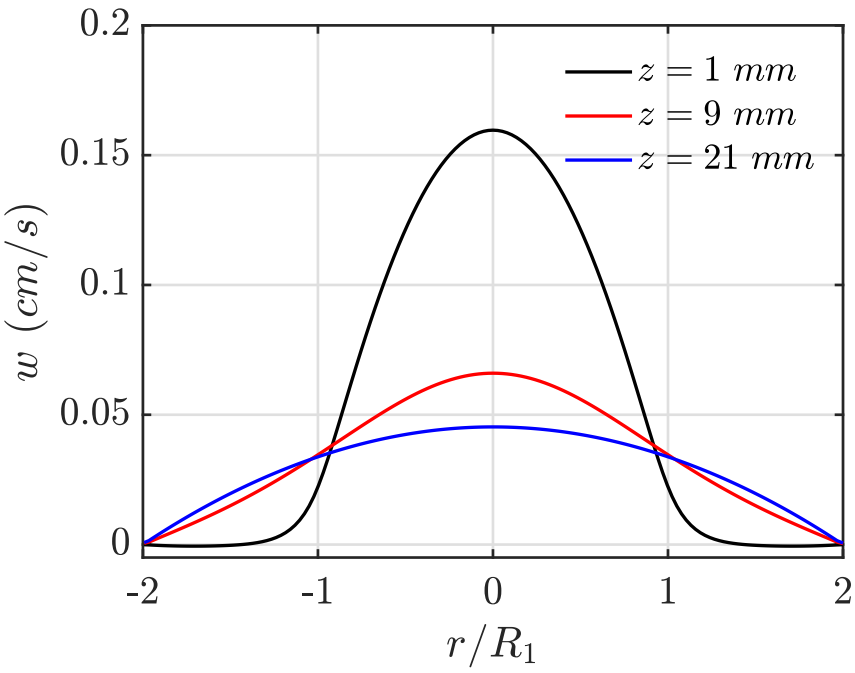


Figure 5.12: Axial velocity profiles at different cross-sections after the singularity for water at $Re_{D_1} = 9$.

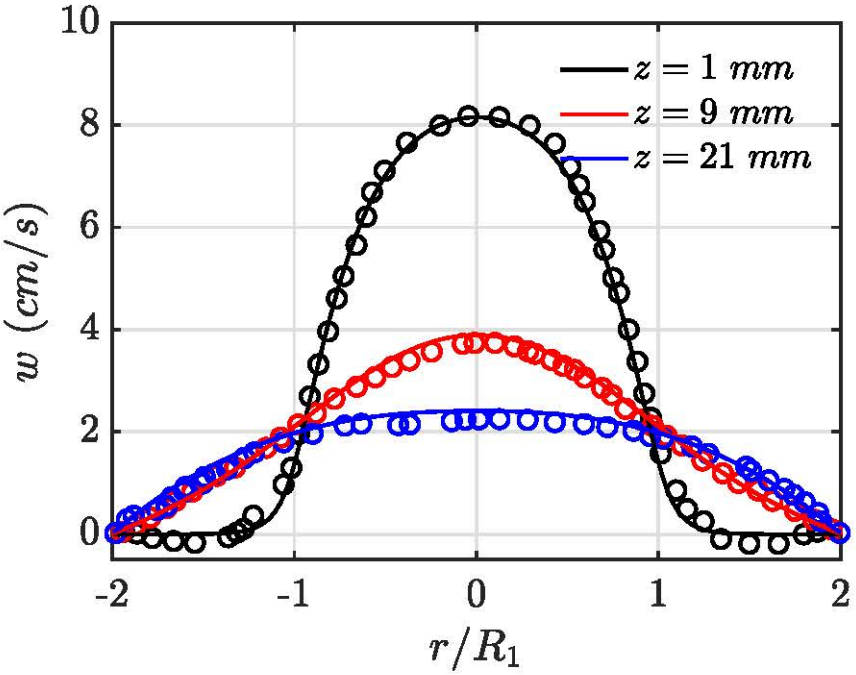


Figure 5.13: Axial velocity profiles at different cross-sections for Xanthan 0.1% flow at $Re_{D_1} = 9$: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation.

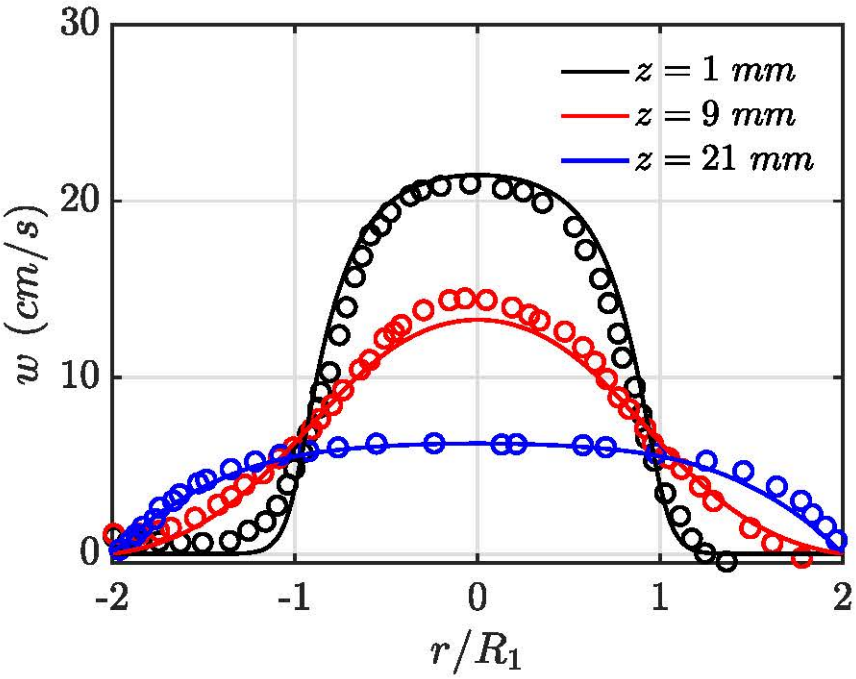


Figure 5.14: Axial velocity profiles at different cross-sections for Xanthan 0.3% flow at $Re_{D_1} = 9$: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation.

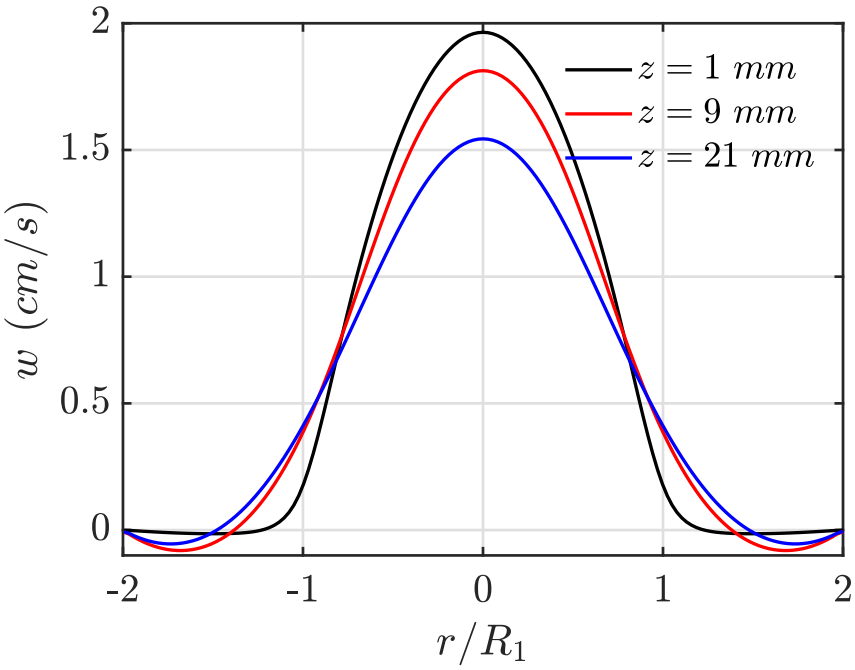


Figure 5.15: Axial velocity profiles in different cross-sections after the singularity for water at $Re_{D_1} = 100$.

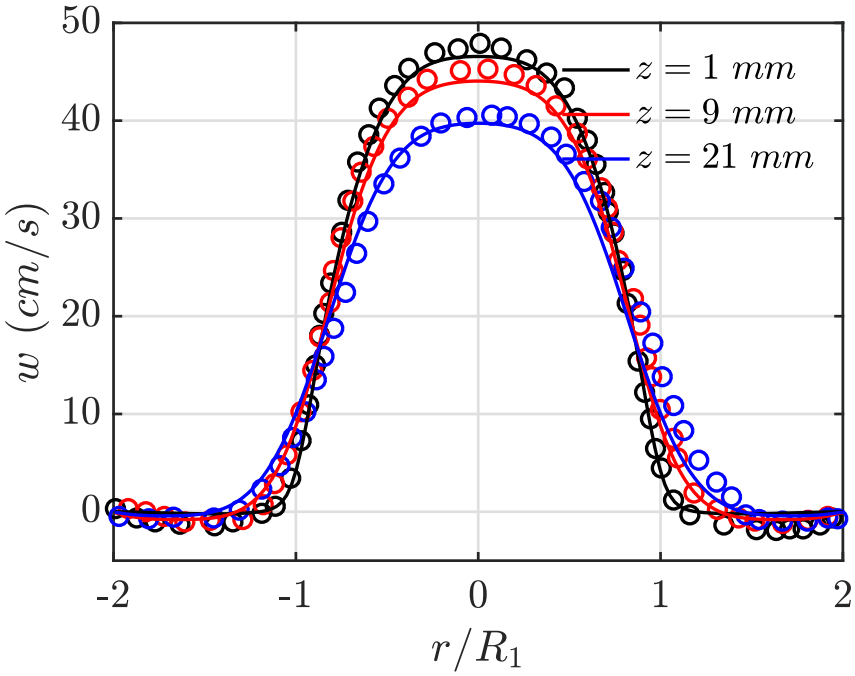


Figure 5.16: Axial velocity profiles in different cross-sections after the singularity for Xanthan 0.1% flow at $Re_{D_1} = 100$: circles (∘) for experimental data [86] and solid lines (–) for LS-STAG computation.

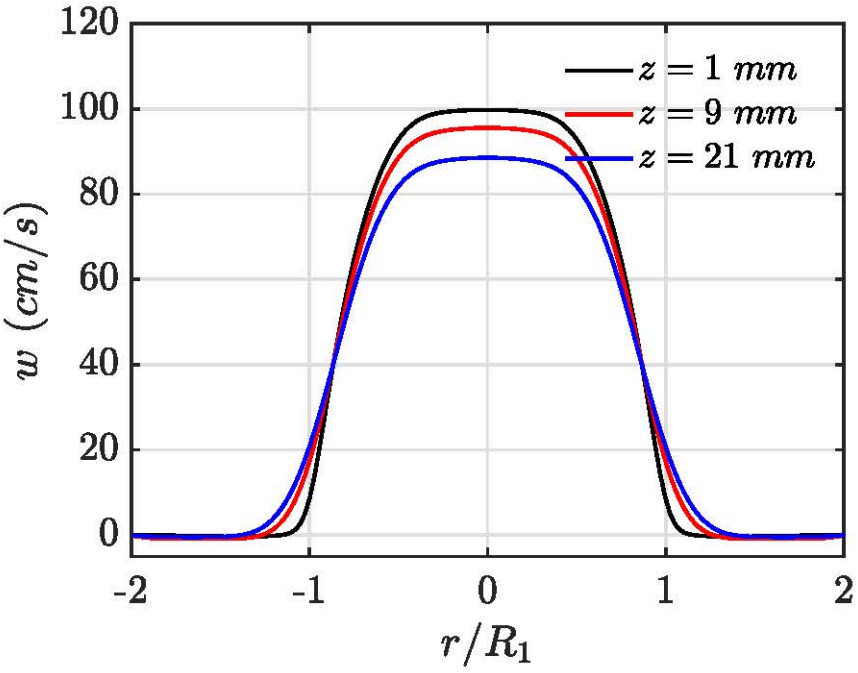


Figure 5.17: Axial velocity profiles in different cross-sections after the singularity for Xanthan 0.3% flow at $Re_{D_1} = 100$.

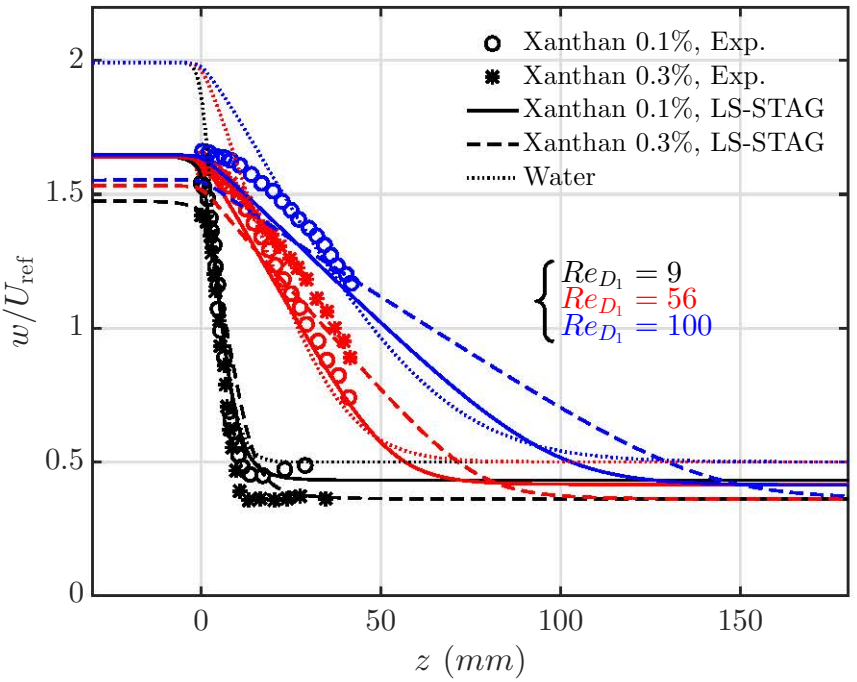


Figure 5.18: The axial velocity profiles at the centerline along the pipe at $Re_{D_1} = 9, 56$ and 100 for water, Xanthan 0.1% and 0.3%. The experimental data of Rigal [86] is also given for the shear-thinning fluids.

Finally, the effects of Reynolds number and shear-thinning behavior on the establishment length is described by Figure 5.18 in which the evolution of normalized axial velocity at the centerline is shown along the pipe at $Re_{D_1} = 9$, 56 and 100. The available experimental data reported by Rigal [86] are also given for Xanthan 0.1% and 0.3%. We can clearly see that for a given fluid, increasing the Reynolds number delays the flow establishment. As discussed earlier in this section, the shear-thinning character has an inverse proportion with the flow establishment such that a fluid with higher shear-thinning behavior has shorter establishment length which is more apparent at higher Reynolds numbers, see Figure 5.18. According to Rigal [86], there is a disparity between the numerical simulations obtained by FLUENT© and the experimental data. However, as seen here, the LS-STAG prediction matches very well with experimental data [86], most notably at lower Reynolds numbers.

5.2 Numerical simulation of granular flows in a Taylor-Couette cell

The aim of this section is to study the robustness of LS-STAG method in the simulation of steady granular flows in a Taylor-Couette cell. We will first require to determine a convergence criterion for the numerical simulations. For this purpose, we will compare different parameters of interest for two distinct test-cases in Section 5.2.1 to find an optimal residuals in which the results are well converged. Subsequently, the LS-STAG results acquired based on this criterion will be compared with available experimental data for three different test-cases reported by [38] in Section 5.2.2.

5.2.1 Study of convergence criterion in granular flows

In this part, we intend to investigate steady granular flows studied by Hanotin [46], to determine a convergence criterion in which the solution is sufficiently converged such that the acquired results are admissible. The experimental setup [46] is a Taylor-Couette cell sketched in Figure 5.19, with the following dimensions:

$$D_\mathrm{i} = 20\ mm, \quad D_\mathrm{e} = 40\ mm, \quad L_\mathrm{z} = 30\ mm, \quad d_\mathrm{g} = (D_\mathrm{e} - D_\mathrm{i})/2 = 10\ mm,$$

in which D_i and D_e are respectively the diameters of inner and outer cylinders and L_z is the length of cylinders and d_g represents the annular gap between two cylinders.

Following to [46], two different flows of *dry* granular particles (glass beads) with the particle density of $\rho_p = 2.4\ g/cm^3$ between two coaxial cylinders are numerically solved by LS-STAG code. The main properties of these two cases are listed in Table 5.4. For both cases, the outer cylinder is fixed while the inner one is rotating with angular velocity of ω_i. In our numerical simulations, the granular particles are characterized as a non-Newtonian fluid in which its viscosity behavior is computed by employing the Hanotin law (1.12) introduced in Section 1.3.3. In all test-cases studied in Sections 5.2.1 and 5.2.2, the solution is independent to the axial direction

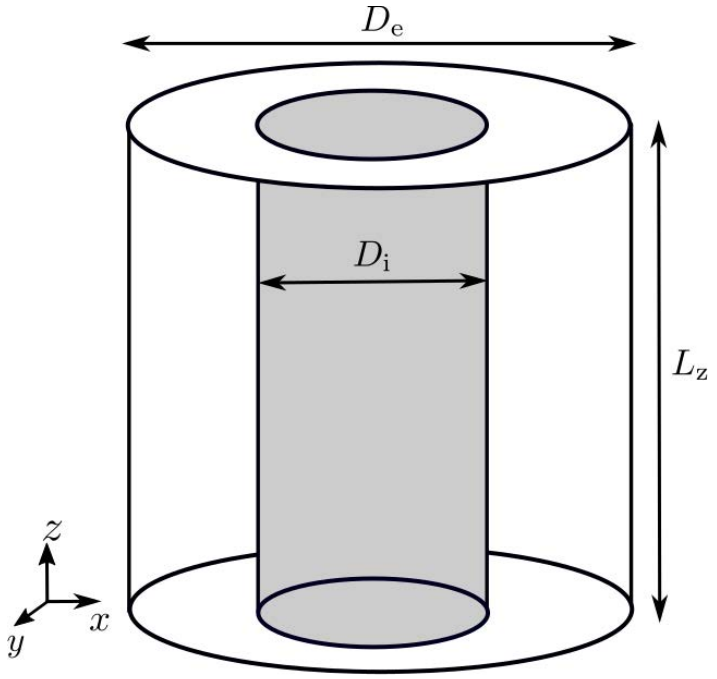


Figure 5.19: Schematics of a Taylor-Couette cell in which D_i and D_e are respectively the diameter of inner and outer cylinders and L_z is the length of cylinders.

z, thus they can be considered as a two-dimensional flow. However, since the LS-STAG code is used to process three-dimensional cases, it is necessary to add at least one grid cell in z direction in order to implement the boundary condition in this direction. For these flows, the periodic boundary conditions are imposed to the pipe ends in z direction. It is noteworthy to mention that $U_{ref} = D_i\,\omega_i/2$ and $D_{ref} = d_g$ are respectively set as the reference velocity and length in Reynolds number definition.

Table 5.4: The main properties of 2 different cases studied for the flows of dry granular particles in a Taylor-Couette cell.

Cases	ω_i (s^{-1})	Re_D	η_0 $(Pa.s)$	η_H $(Pa.s)$	$\dot{\gamma}_c$ (s^{-1})
Run 1	10	0.167	592	0	0.25
Run 2	0.01	4.21×10^{-6}	592	0	0.25

The residual is the most fundamental criterion of convergence as it directly evaluates the error in the solution for different variables such that the lower the residual value is, the more accurate the solution. We will subsequently investigate the evolution of physical quantities of interest as a function of the residuals for "Run 1". Figure 5.20 shows the evolution of residual on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations. As seen, the most intricate variable to converge is the pressure field which evolves slowly, thus these parameter is chosen as the reference convergence criterion for our simulations

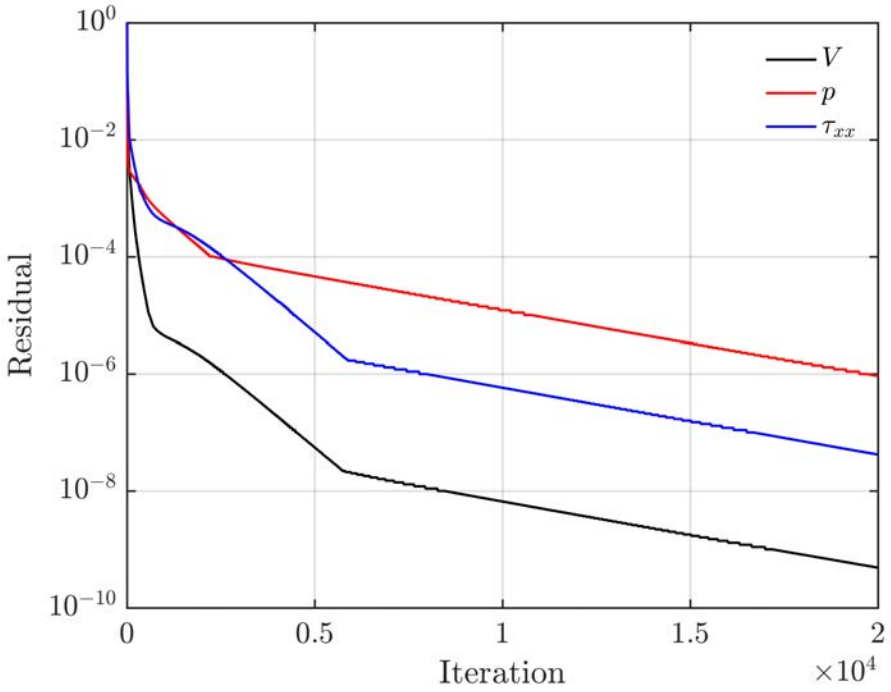


Figure 5.20: Evolution of residuals on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations.

of granular flows. For instance, the variations of residuals on the velocity and normal stresses are respectively about 10^{-6} to 10^{-8} and 10^{-4} to 10^{-6} in the interval from 2000 to 10000 iterations, while ϵ_p varies only from 10^{-4} to 10^{-5} in the same interval.

Although these graphs are rich in information, it is by no means sufficient to determine a relevant stop criterion of the simulation only based on these parameters. Thus, we will have a look on the evolution of an important global parameter in these flows *i.e.* inner cylinder torque, to see whether it tends towards a converged value. The evolution of the torque values exerted on the inner cylinder versus the time iterations is illustrated in Figure 5.21. As observed, the torque value is converged quickly after a few hundreds of iterations. To have a clearer view, the evolution of errors on the torque values is also given, see Figure 5.22. For each time-step, the error is defined such that:

$$res_{\mathrm{c}} = \frac{|M^n - M_{\mathrm{ref}}|}{|M_{\mathrm{ref}}|}, \tag{5.5}$$

where M_{ref} and M^n represent the values of torque for the converged solution and at time n, respectively. These errors decrease rapidly with the time iteration such that we reach an error of $res_{\mathrm{c}} \simeq 10^{-3}$ after only 2000 iterations. At this point, the pressure residual is approximately equal to 10^{-4} which will be shown later that is adequately small to be selected as a stop criterion for this test-case.

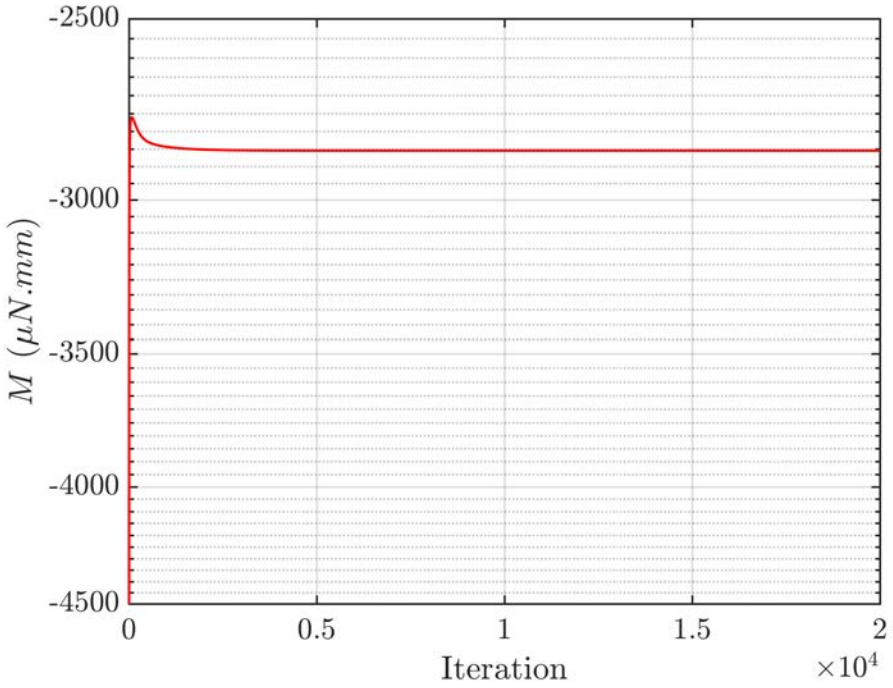


Figure 5.21: Evolution of the torque values exerted on the inner cylinder versus the time iterations in Run 1.

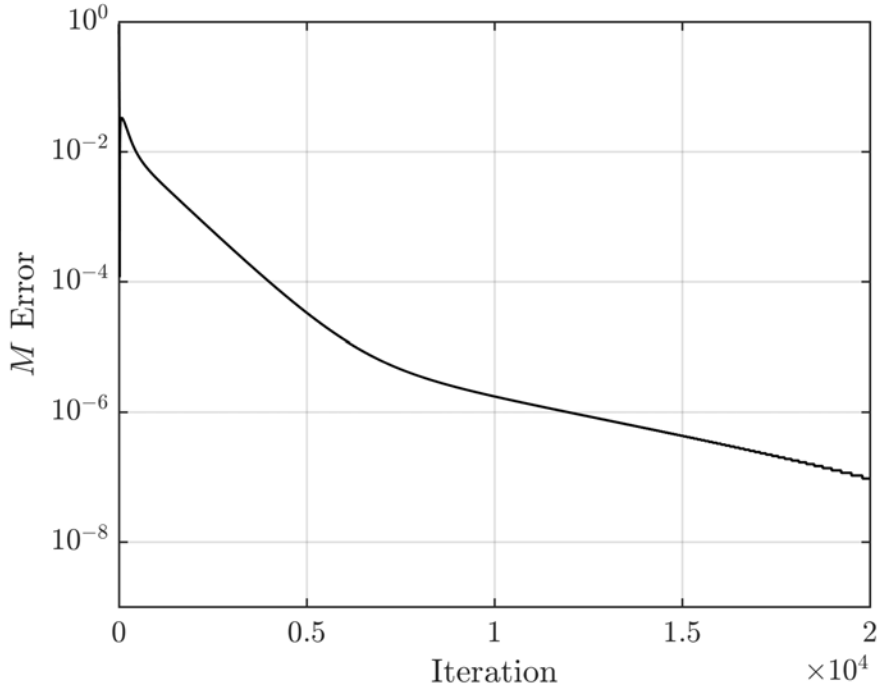


Figure 5.22: Evolution of the torque errors exerted on the inner cylinder versus the time iterations in Run 1.

As a final test of convergence criterion, we have compared the velocity and shear-rate profiles and the torque value of inner cylinder as a function of the pressure residual. Figures 5.23 and 5.24 display respectively the profiles of azimuthal velocity (V) and shear-rate ($\dot{\gamma}$) along the annular gap near the inner cylinder for different residuals of pressure. It is clearly observed that the profiles are almost identical for the pressure residual $\epsilon_p \geq 10^{-4}$. Similarly, the changes of torque values on the inner cylinder are negligible from $\epsilon_p = 10^{-4}$, see Table 5.5. Thus, this value (10^{-4}) can be considered as an optimal residual for the convergence in Run 1.

Table 5.5: The values of torque exerted on the inner cylinder in Run 1 for different values of ϵ_p.

ϵ_p	10^{-3}	10^{-4}	10^{-5}	10^{-6}
M $(\mu\, N.mm)$	2831.3	2852.5	2854.7	2854.7

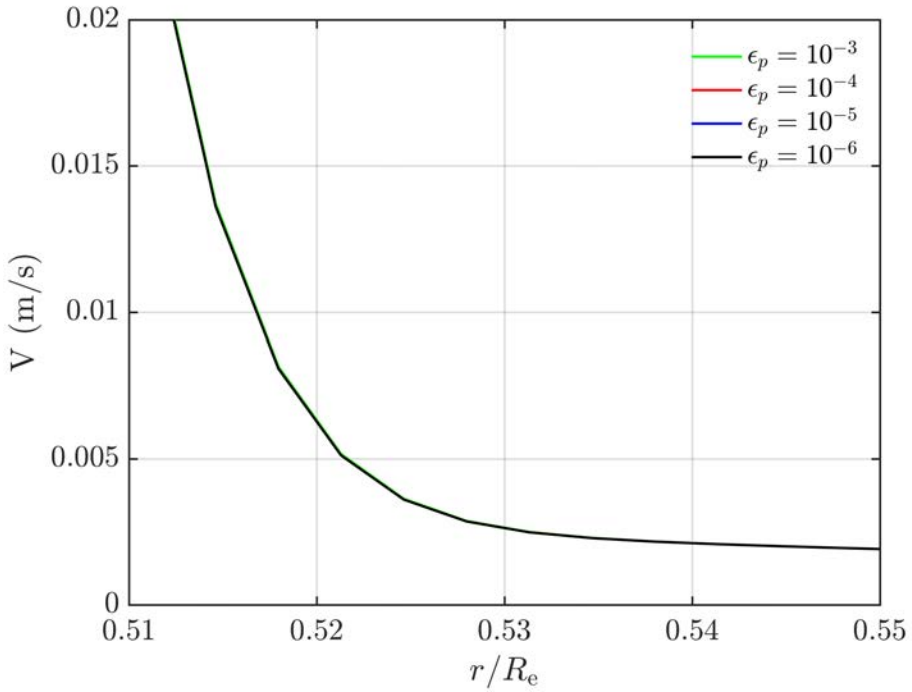


Figure 5.23: A close-up view of azimuthal velocity profiles around the inner cylinder at different residuals ϵ_p for Run 1.

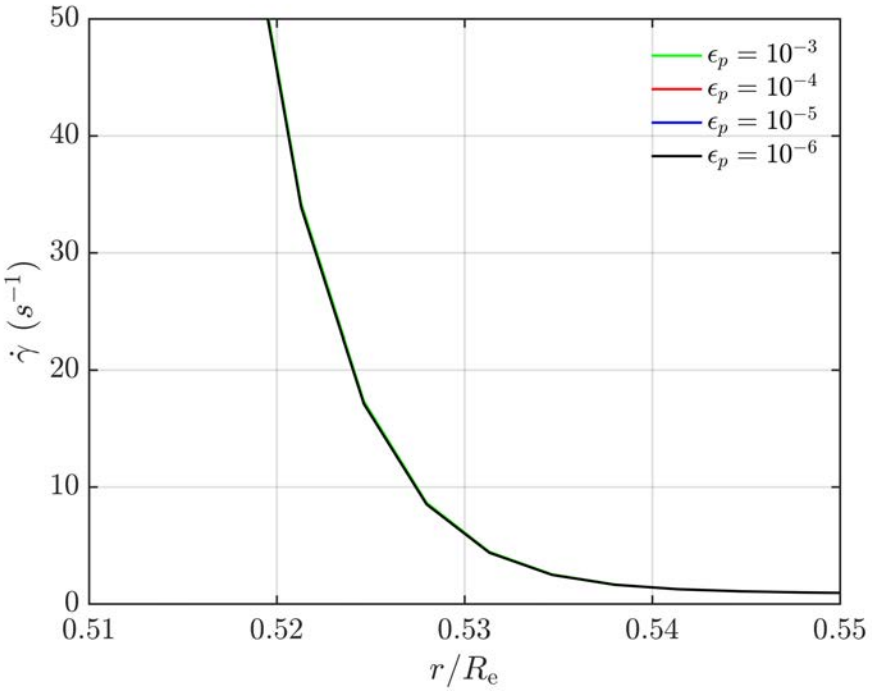


Figure 5.24: A close-up view of shear-rate profiles around the inner cylinder at different residuals ϵ_p for Run 1.

Now, let us take into account the second test-case at very low Reynolds number ("Run 2"). Similar examinations will be also performed for this test-case to determine a protocol for solving efficiently the granular flows in a Taylor-Couette cell. This protocol should be chosen such that we can obtain the satisfactory results and avoid excessively long calculation times. In this direction, we will first study the convergence behavior for different variables (V, p and τ_{xx}), see Figure 5.25. According to this figure, a monotonic and graduate convergence is observed for the pressure while for the other terms, the residuals oscillate excessively from the beginning of simulation which leads to have difficulty in convergence. In fact, at this very low Reynolds number, the small values of the grid size (h) may bring numerical difficulties to the inversion of linear system as it becomes ill-conditioned. Thus, we have observed these oscillatory behaviors in residuals on the velocity and normal stress. In order to alleviate this issue, we have performed similar simulations for a rescaled geometry. For the rescaled version, in order to avoid issue related to the small grid size, the dimensions of domain are chosen sufficiently bigger than the original test-case such like:

$$D_{\mathrm{i}} = 2\ m, \quad D_{\mathrm{e}} = 4\ m, \quad L_{\mathrm{z}} = 1\ m, \quad d_{\mathrm{g}} = 1\ m.$$

The residues on the velocity, pressure and normal stress for the rescaled geometry in Run 2 are shown in Figure 5.26. It is clearly seen that rescaling the geometry significantly improves the convergence behavior compared to the original test-case and prevents the strong oscillatory behavior seen in Figure 5.25. In addition, we can see that the errors on all variables evolve gradually with time iteration. For this test-case, the error of pressure term is again greater than other variables. Hence, we have determined the convergence criterion based on the pressure residual. We will now check whether the optimal value for Run 1 ($\epsilon_p = 10^{-4}$) is suitable for second test-case or not. For Run 2, this pressure residual is obtained after about 8000 iterations. At this point, the errors on velocity and normal stress are sufficiently small ($\epsilon_V \simeq 5 \times 10^{-8}$ and $\epsilon_{\tau_{xx}} \simeq 10^{-5}$). Thus, it seems reasonable to set it as a stop criterion. However, in order to ensure that the steady state solution is achieved, let us examine the evolution of torque and also errors on torque against time iteration, see Figures 5.27 and 5.28. As in Run 1, the torque of inner cylinder reaches a constant value only after a few hundreds iterations. However, the error on torque for Run 2 is relatively higher than that of Run 1. For the optimal value of pressure residual ($\epsilon_p = 10^{-4}$), the torque has an error of $res_{\mathrm{c}} \simeq 10^{-3}$ which is a satisfactory value for this parameter at such very low Reynolds number. The last test is to compare the azimuthal velocity and shear-rate profiles at different pressure residuals. Figures 5.29 and 5.30 display the close-up views of azimuthal velocity and shear-rate profiles around the inner cylinder at different residuals ϵ_p for the rescaled geometry in Run 2. Such like in Run 1, these profiles are almost identical at all pressure residuals. Thus, it is conjectured that $\epsilon_p = 10^{-4}$ is an optimal residual for pressure which will be subsequently used as the convergence criterion for our further computations carried out in Section 5.2.2.

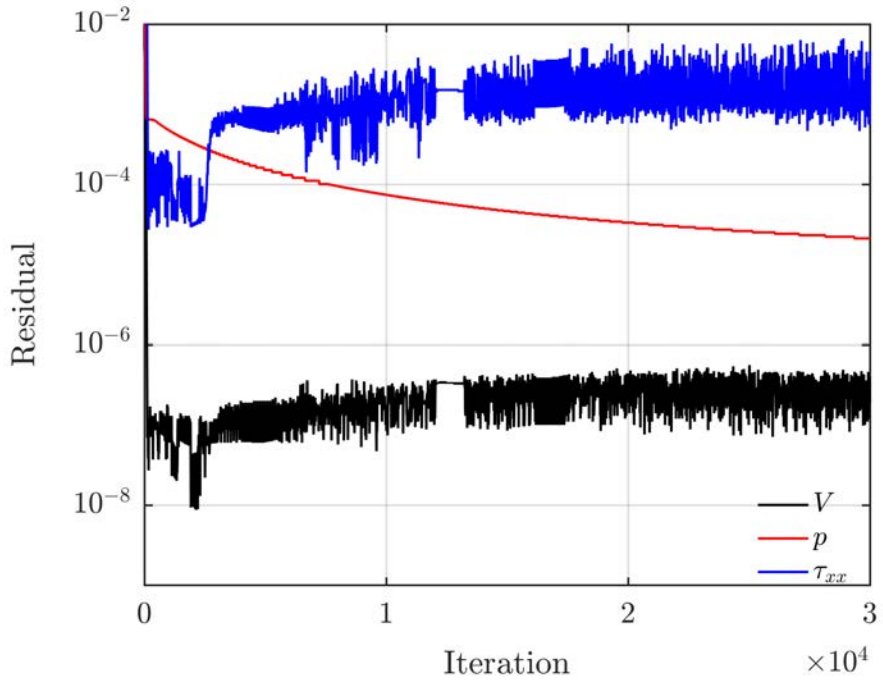


Figure 5.25: Evolution of residuals on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations.

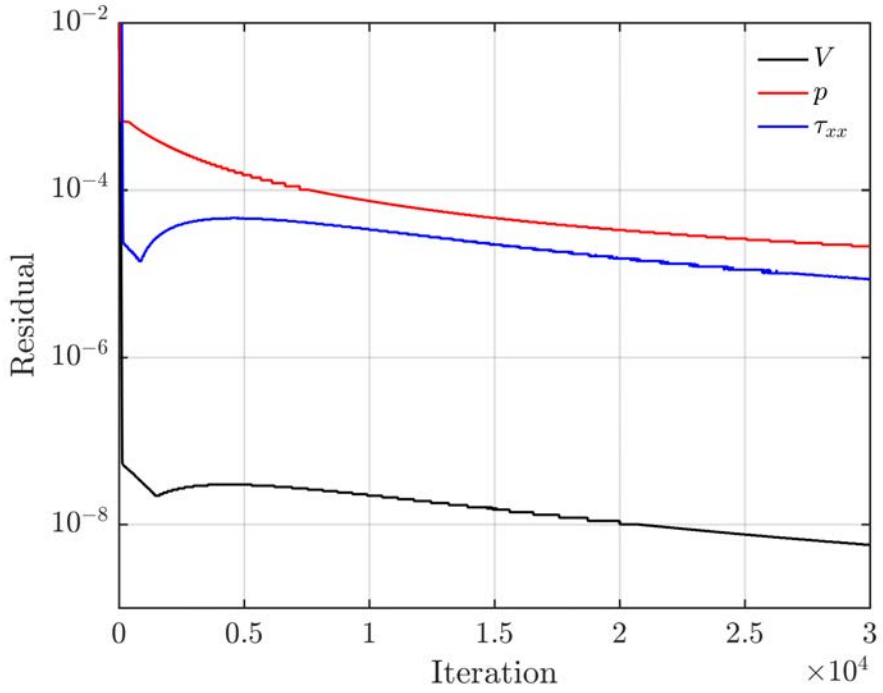


Figure 5.26: Evolution of residues on the azimuthal velocity (ϵ_V), pressure (ϵ_p) and normal stresses ($\epsilon_{\tau_{xx}}$) based on the number of time iterations.

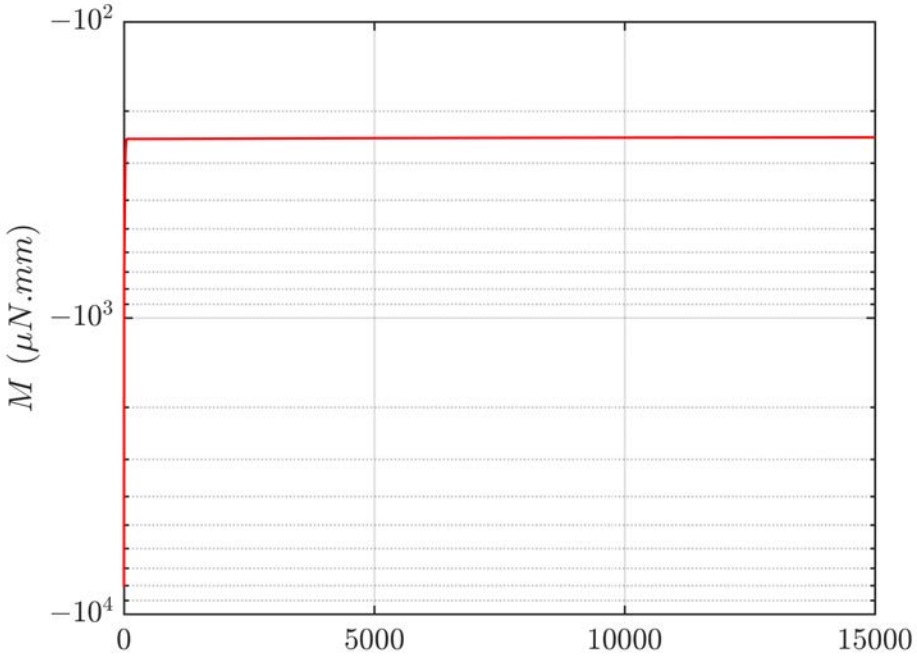


Figure 5.27: Evolution of the torque values exerted on the inner cylinder versus the time iterations in Run 2 for the rescaled geometry.

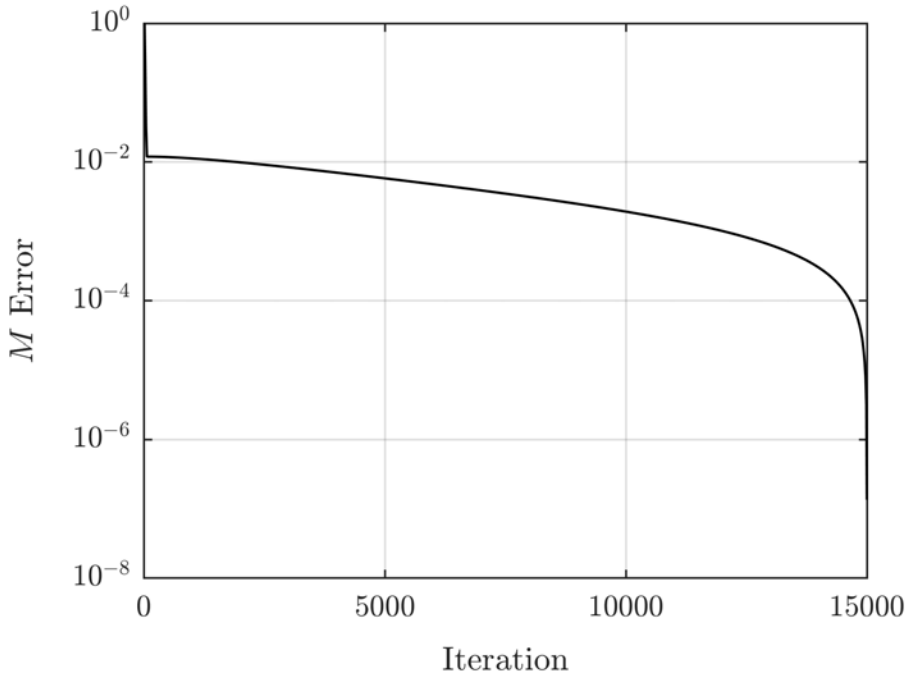


Figure 5.28: Evolution of the torque errors exerted on the inner cylinder versus the time iterations for the rescaled geometry in Run 2.

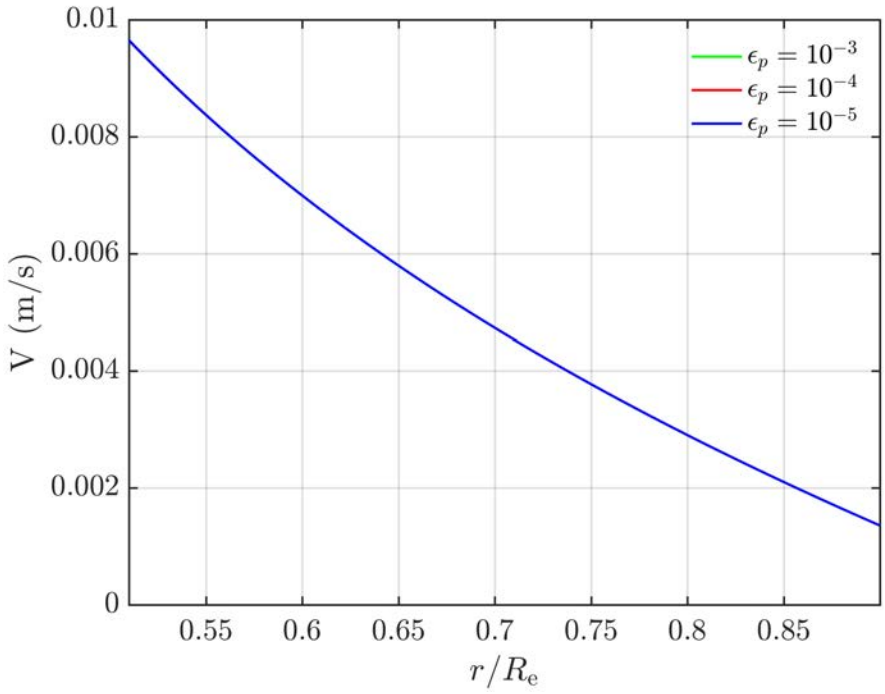


Figure 5.29: A close-up view of azimuthal velocity profiles around the inner cylinder at different residuals ϵ_p for the rescaled geometry in Run 2.

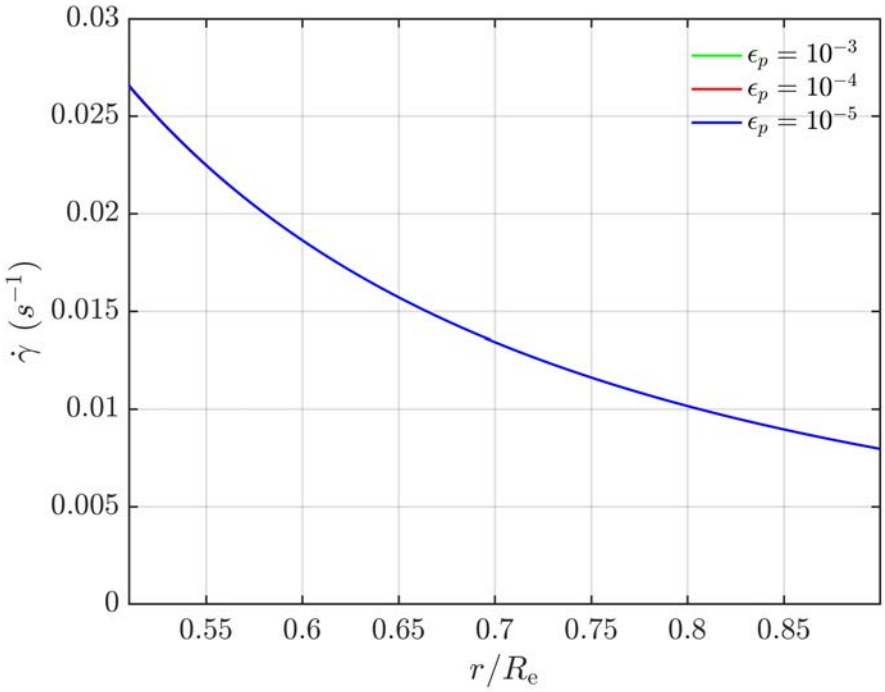


Figure 5.30: A close-up view of shear-rate profiles around the inner cylinder at different residuals ϵ_p for the rescaled geometry in Run 2.

5.2.2 Comparison of LS-STAG results with experimental data of granular flows in Taylor-Couette cell

In this section, we will apply the LS-STAG method to the flow of granular suspensions in a Taylor-Couette cell. The steady flow of granular suspensions in a stress-imposed rheometer (Discovery HR3) has been experimentally investigated by Gaudel *et al.* [38] in LEMTA Laboratory. The granular suspensions are made of borosilicate glass beads (Sigmund Lindner©) immersed in a Newtonian interstitial fluid (glycerol), refer to [38]. The density of suspension ρ_s depends on the fluid $\rho_f = 2.23\ g/m^3$ and particle $\rho_p = 1.26\ g/m^3$ densities such that:

$$\rho_s = \phi\,\rho_p + (1-\phi)\,\rho_f = 1.8517 g/m^3, \qquad \text{with:} \quad \phi = 0.61.$$

The PIV[1] measurements are also provided to capture the velocity profiles in the annular gap. The numerical simulations are performed for different cases summarized in Table 5.4. As in Section 5.2.1, the shear-thinning viscosity is determined by using the Hanotin law (1.12).

Table 5.6: The main properties of 3 different cases studied for the flows of granular suspensions in a Taylor-Couette cell. ω_i represents the angular velocity inner cylinder.

Cases	$\omega_i\ (s^{-1})$	Re_D	$\eta_0\ (Pa.s)$	$\eta_H\ (Pa.s)$	$\dot{\gamma}_c\ (s^{-1})$
Case 1	0.3	1.54×10^{-4}	7200	5.4521	8.4722×10^{-3}
Case 2	0.26	1.31×10^{-4}	10850	10.258	4.7825×10^{-3}
Case 3	0.03	1.56×10^{-6}	30000	6.1432	2.1393×10^{-3}

According to the Gaudel's recent work [38], the experimental apparatus is a "powder cell" analogous to a virtual cylindrical Couette geometry (see Figure 5.19) with the following dimensions:

$$D_i = 15\ mm, \quad D_e = 35\ mm, \quad L_z = 50\ mm, \quad d_g = 10\ mm.$$

However, in our numerical computations, the rescaled geometry with dimensions of $D_i = 1.5\ m$, $D_e = 3.5\ m$, $L_z = 1\ m$ and $d_g = 1\ m$ is used in order to avoid the oscillatory convergence behavior.

In the first step, a grid convergence study is established for Case 1 on a series of refined meshes with square cells of size h whose main properties are listed in Table 5.7. The uniform mesh refinement is applied in x and y directions. Figures 5.31 to 5.33 display respectively the evolution of azimuthal velocity, shear-rate and viscosity along the annular gap for meshes M1–M5. In all figures, the first two meshes show the discrepancies most notably for the shear-rate and viscosity as they are more difficult to compute. The difference is more pronounced as we approach toward the inner rotating cylinder. From the medium mesh M3, the difference of profiles compared to the finest mesh M5 is almost negligible and the results are independent of the grid size as the number of grid points between the annular gap increases.

[1] Particle Image Velocimetry

Table 5.7: Main properties of the grids used for Case 1.

Mesh	$N_x \times N_y$	Cell Types			h/d_g	$\Delta t\, U_{ref}/d_g$
		Cartesian	Cut-cell	Solid		
M1	100×100	6092	568	3340	3.5×10^{-01}	2×10^{-03}
M2	200×200	24900	1136	13964	1.75×10^{-01}	1×10^{-03}
M3	400×400	100840	2280	56880	8.8×10^{-02}	5×10^{-04}
M4	600×600	227068	3416	129516	5.9×10^{-02}	3.3×10^{-04}
M5	800×800	405724	4552	229724	4.4×10^{-02}	2.5×10^{-04}

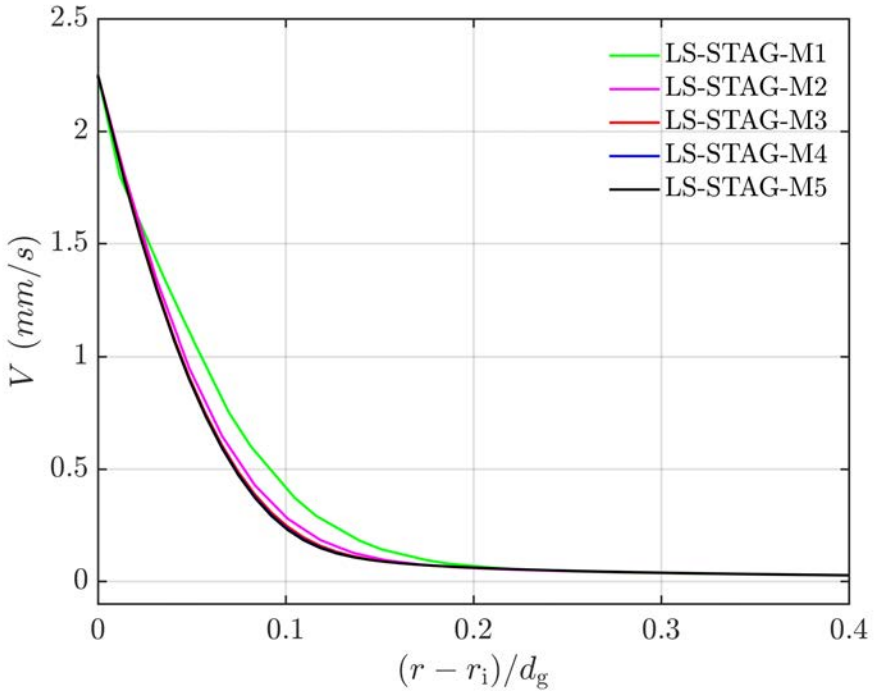


Figure 5.31: A close-up view of the azimuthal velocity profiles along the annular gap for different meshes in Case 1.

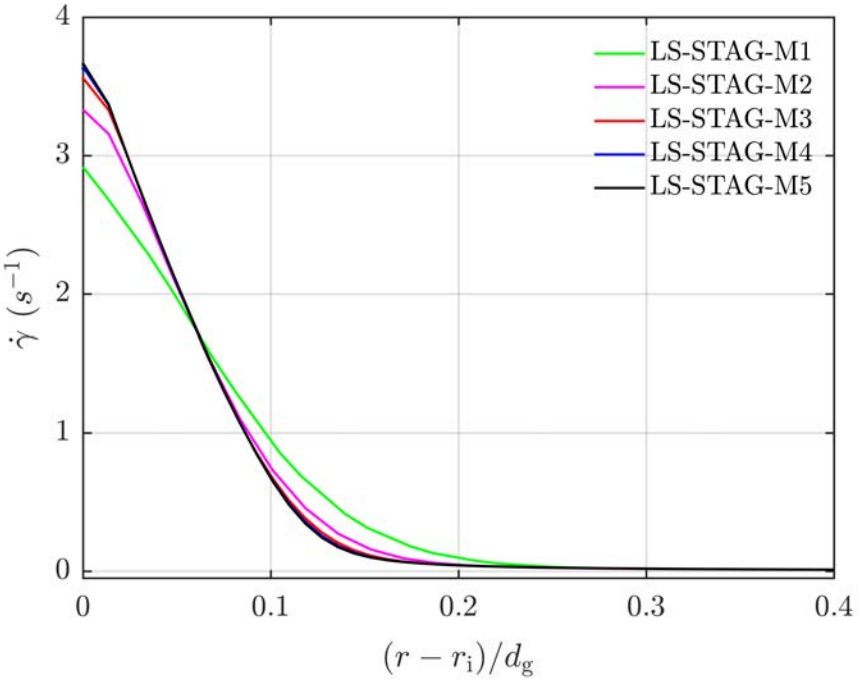


Figure 5.32: A close-up view of the shear-rate profiles along the annular gap for different meshes in Case 1.

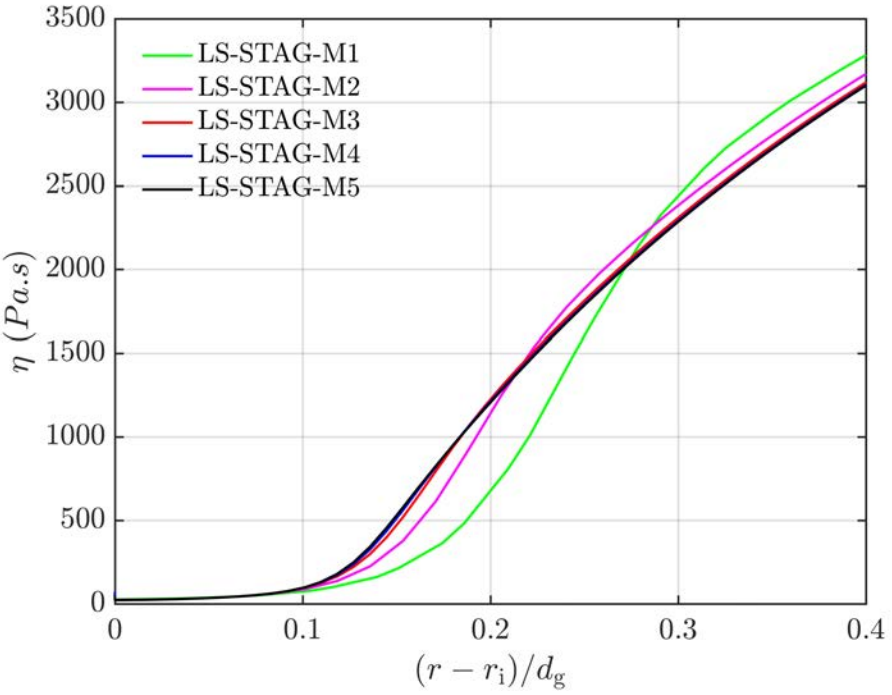


Figure 5.33: A close-up view of the viscosity profiles along the annular gap for different meshes in Case 1.

Another parameter indicative of grid convergence is the value of torque exerted on the inner cylinder. For Case 1, the torque of $M = 1390\,\mu\,N.mm$ is imposed to the granular suspensions by the rheometer [38] which is set as our reference torque value for this test-case. Table 5.8 shows the torque values obtained by LS-STAG for different meshes and their relatives errors compared to the reference value. Although the torque values for the first two meshes are incidentally well predicted, the discrepancies observed in velocity, shear-rate and viscosity profiles (Figures 5.31 to 5.33) reveal that they are not at all capable to properly capture the flow details. For mesh M3, these profiles are well estimated, but relatively high percentage of error in the values of torque hesitates us to choose M3 as the reference grid. By considering the computational time, mesh M4 is used for the remaining computations carried out in this section.

Table 5.8: The torque values exerted on the inner cylinder for different meshes in Case 1.

Cases	$M\ (\mu\,N.mm)$	Error %
LS-STAG, M1	1370.9	1.4
LS-STAG, M2	1426.5	2.6
LS-STAG, M3	1312.0	5.6
LS-STAG, M4	1369.6	1.5
LS-STAG, M5	1409.9	1.4
Experiment [38]	1390	-

Figures 5.34 to 5.36 display the evolution of azimuthal velocity along the annular gap for the cases listed in Table 5.6. The PIV measurements of Gaudel *et al.* [38] and the corresponding velocity profile for a Newtonian fluid with the dynamic viscosity of $\eta = \eta_0$ (see Table 5.6) are also given for comparison. Note that the experimental profiles next to the inner cylinder ($7.5\ mm \geq r \geq 8.5\ mm$) can not be extracted due to the visualization restrictions, refer to [38]. There is a very good agreement between the LS-STAG profiles and the experimental data for all cases. Similar behavior is also observed in the values of torque on inner cylinder indicated in Table 5.9, where the maximal error is about 7% for case 3 with very low Reynolds number.

Table 5.9: The torque values exerted on the inner cylinder for different cases.

Case	$M\ (\mu\,N.mm)$		Error %
	LS-STAG	Gaudel *et al.* [38]	
Case 1	1369.6	1390	1.5
Case 2	1298.6	1390	6.6
Case 3	1129.9	1055	7.1

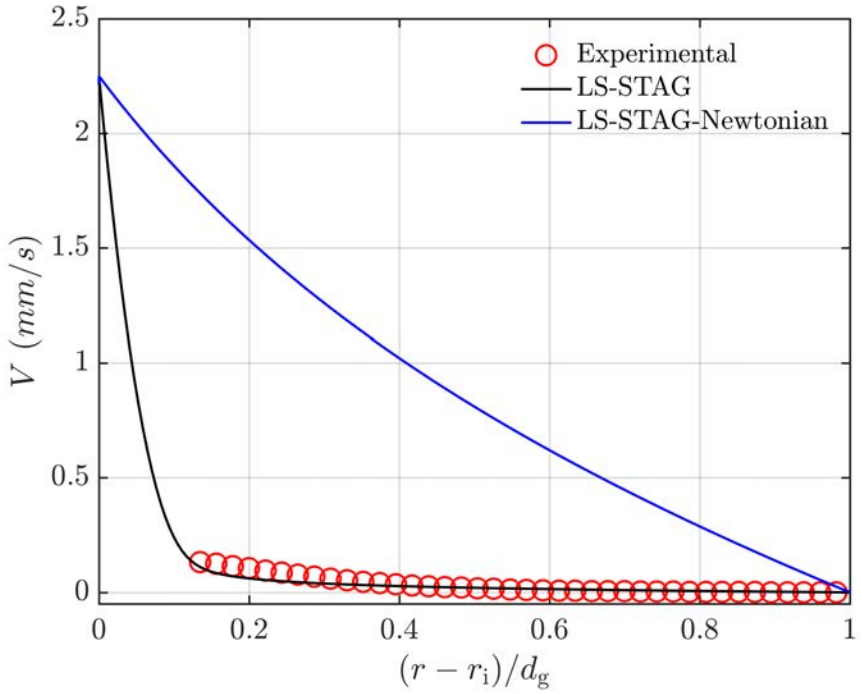


Figure 5.34: The azimuthal velocity profile along the gap for Case 1 along with the experimental data reported by [38] and the velocity profile for corresponding Newtonian fluid.

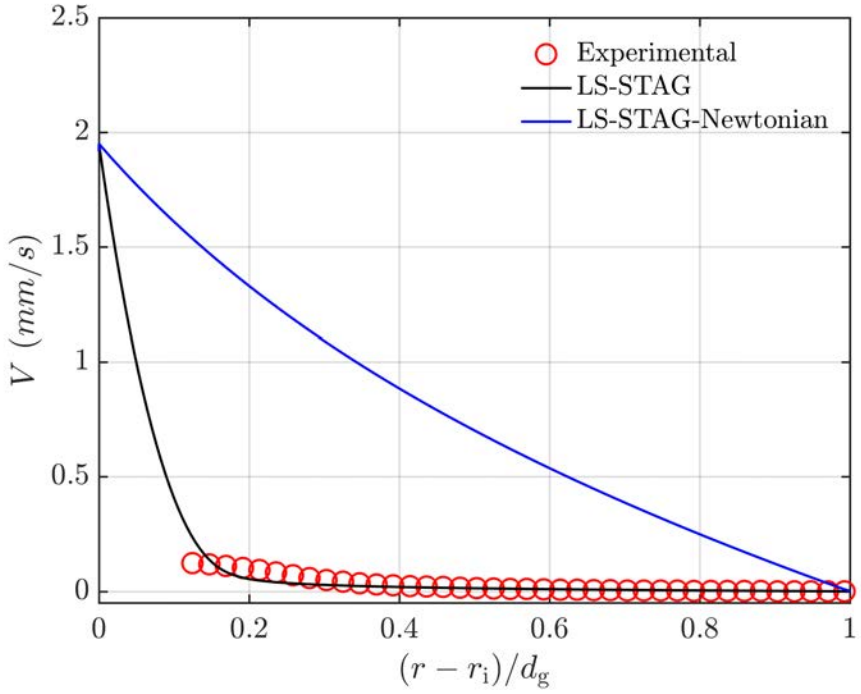


Figure 5.35: The azimuthal velocity profile along the gap for Case 2 along with the experimental data reported by [38] and the velocity profile for corresponding Newtonian fluid.

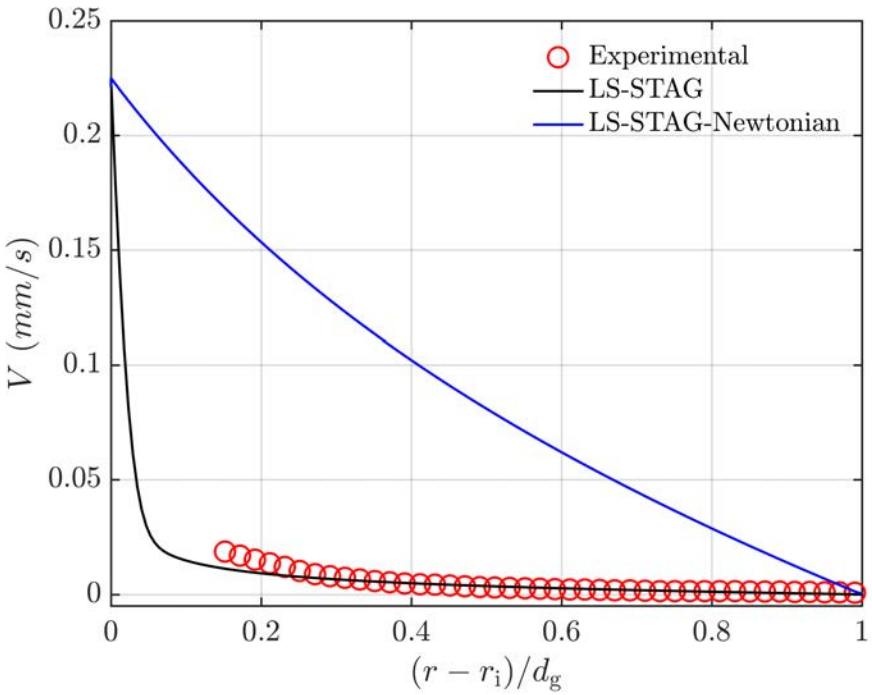


Figure 5.36: The azimuthal velocity profile along the gap in Case 3 along with the experimental data reported by [38] and the velocity profile for corresponding Newtonian fluid.

In granular flows in which the maximal shear-rate is lower than the critical shear-rate $\dot{\gamma}_c$, the viscosity is substantially constant and is approximately equal to the plateau viscosity η_0. Thus, the velocity profiles are close to the Newtonian profiles. In contrast, when the maximum shear-rate is greater than the critical shear-rate, the viscosity varies substantially in the annular gap, so we will see the deviation of the velocity profile from the Newtonian counterpart. According to Figure 5.37 which displays the variation of shear-rate against the annular gap, it can be clearly observed that the maximal shear-rates on the inner cylinder are significantly greater than the corresponding critical shear-rates for all cases. Thus, as expected, the local viscosity varies considerably along the annular gap, see Figure 5.38. Consequently, the velocity profiles deviate from their Newtonian counterparts for all cases indicated in Figures 5.34 to 5.36.

Another interesting quantity in these flows is the local shear stress which is determined such that:

$$\sigma(r) = \dot{\gamma}(r)\,\eta(r).$$

It can be classically proved that the shear stress is inversely proportional to the square of the radius (*i.e.* $\sigma \propto 1/r^2$) for the large gap configuration of the Taylor-Couette cell, see *e.g.* Marchal *et al.* [64]. This behavior is also observed for all test-cases studied in this section, see for instance the evolution of σ against the annular gap in Figure 5.39.

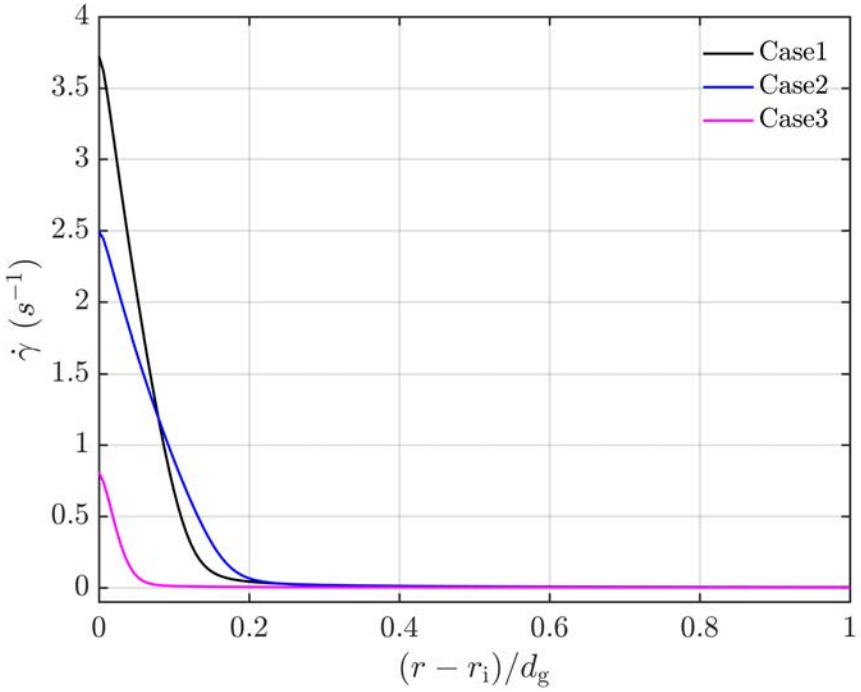


Figure 5.37: The evolution of local shear-rate along the annular gap for different cases.

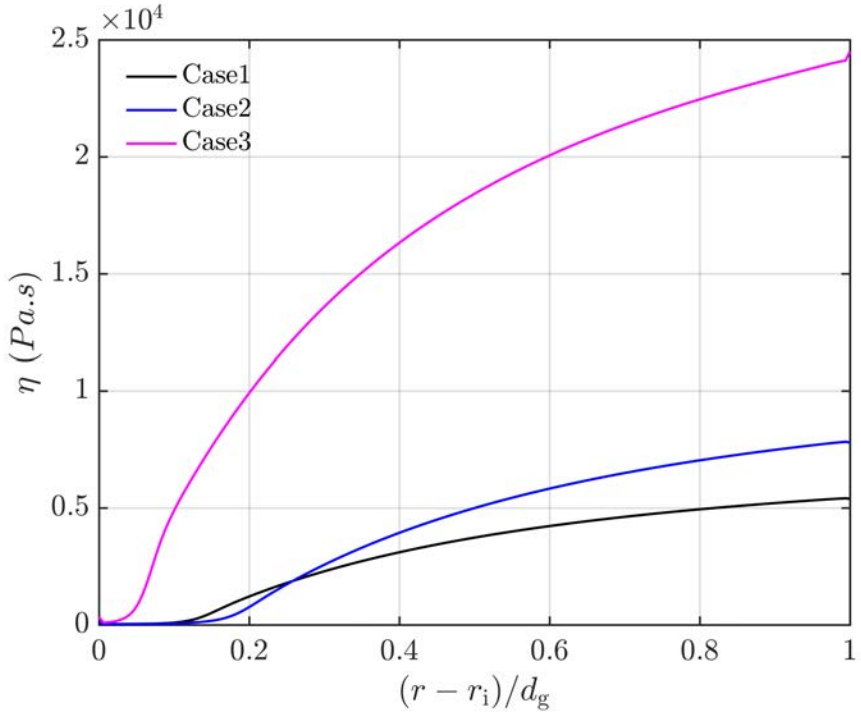


Figure 5.38: The evolution of local viscosity along the annular gap for different cases.

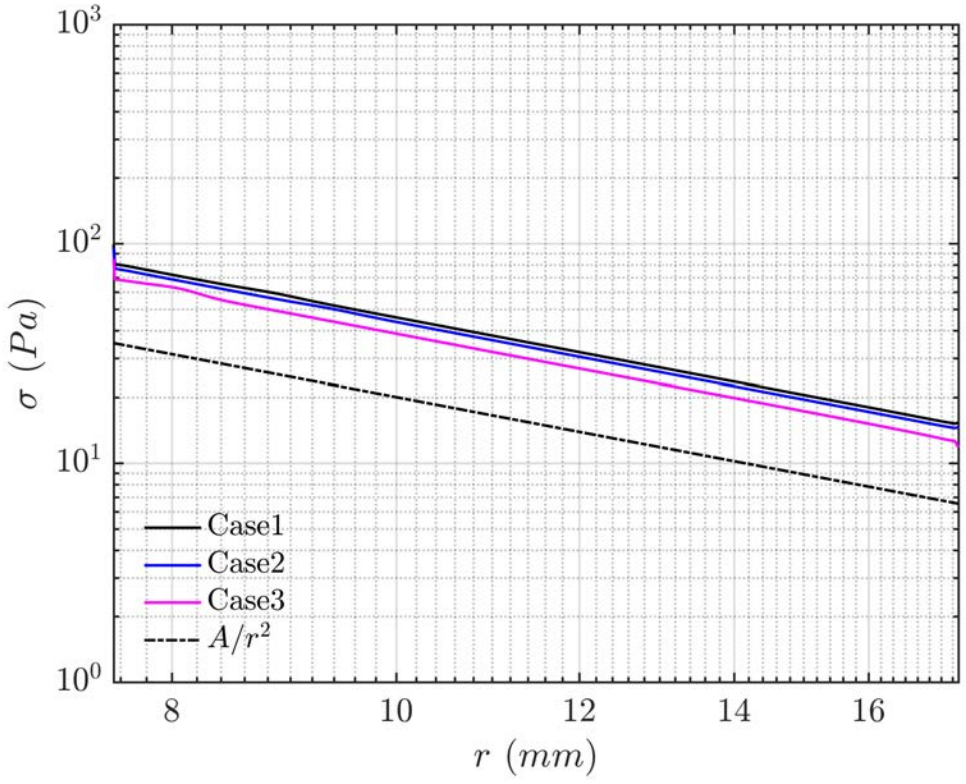


Figure 5.39: The shear stress profile along the annular gap ($r \in [7.5\,17.5]\,mm$) for different cases.

Conclusions and Perspective

The major effort has been made in this thesis is to extend the LS-STAG cut-cell method to 3D extruded configurations in a parallel computing environment. The main accomplishments of this work are summarized in the conclusion part. Some directions and propositions for future work are also given in the perspective part.

Conclusion

– As in 2D version, the discretization of the flow equations for 3D extruded geometries has been performed such that the global invariants of the flow such as total mass, momentum and kinetic energy budget, are strictly preserved in the whole fluid domain. These properties have been successfully devised up to the cut-cells through a precise attention on the terms acting on the immersed boundary in the global conservation equations, both at the continuous and discrete levels.

– In the classical cut-cell methods, the governing equations are discretized in the cut-cells based on ad hoc cell-merging techniques, results in newly formed cell whereby the simplicity of 7-points stencil structure of Cartesian methods is lost. For the 3D extruded version of the LS-STAG method, the various terms have been discretized such that 7-point stencil structure is maintained, leading to significant gain in computer resources (wall time, memory usage).

– The parallelization of LS-STAG solver is achieved by performing a Cartesian decomposition of the domain such as any standard Cartesian method, which shows that the LS-STAG method introduces no specific issue related to parallelism. The parallel scalability of LS-STAG solver is also measured based on the weak and strong scaling for the simulation of 3D periodic and transitional Taylor-Green vortex flow (Section 4.1) for Newtonian and non-Newtonian cases. For both measurements, the typical behavior of the Cartesian-based codes is observed for the LS-STAG solver where its parallel performance deteriorates as the number of processors increases. This behavior is observed for both Newtonian and non-Newtonian cases, however for non-Newtonian fluids, the solver is more efficient than in the Newtonian case.

– The formal order of accuracy of 3D version of LS-STAG solver has been investigated by conducting a grid convergence study on the Hagen-Poiseuille flow for power-law liquids (Section 4.2). Second-order accuracy is obtained in Cartesian cells away from the immersed boundaries. However, the order of accuracy is degraded from super-linear (reported for 2D cases in [22]) to linear only on the whole domain. It is certainly caused by the ad hoc discretization of the off-diagonal elements in velocity gradient tensor, *i.e.* $\partial w/\partial x$ and $\partial w/\partial y$ in which the non-orthogonality contribution (cross diffusion) is not considered. This behavior is also observed in the cut-cell method used by Van der Heiden [99] for similar test-cases.

– A series of benchmark laminar and turbulent flows has been conducted for assessing the performance of the LS-STAG solver in terms of accuracy and computer resources with respect to well-established flow solvers (ANSYS-FLUENT© on body-fitted grids, immersed boundary method). To our knowledge, these issues have seldomly been addressed in the literature to such an extent. The computations of the confined cylinder benchmark (Section 4.3.1) showed that a cut-cell solver was able to provide solution nearly as accurate as a commercial body-fitted CFD package, at a fraction of the computational cost. A close inspection on the accuracy of the LS-STAG method shows that the convergence rate is not as smooth as for methods on body-fitted grids, especially at low Reynolds numbers, which is also associated with the non-orthogonality issue in discretization of viscous terms $\partial w/\partial x$ and $\partial w/\partial y$. Furthermore, the DNS of turbulent cylinder flow (Section 4.3.2) highlighted the superior accuracy of the LS-STAG treatment of complex geometries compared to more traditional immersed boundary methods, for example in [7]. Indeed, since the viscous terms are dominant at low Reynolds numbers, thus the accuracy is degraded due to the ad hoc computation of $\partial w/\partial x$ and $\partial w/\partial y$. As the Reynolds number increases, the pressure terms become predominant and LS-STAG provides results with comparable accuracy to the reference values.

– The flows of shear-thinning fluids (Xanthan solutions with different concentration) in an axisymmetric sudden expansion geometry (Section 5.1) and the flows of vibrated granular materials (dry and suspensions) in a Taylor-Couette cell (Section 5.2) have been also simulated based on available rheological characterization and experimental flow measurements, performed in LEMTA [86, 38], to evaluate the suitability of LS-STAG method for computing non-Newtonian flows. For both test-cases, the computational prediction of flow details is in very good agreement with the experimental data.

Perspective

In this report, the versatility of LS-STAG method has been demonstrated in different applications and subsequently several possible improvements for this method are proposed as follows.

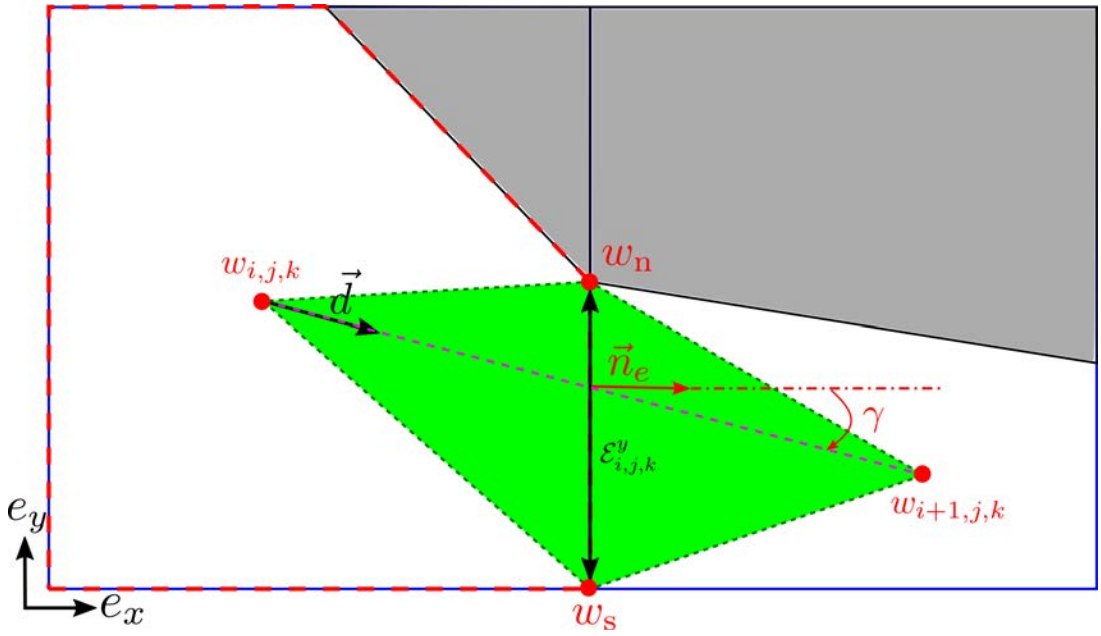


Figure 1: The xy view of two adjacent cut-cells. The velocities involved in diamond cell scheme are $w_{i,j,k}$, $w_{i+1,j,k}$, w_{n} and w_{s}. The vector joining two consecutive velocities $w_{i,j,k}$ and $w_{i+1,j,k}$ is indicated by $\boldsymbol{d}$.

Towards a fully 3D version of LS-STAG method As mentioned, the 3D extruded extension is an important step towards the fully 3D implementation, whereby we can address both discretization and implementation issues on distributed memory machines which may occur in full 3D. The next step is to extend the LS-STAG solver to fully 3D configurations. In these respects, the non-orthogonality issue in discretization of $\partial w/\partial x$ and $\partial w/\partial y$, can be circumvented by applying a finite volume formulation, called the *diamond cell* scheme, used for unstructured grid methods, *e.g.* [11, 24, 25, 65]. For instance in Figure 1, for computing $\partial w/\partial x$ in the pentagonal cut-cell $\Omega_{i,j,k}$, it is necessary to compute the whole velocity gradient, which reads in the local basis of the eastern face $\Gamma^{\mathrm{e}}_{i,j,k}$:

$$\nabla w|_{\mathrm{e}} = \left.\frac{\partial w}{\partial x}\right|_{\mathrm{e}} \boldsymbol{e}_x + \left.\frac{\partial w}{\partial y}\right|_{\mathrm{e}} \boldsymbol{e}_y, \tag{1}$$

since $\boldsymbol{n}_{\mathrm{e}} = \boldsymbol{e}_x$. An elegant way to proceed is to approximate the gradient in the $\boldsymbol{d}$ direction as:

$$\nabla w|_{\mathrm{e}} \cdot \frac{\boldsymbol{d}}{\|\boldsymbol{d}\|} = \cos(\gamma) \left.\frac{\partial w}{\partial x}\right|_{\mathrm{e}} + \sin(\gamma) \left.\frac{\partial w}{\partial y}\right|_{\mathrm{e}} \cong \frac{w_{i+1,j,k} - w_{i,j,k}}{\|\boldsymbol{d}\|}, \tag{2}$$

while the gradient tangential to the face is computed as:

$$\left.\frac{\partial w}{\partial y}\right|_{\mathrm{e}} \cong \frac{w_{\mathrm{n}} - w_{\mathrm{s}}}{\mathcal{E}^y_{i,j,k}}, \tag{3}$$

in which w_{n} and w_{s} are located at the fluid face endpoints $\boldsymbol{x}_{\mathrm{n}}$ and $\boldsymbol{x}_{\mathrm{s}}$. Then, substitution

of (3) in (2) gives the following approximation of the face-normal gradient:

$$\left.\frac{\partial w}{\partial x}\right|_{\mathrm{e}} \cong \frac{w_{i+1,j,k} - w_{i,j,k}}{\|\boldsymbol{d}\| \cos(\gamma)} - \frac{w_{\mathrm{n}} - w_{\mathrm{s}}}{\mathcal{E}^{y}_{i,j,k}} \tan(\gamma), \tag{4a}$$

with

$$\|\boldsymbol{d}\| \cos(\gamma) = x^{\mathrm{c}}_{i+1,j,k} - x^{\mathrm{c}}_{i,j,k}, \quad \tan(\gamma) = \frac{y^{\mathrm{c}}_{i+1,j,k} - y^{\mathrm{c}}_{i,j,k}}{x^{\mathrm{c}}_{i+1,j,k} - x^{\mathrm{c}}_{i,j,k}}. \tag{4b}$$

The first quotient in the RHS of Equation (4a) is the orthogonal contribution of the gradient discretization, while the second quotient which is called the cross-diffusion or non-orthogonal diffusion [103], expresses the non-orthogonal contribution that vanishes when $\boldsymbol{d}$ and $\boldsymbol{n}_{\mathrm{e}}$ are collinear ($\gamma = 0$). Note that for the case indicated in Figure 1, $w_{\mathrm{n}} = w(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k)$ and the value of w_{s} need to be reconstructed from the surrounding w velocities such that:

$$w_{\mathrm{s}} = c_{i,j,k}\, w_{i,j,k} + c_{i+1,j,k}\, w_{i+1,j,k} + c_{i,j-1,k}\, w_{i,j-1,k} + c_{i+1,j-1,k}\, w_{i+1,j-1,k}.$$

Various possibilities are available to determine the reconstruction weights $c_{i,j,k}$ such as volume weighted, inverse distance, least-square, non-linear interpolant, DDFV [49]. This approach which is recently examined for the flow past a confined cylinder (Section 4.3.1), exhibits a considerable improvement in accuracy with a smooth convergence for all parameters discussed in Section 4.3.1.

Development of LS-STAG for multi-physics applications Up to now, the accuracy and robustness of LS-STAG solver has been evaluated for various fluid flow regimes such as laminar and weakly turbulent flows, Newtonian and shear-thinning flows and also vibrated granular flows. However, it will be useful to implement other physical modeling capabilities needed to model flows such as turbulence (using RANS or LES), free surface and multiphase systems in order to acquire a powerful open-source software. It is noteworthy to mention that the heat transfer equations have been recently implemented in the LS-STAG solver and the modeling of multiphase flows are still in progress. In addition, flows with moving boundaries (as done in 2D) can be also taken into account in 3D version.

Implementation of other non-Newtonian behaviors In this study, we have considered only non-Newtonian fluids with shear-thinning behavior. However, in reality the fluids widely found in many industrial applications exhibit various non-Newtonian behaviors such as viscoelastic (well-established in 2D [17]) and time-dependent behaviors. Thus, these behaviors can be also investigated at the future stages of LS-STAG development.

Appendix A

Discretization of the convective fluxes in basic types of cut-cells

In this appendix, we will provide the discretization of the convective fluxes $\int_{\Gamma^u_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n})\,u\ \mathrm{d}S$ for the half control volumes inside three basic types of cut-cells indicated in Figure A.1 which is straightforward extension of 2D version explained in [22] (Appendix A). These fluxes are discretized such that any combination of these half control volumes verifies the skew-symmetry condition (2.28). For an arbitrary control volume $\Omega^u_{i,j,k}$, the convective terms are described as the net fluxes through its elementary faces, *i.e.* $\Gamma^{u,\mathrm{w}}_{i,j,k} \cup \Gamma^{u,\mathrm{s}}_{i,j,k} \cup \Gamma^{u,\mathrm{n}}_{i,j,k} \cup \Gamma^{u,\mathrm{b}}_{i,j,k} \cup \Gamma^{u,\mathrm{t}}_{i,j,k} \cup \Gamma^{u,\mathrm{ib}}_{i,j,k}$, see Equation (3.21). The skew-symmetric discretization of the convective fluxes through these faces will be subsequently determined for different types of cut-cells.

A.1 Discretization of convective fluxes in a triangular cut-cell

In the case of the triangular cut-cell of Figure A.1(a), the discretization is given by:

$$\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u\ \ \mathrm{d}S \cong -\frac{1}{2}\,\overline{u}_{i,j,k}\,\frac{u_{i,j,k}+u^{\mathrm{ib}}_{i,j,k}}{2}, \tag{A.1a}$$

$$\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u\ \ \mathrm{d}S = 0, \tag{A.1b}$$

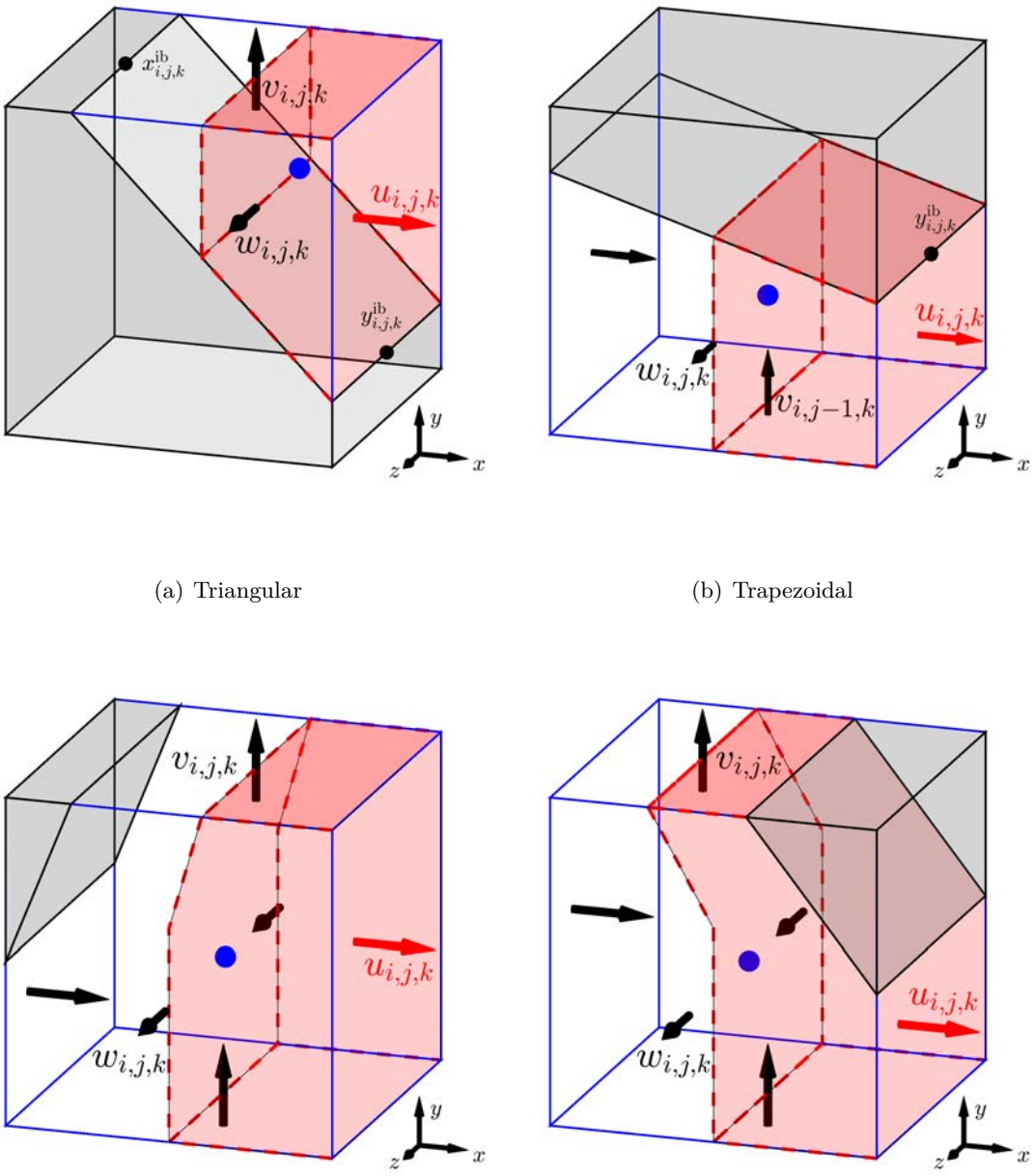
$$\int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u\ \ \mathrm{d}S \cong \frac{1}{2}\,\overline{v}_{i,j,k}\,\frac{u_{i,j,k}+u_{i,j+1,k}}{2}, \tag{A.1c}$$

$$\int_{\Gamma^{u,\mathrm{b}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u\ \ \mathrm{d}S \cong -\frac{1}{2}\,\overline{w}_{i,j,k-1}\,\frac{u_{i,j,k}+u_{i,j,k-1}}{2}, \tag{A.1d}$$

$$\int_{\Gamma^{u,\mathrm{t}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_z)\,u\ \ \mathrm{d}S \cong \frac{1}{2}\,\overline{w}_{i,j,k}\,\frac{u_{i,j,k}+u_{i,j,k+1}}{2}, \tag{A.1e}$$

$$\int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n}^{\mathrm{ib}})\,u\ \ \mathrm{d}S \cong \frac{1}{2}\,\overline{U}^{\mathrm{ib}}_{i,j,k}\,\frac{u_{i,j,k}+u^{\mathrm{ib}}_{i,j,k}}{2}, \tag{A.1f}$$

where $u^{\mathrm{ib}}_{i,j,k} = \frac{1}{2}\,u(x^{\mathrm{ib}}_{i,j,k}, y_j, z_k) + \frac{1}{2}\,u(x_i, y^{\mathrm{ib}}_{i,j,k}, z_k)$.



(a) Triangular

(b) Trapezoidal

(c) Pentagonal

(d) Pentagonal

Figure A.1: Representation of the half control volumes for $u_{i,j,k}$ inside the generic types of cut-cells by red dashed line (--).

Note that the contribution of the complementary half control volume in adjacent cell shall be added to obtain the coefficients of discrete convective terms.

A.2 Discretization of convective fluxes in a trapezoidal cut-cell

For the case of the trapezoidal cut-cell of Figure A.1(b), the discrete forms of convective fluxes are given by:

$$\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\; \mathrm{d}S \cong -\frac{1}{2}\left(\overline{u}_{i,j,k}+\overline{u}_{i-1,j,k}\right)\frac{u_{i,j,k}+u_{i-1,j,k}}{2}, \tag{A.2a}$$

$$\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S \cong -\frac{1}{2}\,\overline{v}_{i,j-1,k}\,\frac{u_{i,j,k}+u_{i,j-1,k}}{2}, \tag{A.2b}$$

$$\int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S = 0, \tag{A.2c}$$

$$\int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n}^{\mathrm{ib}})\,u \;\; \mathrm{d}S \cong \frac{1}{2}\,\overline{U}^{\mathrm{ib}}_{i,j,k}\,\frac{u_{i,j,k}+u(x_i,y^{\mathrm{ib}}_{i,j,k},z_k)}{2}. \tag{A.2d}$$

The contribution of top and bottom faces are determined such like Equations (A.1d) and (A.1e).

A.3 Discretization of convective fluxes in a pentagonal cut-cell

For the pentagonal cut-cells, two distinct cases are investigated. In the case of the pentagonal cut-cell of Figure A.1(c), the discretization is:

$$\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\; \mathrm{d}S \cong -\frac{1}{2}\left(\overline{u}_{i,j,k}+\overline{u}_{i-1,j,k}\right)\frac{u_{i,j,k}+u_{i-1,j,k}}{2}, \tag{A.3a}$$

$$\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S \cong -\frac{1}{2}\,\overline{v}_{i,j-1,k}\,\frac{u_{i,j,k}+u_{i,j-1,k}}{2}, \tag{A.3b}$$

$$\int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S \cong \frac{1}{2}\,\overline{v}_{i,j,k}\,\frac{u_{i,j,k}+u_{i,j+1,k}}{2}, \tag{A.3c}$$

$$\int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n}^{\mathrm{ib}})\,u \;\; \mathrm{d}S \cong \frac{1}{2}\,\overline{U}^{\mathrm{ib}}_{i,j,k}\,\frac{u_{i,j,k}+u_{i-1,j,k}}{2}, \tag{A.3d}$$

and in the case of the pentagonal cut-cell of Figure A.1(d), the fluxes are discretized as:

$$\int_{\Gamma^{u,\mathrm{w}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_x)\,u \;\; \mathrm{d}S \cong -\frac{1}{2}\left(\overline{u}_{i,j,k}+\overline{u}_{i-1,j,k}\right)\frac{u_{i,j,k}+u_{i-1,j,k}}{2}, \tag{A.4a}$$

$$\int_{\Gamma^{u,\mathrm{s}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S \cong -\frac{1}{2}\,\overline{v}_{i,j-1,k}\,\frac{u_{i,j,k}+u_{i,j-1,k}}{2}, \tag{A.4b}$$

$$\int_{\Gamma^{u,\mathrm{n}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{e}_y)\,u \;\; \mathrm{d}S \cong \frac{1}{2}\,\overline{v}_{i,j,k}\,\frac{u_{i,j,k}+u(x_i,y^{\mathrm{ib}}_{i,j,k},z_k)}{2}, \tag{A.4c}$$

$$\int_{\Gamma^{u,\mathrm{ib}}_{i,j,k}} (\boldsymbol{v}\cdot\boldsymbol{n}^{\mathrm{ib}})\,u \;\; \mathrm{d}S \cong \overline{U}^{\mathrm{ib}}_{i,j,k}\,\frac{u_{i,j,k}+u(x_i,y^{\mathrm{ib}}_{i,j,k},z_k)}{2} - \frac{1}{2}\,\overline{U}^{\mathrm{ib}}_{i,j,k}\,\frac{u_{i,j,k}+u_{i-1,j,k}}{2}. \tag{A.4d}$$

Analogously, the contribution of top and bottom faces for both cases are determined such like Equations (A.1d) and (A.1e).

Bibliography

[1] Ivar Aavatsmark, T. Barkve, Ø Bøe, and T. Mannseth. Discretization on non-orthogonal, quadrilateral grids for inhomogeneous, anisotropic media. *Journal of Computational Physics*, 127(1):2–14, 1996.

[2] Philippe Angot, Charles-Henri Bruneau, and Pierre Fabrie. A penalization method to take into account obstacles in incompressible viscous flows. *Numerische Mathematik*, 81(4):497–520, 1999.

[3] Akio Arakawa. Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. part I. *Journal of Computational Physics*, 1(1):119–143, 1966.

[4] Uri M. Ascher and Linda R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, 1998.

[5] D. Badekas and D. D. Knight. Eddy correlations for laminar axisymmetric sudden expansion flows. *Journal of Fluids Engineering*, 114(1):119–121, 1992.

[6] Allison H. Baker, Robert D. Falgout, Tzanio V. Kolev, and Ulrike Meier Yang. Scaling Hypre's multigrid solvers to 100,000 cores. In *High-Performance Scientific Computing*, pages 261–279. Springer, 2012.

[7] Elias Balaras. Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations. *Computers & Fluids*, 33(3):375–404, 2004.

[8] Howard A. Barnes, John Fletcher Hutton, and Kenneth Walters. *An Introduction to Rheology*. Elsevier, 1989.

[9] E. Bayraktar. Private communication, 2016.

[10] Evren Bayraktar, Otto Mierka, and Stefan Turek. Benchmark computations of 3D laminar flow around a cylinder with CFX, OpenFOAM and FeatFlow. *International Journal of Computational Science and Engineering*, 7(3):253–266, 2012.

[11] Enrico Bertolazzi and Gianmarco Manzini. On vertex reconstructions for cell-centered finite volume approximations of 2D anisotropic diffusion problems. *Mathematical Models and Methods in Applied Sciences*, 17(01):1–32, 2007.

[12] Richard P. Beyer and Randall J. LeVeque. Analysis of a one-dimensional model for the immersed boundary method. *SIAM Journal on Numerical Analysis*, 29(2):332–364, 1992.

[13] Robert Byron Bird, Robert Calvin Armstrong, Ole Hassager, and Charles F. Curtiss. *Dynamics of Polymeric Liquids (Volume I)*. Wiley New York, 1977.

[14] I. Bosnyakov, S. Mikhaylov, V. Podaruev, A. Troshin, V. Vlasenko, and A. Wolkov. Application of high-order discontinuous Galerkin method to LES/DES test cases using computers with high number of cores. In *23rd AIAA Computational Fluid Dynamics Conference*, pages 3943–3955, 2017.

[15] Olivier Botella, Mazigh Ait-Messaoud, Adrien Pertat, Yoann Cheny, and Claire Rigal. The LS-STAG immersed boundary method for non-newtonian flows in irregular geometries: flow of shear-thinning liquids between eccentric rotating cylinders. *Theoretical and Computational Fluid Dynamics*, 29(1-2):93–110, 2015.

[16] Olivier Botella and Yoann Cheny. The LS-STAG method for viscous incompressible flows in irregular geometries: Basics of the discretization and application to viscoelastic flows. In *ASME 2010 3rd Joint US-European Fluids Engineering Summer Meeting collocated with 8th International Conference on Nanochannels, Microchannels, and Minichannels*, pages 2441–2451. American Society of Mechanical Engineers, 2010.

[17] Olivier Botella, Yoann Cheny, Farhad Nikfarjam, and Marcela Stoica. Application of the LS-STAG immersed boundary/cut-cell method to viscoelastic flow computations. *Communications in Computational Physics*, 20(4):870–901, 2016.

[18] Olivier Botella, Farhad Nikfarjam, Marcela Stoica, and Yoann Cheny. Entry flow computations of shear-thinning and viscoelastic liquids with the LS-STAG immersed boundary method. In *Eighth International Conference on Computational Fluid Dynamics (ICCFD8)*, 2014.

[19] Malte Braack and Thomas Richter. Solutions of 3D Navier–Stokes benchmark problems with adaptive finite elements. *Computers & Fluids*, 35(4):372–392, 2006.

[20] Yoann Cheny. *La méthode LS-STAG: une nouvelle approche de type frontière immergée/level-set pour la simulation d'écoulements visqueux incompressible en géométries complexes. Application aux fluides Newtonians et viscoélastiques.* PhD thesis, Nancy 1, 2009.

[21] Yoann Cheny and Olivier Botella. An immersed boundary/level-set method for incompressible viscous flows in complex geometries with good conservation properties. *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique*, 18(7-8):561–587, 2009.

[22] Yoann Cheny and Olivier Botella. The LS-STAG method: A new immersed boundary/level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties. *Journal of Computational Physics*, 229(4):1043–1076, 2010.

[23] Donald Keith Clarke, H. A. Hassan, and M. D. Salas. Euler calculations for multielement airfoils using Cartesian grids. *AIAA Journal*, 24(3):353–358, 1986.

[24] William John Coirier. *An adaptively-refined, Cartesian cell-based scheme for the Euler and Navier-Stokes equations*. PhD thesis, University of Michigan, 1994.

[25] Yves Coudière, Jean-Paul Vila, and Philippe Villedieu. Convergence rate of a finite volume scheme for a two dimensional convection-diffusion problem. *ESAIM: Mathematical Modelling and Numerical Analysis*, 33(3):493–516, 1999.

[26] Malcolm M. Cross. Rheology of non-Newtonian fluids: a new flow equation for pseudoplastic systems. *Journal of Colloid Science*, 20(5):417–437, 1965.

[27] Michel Crouzeix and Pierre-Arnaud Raviart. Conforming and nonconforming finite element methods for solving the stationary Stokes equations I. *Revue Française d'Automatique, Informatique, Recherche Opérationnelle. Mathématique*, 7(3):33–75, 1973.

[28] Abhijit P. Deshpande, J. Murali Krishnan, and Sunil Kumar. *Rheology of Complex Fluids*. Springer Science & Business Media, 2010.

[29] Jerome Droniou. Finite volume schemes for diffusion equations: introduction to and review of modern methods. *Mathematical Models and Methods in Applied Sciences*, 24(08):1575–1619, 2014.

[30] R. Eymard, T. Gallouet, and R. Herbin. Finite volume methods. In P. Ciarlet and J. L. Lions, editors, *Handbook for Numerical Analysis*, pages 715–1022, 2000.

[31] Robert Eymard, Thierry Gallouët, and Raphaele Herbin. Discretization of heterogeneous and anisotropic diffusion problems on general nonconforming meshes SUSHI: a scheme using stabilization and hybrid interfaces. *IMA Journal of Numerical Analysis*, 30(4):1009–1043, 2009.

[32] E. A. Fadlun, R. Verzicco, Paola Orlandi, and J. Mohd-Yusof. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *Journal of Computational Physics*, 161(1):35–60, 2000.

[33] Robert D. Falgout, Jim E. Jones, and Ulrike Meier Yang. Conceptual interfaces in Hypre. *Future Generation Computer Systems*, 22(1):239–251, 2006.

[34] Robert D. Falgout and Ulrike Meier Yang. Hypre: A library of high performance preconditioners. In *International Conference on Computational Science*, pages 632–641. Springer, 2002.

[35] Stanley Fedkiw, Ronald Osher, and Stanley Osher. Level set methods and dynamic implicit surfaces. *Surfaces*, 44:77, 2002.

[36] Joel H. Ferziger and Milovan Peric. *Computational Methods for Fluid Dynamics*. Springer Science & Business Media, 2012.

[37] Ansys Fluent. ANSYS Fluent theory guide (release 14.0), 2011.

[38] Naïma Gaudel, Sébastien Kiesgen de Richter, Nicolas Louvet, Mathieu Jenny, and Salaheddine Skali-Lami. Bulk and local rheology in a dense and vibrated granular suspension. *Physical Review E*, 96(6):062905, 2017.

[39] Yong X. Gen. *Continuum Mechanics - Progress in Fundamentals and Engineering Applications*. Rijeka, Croatia: InTech, 2012.

[40] Reza Ghias, Rajat Mittal, and Thomas S. Lund. A non-body conformal grid method for simulation of compressible flows with complex immersed boundaries. *AIAA Paper*, 80(2004):6, 2004.

[41] Frederic Gibou, Ronald P Fedkiw, Li-Tien Cheng, and Myungjoo Kang. A second-order-accurate symmetric discretization of the poisson equation on irregular domains. *Journal of Computational Physics*, 176(1):205–227, 2002.

[42] A. Gilmanov, F. Sotiropoulos, and E. Balaras. A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids. *Journal of Computational Physics*, 191(2):660–669, 2003.

[43] D. Goldstein, R. Handler, and L. Sirovich. Modeling a no-slip flow boundary with an external force field. *Journal of Computational Physics*, 105(2):354–366, 1993.

[44] A. L. Halmos and D. V. Boger. The behavior of a power-law fluid flowing through a sudden expansion. Part II. experimental verification. *AIChE Journal*, 21(3):550–553, 1975.

[45] A. L. Halmos, D. V. Boger, and A. Cabelli. The behavior of a power-law fluid flowing through a sudden expansion: Part I. a numerical solution. *AIChE Journal*, 21(3):540–549, 1975.

[46] Caroline Hanotin. *Rhéophysique des suspensions granulaires vibrées*. PhD thesis, Université de Lorraine, 2014.

[47] Francis H. Harlow and J. Eddie Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids*, 8(12):2182–2189, 1965.

[48] John C. Hart. Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer*, 12(10):527–545, 1996.

[49] F. Hermeline. A finite volume method for the approximation of diffusion operators on distorted meshes. *Journal of Computational Physics*, 160(2):481–499, 2000.

[50] Julian C. R. Hunt, Alan A. Wray, and Parviz Moin. Eddies, streams, and convergence zones in turbulent flows. In *Proceedings of the 1988 Summer Program*, pages 193–208, 1988.

[51] Gianluca Iaccarino, Georgi Kalitzin, and Christopher J Elkins. Numerical and experimental investigation of the turbulent flow in a ribbed serpentine passage. Technical report, Stanford University CA Department of Mechanical Engineering, 2003.

[52] Gianluca Iaccarino, Georgi Kalitzin, and Bahram Khalighi. Towards an immersed boundary RANS flow solver. *AIAA Paper*, 770, 2003.

[53] Gianluca Iaccarino and Roberto Verzicco. Immersed boundary technique for turbulent flow simulations. *Applied Mechanics Reviews*, 56(3):331–347, 2003.

[54] Volker John. Higher order finite element methods and multigrid solvers in a benchmark problem for the 3D Navier–Stokes equations. *International Journal for Numerical Methods in Fluids*, 40(6):775–798, 2002.

[55] Volker John. On the efficiency of linearization schemes and coupled multigrid methods in the simulation of a 3D flow around a cylinder. *International Journal for Numerical Methods in Fluids*, 50:845–862, 2006.

[56] S. Kang, G. Iaccarino, and P. Moin. Accurate and efficient immersed-boundary interpolations for viscous flows. *Center for Turbulence Research Briefs*, pages 31–43, 2004.

[57] Khodor Khadra, Philippe Angot, Sacha Parneix, and Jean-Paul Caltagirone. Fictitious domain approach for numerical modelling of Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 34(8):651–684, 2000.

[58] M. P. Kirkpatrick, S. W. Armfield, and J. H. Kent. A representation of curved boundaries for the solution of the Navier-Stokes equations on a staggered three-dimensional Cartesian grid. *Journal of Computational Physics*, 184:1–36, 2003.

[59] Arthur G. Kravchenko, Parviz Moin, and Karim Shariff. B-spline method and zonal grids for simulations of complex turbulent flows. *Journal of Computational Physics*, 151(2):757–789, 1999.

[60] Ming Chih Lai and Charles S. Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *Journal of Computational Physics*, 160(2):705–719, 2000.

[61] Enzo O. Macagno and Tin-Kan Hung. Computational and experimental study of a captive annular eddy. *Journal of Fluid Mechanics*, 28:43–64, 4 1967.

[62] Sekhar Majumdar, Gianluca Iaccarino, and Paul Durbin. RANS solvers with adaptive structured boundary non-conforming grids. *Annual Research Briefs, Center for Turbulence Research, Stanford University*, pages 353–466, 2001.

[63] N. N. Mansour, P. Moin, W. C. Reynolds, and J. H. Ferziger. Improved methods for large-eddy simulations of turbulence. In *Symposium on Turbulent Shear Flows*, volume 1, pages 14–21, 1977.

[64] Ph. Marchal, N. Smirani, and L. Choplin. Rheology of dense-phase vibrated powders and molecular analogies. *Journal of Rheology*, 53(1):1–29, 2009.

[65] S. R. Mathur and J. Y. Murthy. A pressure-based method for unstructured meshes. *Numerical Heat Transfer*, 31(2):195–215, 1997.

[66] R. Mittal and S. Balachandar. On the inclusion of three-dimensional effects in simulations of two-dimensional bluff-body wake flows. In *Proceedings of the ASME Fluids Engineering Division Summer Meeting, Vancouver, BC, Canada*, 1997.

[67] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005.

[68] Rajat Mittal, Veeraraghavan Seshadri, and Holavanahalli S Udaykumar. Flutter, tumble and vortex induced autorotation. *Theoretical and Computational Fluid Dynamics*, 17(3):165–170, 2004.

[69] Yohei Morinishi, Thomas S. Lund, Oleg V. Vasilyev, and Parviz Moin. Fully conservative higher order finite difference schemes for incompressible flow. *Journal of Computational Physics*, 143(1):90–124, 1998.

[70] F. Moukalled, L. Mangani, M. Darwish, et al. *The Finite Volume Method in Computational Fluid Dynamics*. Springer, 2016.

[71] Debabrata Nag and Amitava Datta. Variation of the recirculation length of Newtonian and non-Newtonian power-law fluids in laminar flow through a suddenly expanded axisymmetric geometry. *Journal of Fluids Engineering*, 129(2):245–250, 2007.

[72] Franck Nicoud. Conservative high-order finite-difference schemes for low-mach number flows. *Journal of Computational Physics*, 158(1):71–97, 2000.

[73] C. Norberg. Flow around a circular cylinder: aspects of fluctuating lift. *Journal of Fluids and Structures*, 15(3-4):459–469, 2001.

[74] I. Orlanski. A simple boundary condition for unbounded hyperbolic flows. *Journal of Computational Physics*, **21**, 251 - 269, 1976.

[75] Stanley Osher and Ronald P. Fedkiw. Level set methods: an overview and some recent results. *Journal of Computational Physics*, 169(2):463–502, 2001.

[76] Bockchoon Pak, Young I. Cho, and Stephen U. S. Choi. Separation and reattachment of non-Newtonian fluid flows in a sudden expansion pipe. *Journal of Non-Newtonian Fluid Mechanics*, 37(2):175–199, 1990.

[77] Suhas Patankar. *Numerical Heat Transfer and Fluid Flow*. CRC Press, 1980.

[78] Charles S Peskin. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics*, 10(2):252–271, 1972.

[79] Charles S Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

[80] V Puzikova and I Marchevsky. Application of the LS-STAG immersed boundary method for numerical simulation in coupled aeroelastic problems. In *Proceedings of the 11th World Congress on Computational Mechanics (WCCM XI), E. Oñte, J. Oliver and A. Huerta (Eds)*, pages 20–25, 2014.

[81] V Puzikova and I Marchevsky. Extension of the LS-STAG cut-cell immersed boundary method for RANS-based turbulence models. In *Proceedings of the International Summer School-Conference "Advanced Problems in Mechanics", June*, 2014.

[82] Valeria V Puzikova. On generalization of the LS-STAG immersed boundary method for large eddy simulation and detached eddy simulation. In *Advanced Problems in Mechanics International Summer School-Conference (Institute for Problems in Mechanical Engineering, Saint-Petersburg, 2015)*, pages 411–417, 2015.

[83] B. N. Rajani, A. Kandasamy, and Sekhar Majumdar. Numerical simulation of laminar flow past a circular cylinder. *Applied Mathematical Modelling*, 33:1228–1247, 2009.

[84] M. Anandha Rao. *Rheology of Fluid and Semisolid Foods: Principles and Applications*. Springer Science & Business Media, 2010.

[85] Pierre-Arnaud Raviart and Jean-Marie Thomas. A mixed finite element method for 2-nd order elliptic problems. In *Mathematical aspects of finite element methods*, pages 292–315. Springer, 1977.

[86] Claire Rigal. *Comportement de fluides complexes sous écoulement : Approche expérimentale par résonance magnétique nucléaire et techniques optiques et simulations numeriques*. PhD thesis, Université de Lorraine, 2012.

[87] E. M. Saiki and S. Biringen. Numerical simulation of a cylinder in uniform flow: application of a virtual boundary method. *Journal of Computational Physics*, 123(2):450–465, 1996.

[88] P. Saramito. Numerical simulation of viscoelastic fluid flows using incompressible finite element method and a θ-method. *Mathematical Modelling and Numerical Analysis*, 28(1):1–35, 1994.

[89] Ruben Scardovelli and Stéphane Zaleski. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, 31(1):567–603, 1999.

[90] Michael Schäfer, Stefan Turek, Franz Durst, Egon Krause, and Rolf Rannacher. Benchmark computations of laminar flow around a cylinder. In Ernst Heinrich Hirschel, editor, *Notes on Numerical Fluid Mechanics (NNFM)*, volume 48, chapter Flow Simulation with High-Performance Computers II, pages 547–566. Vieweg+Teubner Verlag, 1996.

[91] Kai Schneider. Immersed boundary methods for numerical simulation of confined fluid and plasma turbulence in complex geometries: a review. *Journal of Plasma Physics*, 81(6), 2015.

[92] Kai Schneider and Marie Farge. Decaying two-dimensional turbulence in a circular container. *Physical review letters*, 95(24):244502, 2005.

[93] P. S. Scott, F. A. Mirza, and J. Vlachopoulos. A finite element analysis of laminar flows through planar and axisymmetric abrupt expansions. *Computers & Fluids*, 14(4):423–432, 1986.

[94] Kenji Takizawa, Takashi Yabe, Yumiko Tsugawa, Tayfun E Tezduyar, and Hiroki Mizoe. Computation of free-surface flows and fluid–object interactions with the cip method based on adaptive meshless soroban grids. *Computational Mechanics*, 40(1):167–183, 2007.

[95] H. S. Udaykumar, Heng-Chuan Kan, Wei Shyy, and Roger Tran-Son-Tay. Multiphase dynamics in arbitrary geometries on fixed Cartesian grids. *Journal of Computational Physics*, 137(2):366–405, 1997.

[96] H. S. Udaykumar, R. Mittal, P. Rampunggoon, and A. Khanna. A sharp interface Cartesian grid method for simulating flows with complex moving boundaries. *Journal of Computational Physics*, 174:345–380, 2001.

[97] H. S. Udaykumar, Rajat Mittal, and Wei Shyy. Computation of solid–liquid phase fronts in the sharp interface limit on fixed grids. *Journal of Computational Physics*, 153(2):535–574, 1999.

[98] Henri J. L. van der Heiden, Arthur E.P Veldman, and Roel Luppes. A Quasi-3-D extension of the LS-STAG cut-cell immersed boundary method. In *Joint EUROMECH/ERCOFTAC Colloquium*, volume 549, 2013.

[99] P. Van der Plas, H. J. L. Van der Heiden, A. E. P. Veldman, R. Luppes, and R. W. C. P. Verstappen. Efficiently simulating viscous flow effects by means of regularization turbulence modeling and local grid refinement. In *Seventh International Conference on Computational Fluid Dynamics (ICCFD7), Hawaii, paper ICCFD-2503*, 2012.

[100] J. Van Kan, Cornelis Vuik, and P. Wesseling. Fast pressure calculation for 2-d and 3-d time dependent incompressible flow. *Numerical Linear Algebra with Applications*, 7(6):429–447, 2000.

[101] O. V. Vasilyev. High order finite difference schemes on non-uniform meshes with good conservation properties. *Journal of Computational Physics*, 157:746–761, 2000.

[102] R. W. C. P. Verstappen and A. E. P. Veldman. Symmetry-preserving discretization of turbulent flow. *Journal of Computational Physics*, 187(1):343–368, 2003.

[103] Zahir U. A. Warsi. *Fluid Dynamics: Theoretical and Computational Approaches*. CRC Press, 2005.

[104] C. Weiselsberger. New data on the law of hydro and aerodynamic resistance. *Physikalische Zeitschrift*, 22:321–8, 1921.

[105] Charles H. K. Williamson. Vortex dynamics in the cylinder wake. *Annual Review of Fluid Mechanics*, 28(1):477–539, 1996.

[106] Tao Ye, Rajat Mittal, HS Udaykumar, and Wei Shyy. An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries. *Journal of Computational Physics*, 156(2):209–240, 1999.

[107] N. Zhang and Zhongquan Charlie Zheng. An improved direct-forcing immersed-boundary method for finite difference applications. *Journal of Computational Physics*, 221(1):250–268, 2007.

Résumé

Extension de la méthode LS-STAG de type frontière immergée/cut-cell aux géométries 3D extrudées. Applications aux écoulements newtoniens et non newtoniens.

La méthode LS-STAG est une méthode de type frontière immergée/*cut-cell* pour le calcul d'écoulements visqueux incompressibles qui est basée sur la méthode MAC pour grilles cartésiennes décalées, où la frontière irrégulière est nettement représentée par sa fonction level-set, résultant en un gain significatif en ressources informatiques par rapport aux codes MFN commerciaux utilisant des maillages qui épousent la géométrie. La version 2D est maintenant bien établie et ce manuscrit présente son extension aux géométries 3D avec une symétrie translationnelle dans la direction z (configurations extrudées 3D). Cette étape intermédiaire sera considérée comme la clé de voûte du solveur 3D complet, puisque les problèmes de discrétisation et d'implémentation sur les machines à mémoire distribuée sont abordés à ce stade de développement. La méthode LS-STAG est ensuite appliquée à divers écoulements newtoniens et non-newtoniens dans des géométries extrudées 3D (conduite axisymétrique, cylindre circulaire, conduite cylindrique avec élargissement brusque, *etc.*) pour lesquels des résultats de références et des données expérimentales sont disponibles. Le but de ces investigations est d'évaluer la précision de la méthode LS-STAG, d'évaluer la polyvalence de la méthode pour les applications d'écoulement dans différents régimes (fluides newtoniens et rhéofluidifiants, écoulement laminaires stationnaires et instationnaires, écoulements granulaires) et de comparer ses performances avec de méthodes numériques bien établies (méthodes non structurées et de frontières immergées).

Mots-clés: Mécanique des Fluides Numérique (MFN), fluides non-newtoniens, méthode frontière immergée, méthode cut-cell, calcul à haute performance.

Abstract

Extension of the LS-STAG immersed boundary/cut-cell method to 3D extruded geometries. Application to Newtonian and non-Newtonian flows.

The LS-STAG method is an immersed boundary/*cut-cell* method for viscous incompressible flows based on the staggered MAC arrangement for Cartesian grids where the irregular boundary is sharply represented by its level-set function. This approach results in a significant gain in computer resources compared to commercial body-fitted CFD codes. The 2D version of LS-STAG method is now well-established and this manuscript presents its extension to 3D geometries with translational symmetry in the z direction (3D extruded configurations). This intermediate step will be regarded as the milestone for the full 3D solver, since both discretization and implementation issues on distributed memory machines are tackled at this stage of development. The LS-STAG method is then applied to Newtonian and non-Newtonian flows in 3D extruded geometries (axisymmetric pipe, circular cylinder, duct with an abrupt expansion, *etc.*) for which benchmark results and experimental data are available. The purpose of these investigations is to evaluate the accuracy of LS-STAG method, to assess the versatility of method for flow applications at various regimes (Newtonian and shear-thinning fluids, steady and unsteady laminar to turbulent flows, granular flows) and to compare its performance with well-established numerical methods (body-fitted and immersed boundary methods).

Keywords: Computational Fluid Dynamics (CFD), non-Newtonian fluids, immersed boundary methods, cut-cell methods, High-Performance Computing (HPC).