

Experimental and Numerical Investigation of Shallow Mixing Layer

Atefeh Fazlollahi

A Thesis

In the Department

of

Building, Civil and Environmental Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy (Civil Engineering) at

Concordia University

Montréal, Québec, Canada

November 2021

©Atefeh Fazlollahi, 2021

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Atefeh Fazlollahi

Entitled: Experimental and numerical investigation of shallow mixing layer

and submitted in partial fulfillment of the requirements for the degree of

Doctor Of Philosophy (Civil Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____Chair
Dr. Pouya Valizadeh

_____External Examiner
Dr. Yee-Chung Jin

_____External to Program
Dr. Lyes Kadem

_____Examiner
Dr. Zhi Chen

_____Examiner
Dr. Attila Michael Zsaki

_____Thesis Supervisor
Dr. Samuel Li

Approved by _____
Dr. Mazdak Nik-Bakht, Graduate Program Director

12/16/2021 _____
Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering and Computer Science

Abstract

Experimental and numerical investigation of shallow mixing layer

Atefeh Fazlollahi Ph.D.

Concordia University, 2022

When two nearly parallel streams of different velocities join together at a river confluence, a mixing layer develops along the joining interface. The mixing layer is termed a shallow mixing layer (SML) when the flow has a small depth and thus is significantly influenced by riverbed friction. SML has impacts on the riverine environment, river ecosystems, and hydraulic engineering design. The effects of the velocity ratio (V_r) between the two incoming streams on SML characteristics have not been addressed adequately from both Lagrangian and Eulerian perspectives. This study employs the Lagrangian and Eulerian approaches to explore the SML. Specifically, this study aims to answer these main questions: How does SML behave under different ratios of incoming flow velocities? How does the shear caused by the velocity gradient influence the fluid particles in SML? To what extent previous relations proposed to determine mixing layer width are practical? What factors control the pairing of adjacent eddies and the growth of large-scale coherent structures? How are the fluid mass and momentum exchanges affected by V_r ?

The dye visualisation and particle tracking velocimetry (PTV) techniques were used in the laboratory experiments of SML. In the dye visualisation technique, a tracer's motion is recorded using a single camera. PTV is considered an optical flow measurement technique in which neutrally buoyant particles are tracked in consecutive image frames. PTV provides the essential means for Lagrangian studies of SML. For numerical simulations of SML, the smoothed particle hydrodynamics (SPH) model was used. SPH is considered a Lagrangian CFD method in which a continuum is discretised using a set of material points or particles. Laboratory experiments of SML were conducted at three velocity ratios: $V_r = 1, 1.14, \text{ and } 1.5$, and PTV measurements were made from the region between the joining location ($x = 0 \text{ m}$) and 0.3 m downstream ($x = 0.3 \text{ m}$) in the confluence. The dye visualisation experiments consisted of the same three velocity ratios, and the visualisation covered the region of $x = 0\text{--}1.2 \text{ m}$. The SPH simulations included three velocity ratios: $V_r = 1.14, 1.5, \text{ and } 3$; the domain covered $x = 0\text{--}1 \text{ m}$.

The PTV measurements show that the boundary layers, which develop on the sidewalls of a splitter plate used to separate the two incoming streams before joining, and the wake effect cause a velocity deficit in the confluence and limit the mixing layer growth. The SPH results reveal that a smaller velocity ratio results in a more visible velocity deficit in streamwise velocity profiles due to the relative importance of the wake versus velocity gradient in the SML. PTV application for SML investigation has been found to require special technical considerations, some of which were introduced in this study. The technical measures in the data acquisition and the Python script developed for particle trajectory analysis improved the PTV technique in studying SML. Finite-time Lyapunov exponents (FTLE) results indicate that for particle trajectories located inside the mixing layer, a divergence was evident with a positive value of FTLE. However, for the particle trajectories out of the mixing layer, both positive and negative FTLE can be observed. The dye

visualisation results show that turbulent instabilities still form in the absence of velocity gradient ($V_r = 1$). When the velocity gradient exists ($V_r > 1$), the instabilities persist, and a pairing of eddies is observed. The intermittency of SML is observed with a lack of a temporally fixed pattern in the vortex arrangement at a fixed location. The dye visualisation results show a linear relation between eddy spacing and downstream distance, with the most frequent eddy spacing being $0.42x$.

A new approach to the determination of mixing layer width was proposed based on the boundary layer definition. Results show that for smaller velocity ratios, the mixing layer width determined by the boundary layer method is smaller than those from existing empirical relations. A pairing process of vortices occurs less often when V_r is as small as 1.14, compared to that for $V_r = 1.5$ and 3. The results also show that pairing activities in SML are affected mainly by the average vorticity magnitude of two neighbouring eddies rather than their relative distance. The Okubo-Weiss parameter of SML indicates that the general form of an eddy in SML consists of an inner vorticity-dominated region at the core and an outer region, which is strain-dominated and surrounding the inner region. The strain-dominated boundary of the eddies performs such a barrier for the particles inside the eddy until the eddies decay or are paired with other eddies. Smaller velocity ratios result in lower mass transfer with a constant rate from the tip of the splitter plate to the downstream whilst for $V_r = 3$, the mass transfer rate increases moving downstream due to the larger eddies and more profound pairing process. In the SML, the intermittent crests and troughs in momentum transfer indicate the evolution of eddies.

Acknowledgments

I would like to thank my supervisor, Dr. S. Samuel Li, for his constant support, help and guidance throughout this research study. I would also like to take this opportunity to express my gratitude to the members of my thesis committee for the reviewing of this thesis and their valuable comments.

I would like to thank Dr. Nahid Shakib for her comments on proofreading my dissertation. I am also grateful to Dr. Susan Gaskin for the guidance in the PTV experiments. I am grateful that McGill University provided the facility for performing the PTV experiments. Although he was not directly contributed to this research study, I would like to thank my master's supervisor, Dr. Hossein Afzalimehr. His point of view through teaching fluid mechanics was so inspiring that it took me to this point!

I will be failing in my duty if I do not mention special thanks to my husband, Ahmad, who stood by my side until the end of this journey. I am also grateful to my family. Although I was very far from them, their love has been such an energy booster for overcoming challenges.

Dedication

Dedicated to my mother

Table of Contents

List of Tables	ix
List of Figures.....	ix
List of Symbols.....	xiv
List of Abbreviation.....	xviii
Chapter 1.....	1
Introduction.....	1
1.1 Introduction and problem definition	1
1.2 Motivation.....	3
1.3 Objectives and thesis layout.....	3
Chapter 2.....	5
General background.....	5
2.1 Introduction.....	5
2.2 Free mixing flow	5
2.3 Shallow mixing flow (SML).....	6
Chapter 3.....	15
Investigation of SML using particle tracking velocimetry and dye visualization	15
3.1 Introduction.....	15
3.2 Experimental setup.....	19
3.3 PTV data acquisition.....	22
3.3.1 Image processing	25
3.3.2 Dataset analysis.....	29
3.3.3 Methods for multivariate outlier detection	30
3.4 Ensemble average flow velocity	38
3.5 Particle trajectory	41
3.6 Limitations of experiments	52
3.7 Dye visualization experiment	54
3.7.1 Mixing mechanisms	54
3.7.2 Characteristics of coherent structures	57
3.8 Conclusion	59
Chapter 4.....	61
Smoothed particle hydrodynamics simulation of shallow mixing layer	61
4.1 Introduction.....	61

4.2	Smoothed particle hydrodynamics theory	63
4.3	Methods in DualSPHysics	65
4.3.1	Boundary treatment.....	67
4.3.2	Shifting algorithm	68
4.3.3	Specifications of model parameters	69
4.4	Results.....	73
4.4.1	Effect of wake	74
4.4.2	Temporal and spatial evolutions of flow structures	76
4.4.3	Exchange of fluid mass and momentum.....	85
4.5	Conclusion	92
	Chapter 5.....	94
	Conclusions and future directions	94
5.1	Summary of research contributions	94
5.2	Conclusions.....	94
5.3	Limitations	96
5.4	Recommendations for future research	96
	References.....	97
	Appendix A1: Extensible Markup Language (XML) file for SPH simulation, Run A.....	102
	Appendix A2: Extensible Markup Language (XML) file for SPH simulation, Run B.....	109
	Appendix A3: Extensible Markup Language (XML) file for SPH simulation, Run C	116
	Appendix B: Bash file for SPH simulation.....	123

List of Tables

Table 3.1. Hydraulic conditions of experiments. The subscript 0 refers to the longitudinal location at the splitter plate tip

Table 3.2. The variance of streamwise (u) and spanwise velocity (v) for raw data (without filtering) and two studied filtering methods

Table 4.1. Hydraulic conditions for SPH simulations of shallow mixing layers. The subscripts 1 and 2 refer to the low and high-velocity sides, respectively

Table 4.2. Comparison of mixing layer width determined using boundary layer definition (this study) and methods based on maximum velocity gradient (previous studies)

Table 4.3. Comparison of mixing layer width determined using PTV experiments and SPH simulations. The superscript a indicates two values for $x = 0.1$ and 0.25 m, respectively

List of Figures

Figure 1.1. The confluence of the Thompson and Fraser Rivers in Lytton, B.C., Canada (from <http://twistedstifter.com/2012/04/confluences-around-the-world/>)

Figure 1.2. Large coherent structures in the SML are made visible by injecting the dye (from van Prooijen et al. 2000)

Figure 2.1. Mixing width growing rate curve, fitted to experimental data (from Chu and Babarutsi 1988)

Figure 2.2. The measured and modelled values of the mixing layer width for two cases where the flow depth is 42 and 67 mm. The plus and dot symbols represent the experimental data, and the solid and dash lines represent δ from Eq. 2.8 (from van Prooijen and Uijttewaai 2002)

Figure 2.3. Velocity contour line at cross-sections at different longitudinal distances from the splitter plate. From upstream to downstream, the cross-sections are A, B, C, D, and E (from Sukhodolov et al. 2010)

Figure 2.4. Normalized width of SML as a function of x . Solid line **a** represents FML, the solid lines labelled as **b** indicate solutions of Eq. 2.9, the dash lines with symbols show the results of a parabolic function, and line **c** represents the exponential function for the determination of SML width (from Sukhodolov et al. 2010)

Figure 2.5. Velocity magnitude in streamwise direction and mean velocity vectors (from Sukhodolov et al. 2017)

Figure 2.6. Scalar concentration in the mean and instantaneous flow: (a) flow depth is equal to 67 mm; (b) flow depth is equal to 33.5 mm (modified from Cheng and Constantinescu 2020)

Figure 2.7. Sketch of the computational domain used to study confluent flows between non-parallel streams (from Cheng and Constantinescu 2021)

Figure 3.1. Random position error in the x, y, and z-directions (panel a, b, c respectively) for flow mean velocity of 0.19 m/s (from Akutina 2015)

Figure 3.2. Eulerian data from 3D-PTV (a) Discrete radial bins at a distance of r_i from the pipe centerline, (b) Number of velocity vectors collected at each bin as a function of distance from the wall (modified from Oliveira et al. 2013)

Figure 3.3. Ensemble averaged velocity profile for 3D-PTV and DNS data (from Oliveira et al. 2013)

Figure 3.4. Top view of the recirculating flume used for PTV experiments

Figure 3.5. Variable flow drive to control the flow rate in experiments

Figure 3.6. Typical point gage for measuring flow depth

Figure 3.7. Filtering apparatus used to remove external particles before recirculating into the flume

Figure 3.8. Flowchart of sequential stages of data acquisition and processing using 3D PTV

Figure 3.9. Rice bran wax microspheres sample used as seeding particle in PTV experiments (from Martinez 2017)

Figure 3.10. Calibration object for 3D-PTV used in this study (left: design, right: machined)

Figure 3.11. Camera setup (left) and illumination setup (right) for PTV experiments

Figure 3.12. Example of PTV images uploaded in OpenPTV environment

Figure 3.13. Preparation of calibration images using a MATLAB script

Figure 3.14. Parameter setting for calibration for OpenPTV

Figure 3.15. Co-linearity condition for image recording (from Maas et al. 1993)

Figure 3.16. The Final stage of calibration, dots on the calibration object are all recognized by OpenPTV

Figure 3.17. Schematic definition of multivariate outlier on u-v plane. Point A is not defined as a multivariate outlier whereas point B is defined as a multivariate outlier

Figure 3.18. Illustration of SVM method for classifying data: (a) non-optimal hyperplane; (b) optimal hyperplane

Figure 3.19. IsolationForest illustration to detect outlier. Outliers are more susceptible to isolation and hence have short path lengths: (a) data recognized by tree (b) outlier detected by tree (c) comparison of the average path length of data and outlier (Liu et al. 2008)

Figure 3.20. u - v scatter plot of four samples filtered by SVM method. Yellow dots are the regular and purple dots are the outliers. (A) sample A, (B) sample B, (C) sample C and (D) sample D

Figure 3.21. u - v scatter plot of four samples filtered by IsolationForest method, (A) sample A, (B) sample B, (C) sample C, and (D) sample D

Figure 3.22. Schematic of a box plot components

Figure 3.23. Data distribution for data before (raw data) and after (R-forest and SVM) filtering methods for four samples: (A) sample A; (B) sample B; (C) sample C; (D) sample D

Figure 3.24. Ensemble average velocity profile for $V_r=1.5$ at three distances from the splitter plate (0.01, 0.1, 0.25)

Figure 3.25. Ensemble average velocity profile for $V_r=1.14$ at three distances from the splitter plate (0.01, 0.1, 0.25)

Figure 3.26. Mixing width comparison of PTV tests with empirical equations 2.8 and 2.9: (a) $V_r=1.5$; (b) $V_r=1.14$

Figure 3.27. Comparison of velocities for $\Delta t=1/600$ s and $\Delta t=2/600$ s: a, d) velocity in the x -direction; b, e) velocity in the y -direction; c, f) velocity in the z -direction.

Figure 3.28. Sample trajectory of a single particle for $V_r=1.14$

Figure 3.29. Particle trajectories identified by PTV in a SML collected from the same setup as this study (from Martinez 2017)

Figure 3.30. Particle trajectories identified by PTV in a shallow embayment (From Akutina 2015)

Figure 3.31. Trajectories of particles successfully tracked in 150 frames or more for $V_r=1$ as an example output of a script developed in the current study

Figure 3.32. Trajectories of particles successfully tracked in 150 frames or more for $V_r=1.14$

Figure 3.33. Particle trajectories in plane view for $V_r=1.14$

Figure 3.34. FTLE for particle pairs initially located in the mixing layer for $V_r=1.14$

Figure 3.35. FTLE for particle pairs initially located out of the mixing layer for $V_r=1.14$

Figure 3.36. Schematic of particle tracking procedure (from Malik et al. 1993)

Figure 3.37. Flow visualization for streams with the same velocity. The distance between transverse lines is 10 cm

Figure 3.38. Pairing process as a function of downstream distance for $V_r=1.5$

Figure 3.39. Flow visualization for streams with a velocity ratio of 1.14

Figure 3.40. Random Frames of visualization at a specific location (x : 110-170cm) and different times for velocity ratio equal to 1.5

Figure 3.41. Deformation of eddies with elliptic shape for velocity ratio equal to 1.5

Figure 3.42. Eddy spacing histogram for $0 < x < 2$ m downstream of the splitter plate for $V_r = 1.5$

Figure 3.43. Eddy spacing distribution in the downstream distance (x) for $V_r = 1.5$

Figure 4.1. Effect of strong gradients of a function on truncation error. The finite-difference approximation (slope of dashed line) accurately reproduces $\partial v / \partial x$ but not $\partial u / \partial x$. (from Zikanov, 2019)

Figure 4.2. Representation of the kernel function (W) and its domain of influence for particle P (from Fraga Filho et al. 2019)

Figure 4.3. Preprocessing, processing, and post-processing stages of DualsSPHysics model (modified from DualsSPHysics manual)

Figure 4.4. Open boundary treatment in DualsSPHysics (modified from DualsSPHysics manual)

Figure 4.5. Applying shifting algorithm to avoid particle agglomeration in SPH simulation (modified from DualsSPHysics wiki)

Figure 4.6. (a) Mixing layer in the confluence of the rivers Rhona and Arve in Geneva, Switzerland (modified from World Geography); (b) plane view of a definition diagram of a shallow mixing layer, where the tip of the splitter plate is at point O; (c) elevation view of a channel cross-section

Figure 4.7. 2D support domain of particle P with 21 particles (modified from Fraga Filho 2019)

Figure 4.7. Effect of smoothing length on flow patterns in a dam break example at different time steps (from Wu et al. 2016)

Figure 4.8. Transverse profiles of streamwise velocity at selected streamwise distances from the tip of splitter plate for: a) Run A with $V_r = 1.15$; b) Run B with $V_r = 1.5$; c) Run C with $V_r = 3$. The model time is $t = 7$ s.

Figure 4.9. Vorticity contour for a) Run A, b) Run B and c) Run C, $t = 7$ s

Figure 4.10. Schematic illustration of the roll-up process in mixing layer (from Lesieur 2008)

Figure 4.11. Particle vorticity field at $x < 0.3$ m for Run A

Figure 4.12. The pairing process for two selected vortices at $V_r = 1.5$ from $t = 3.05$ s (a) to $t = 3.2$ s (d) and one example frame of the dye visualization experiment (e)

Figure 4.13. The pairing process for Run C at $t = 6$ s (a) to $t = 6.5$ s (e)

Figure 4.14. Okubo-Weiss parameter contours for three different velocity ratio a) Run A, b) Run B, c) Run C, at $t = 7s$

Figure 4.15. Fluid mass transfer along the channel centerline ($y = 0$) for (a) Run A (b) Run B and (c) Run C

Figure 4.16. Fluid momentum transfer along the channel centerline ($y = 0$) for (a) Run A (b) Run B and (c) Run C

Figure 4.17. Mixing layer width for a) Run A, b) Run B, and c) Run C at $t = 7s$ compared to Equations 2.8 and 2.9

List of Symbols

a	acceleration (m/s ²)
b	channel width (m)
C_{sm}	dimensionless Smagorinsky constant
c	principal distance (m)
\bar{c}_{ab}	average speed of sound determined for particles a and b (m/s)
c_f	dimensionless bed-friction coefficient
\bar{c}_f	dimensionless mean bed friction coefficient
c_s	speed of sound (m/s)
D	number of dimensions (1D, 2D or 3D).
d	depth of flow (m)
d_1	univariate threshold of an outlier in u (m/s)
d_2	univariate threshold of an outlier in v (m/s)
dp	particle spacing (m)
$d(k)$	distance between two particle trajectories from frame k (m)
d_{ij}	Euclidean distance between two eddies (m)
E_r	error in particle position from PTV (m)
e_i^f	most likely expected location of particle in frame f (m)
F	focal distance of camera
f	frame number
Fr	Froude number
g	gravitational acceleration (m/s ²)
h	smoothing length (m)
i	grid node identity
K	scaling factor of support domain for kernel function
k	sequential number of frames
l_m	eddy spacing (m)
m_b	mass of particle b (kg)

N	normal vector to hyperplane
n_0	data outlier
n_i	normal data point
P	pressure (Pa)
p^i	projection of X_i onto the normal vector ($i = 1$ or $i = 2$)
Q	discharge (m^3/s or L/s)
r	position of particle
r_i	distance from pipe centerline (m)
R	pipe radius (m)
R_{PTV}	3×3 rotation matrix
R_x, R_y, R_z	angles of rotation around x-, y-, z- axis
Re	Reynolds number
S	dimensionless stability parameter
S_c	dimensionless critical value of S
S_n	normal stress (s^{-1})
S_s	shear stresses (s^{-1})
S_{xy}	velocity strain rate (s^{-1})
t	time (s)
$t_{a,i}$	particle identity for particle a at time step i
\bar{U}	average velocity across the mixing layer $\bar{U} = (U_1 + U_2)/2$ (m/s)
\bar{U}_0	average velocity across the mixing layer at the initial contact location (m/s)
U_1	streamwise velocity of the low-speed stream (m/s)
U_2	streamwise velocity of the high-speed stream (m/s)
U_c	streamwise velocity in the centre of the mixing layer (m/s)
U_{c0}	streamwise velocity in the centre of the mixing layer at the initial contact location (m/s)
U_{cp}	mean centerline pipe velocity (m/s)
U_z	mean axial velocity in pipe (m/s)
U_{3D-PTV}	streamwise component of flow velocity in pipe from PTV (m/s)
U_{DNS}	streamwise component of flow velocity in pipe from DNS (m/s)

u	streamwise component of flow velocity (m/s)
u_i	streamwise velocity at grid node i (m/s)
\bar{u}	ensemble average streamwise velocity (m/s)
u_*	shear velocity (m/s)
\mathbf{V}	velocity vector (m/s)
v	spanwise component of flow velocity (m/s)
v_b	volume of particle b (m^3)
V_r	dimensionless velocity ratio
W	kernel function in SPH
w	vertical component of flow velocity (m/s)
X_0	lateral coordinate of the projection center (m)
X_i	object point coordinate in the lateral direction (m)
X_1, X_2	support vectors
x	longitudinal coordinate (m)
x_0	virtual origin in the longitudinal direction (m)
x_h	principle point coordinate in the lateral direction (m)
x_i	particle trajectory from frame i (m)
x_j	particle trajectory from frame j (m)
x_p	horizontal coordinate of the pinhole relative to the center point of image (m)
x'_i	projection of x_i (m)
x_i^f	location of particle i from frame f (m)
x^*	dimensionless downstream distance of SML
Y_0	longitudinal coordinate of the projection center (m)
Y_i	object point coordinate in the longitudinal direction (m)
y	transverse coordinate (m)
y_h	principal point coordinate in the vertical (m)
y_p	coordinate of the pinhole relative to the center of image in the longitudinal direction (m)
y'_i	projection of y_i (m)
Z_0	vertical coordinate of the projection center (m)

Z_i	object point coordinate in the vertical direction (m)
z	vertical coordinate (m)
z_0	roughness height (m)

Greek

α	dimensionless entrainment coefficient
α_{sph}	dimensionless coefficient in artificial viscosity
β	dimensionless constant
Δr	spatial interval in the radial direction (m)
ΔS	displacement of a seeded particle in time interval Δt (m)
Δt	time interval (s)
ΔU	velocity difference across the mixing layer (m/s)
ΔU_0	velocity difference across the mixing layer at the initial contact location (m/s)
Δx	grid size (m)
δ_0	initial mixing layer width (m)
δ^*	dimensionless mixing layer width
θ	dimensionless characteristic mixing layer parameter
κ	von Karman constant
λ_i	dimensionless scaling factor
ν	kinematic viscosity of water (m ² /s)
ν_0	kinematic viscosity (m ² /s)
ν_t	turbulent eddy viscosity (m ² /s)
ξ	dimensionless polytropic index
ρ_b	density of particle b (kg/m ³)
σ_u	variance of u (m ² /s ²)
σ_v	variance of v (m ² /s ²)
τ	SPS stress tensor (pa)
ω	vorticity (s ⁻¹)
ω_{ij}	average vorticity (s ⁻¹)

Ω Okubo-Weiss parameter (s^{-1})

List of Abbreviations

ADV acoustic doppler velocimetry
BL boundary layer
CFD computational fluid dynamics
CSs coherent structures
CFL Courant–Friedrichs–Lewy
DES detached eddy simulation
DBC dynamic boundary condition
FML free mixing layer
FTLE finite-time Lyapunov exponents
fps frame per second
LED light emitting diode
MPS moving particle semi-implicit
OPC open boundary condition
PIV particle Image velocimetry
PTV particle tracking velocimetry
RBW rice bran wax
SML shallow mixing layer
SPS sub-particle scale
SOV streamwise-oriented vortical
SVM support vector machine
VFD variable flow drive
VoF volume of fluid
3-D three-dimensional
2-D two dimensional

Chapter 1

Introduction

1.1 Introduction and problem definition

River flows are usually considered shallow flows because the horizontal dimensions of river channels are typically much larger than their vertical dimension. When a shallow flow is subject to lateral shear from a transverse velocity gradient, a mixing layer is formed. The development of a shallow mixing layer (SML) at a river confluence can be affected by various factors, including but not limited to the velocity difference between the two streams entering the confluence, bed friction and the shallowness of the flow. An example of a SML is presented in Figure 1. 1.



Figure 1.1. The confluence of the Thompson and Fraser Rivers in Lytton, B.C., Canada
(from <http://twistedgifter.com/2012/04/confluences-around-the-world/>)

Significant research attention has been paid to the contribution of large-scale two-dimensional (2-D) turbulent flow structures in the transport of momentum and mass in various flow configurations such as wakes, shear layers, and flows around bends. 2-D coherent structures (2DCSs) are defined as large-scale instabilities originating from internal oscillations caused by local or distributed disturbances in flows. These coherent structures are two-dimensional because of their large aspect ratio or the ratio of the lateral or horizontal length scale to the vertical scale. It is noteworthy that all turbulent flows are essentially three-dimensional (3-D), and for each direction, the characteristic-length and velocity scales that apply to the largest eddies (or coherent structures) can be identified. For shallow flows, this results in a strong anisotropy at the largest scales. The turbulence properties are not linked to the flow geometry at smaller scales, increasing

isotropy towards the smallest dissipative scales. However, the vertical scale of the largest flow structures is limited by the flow depth in shallow flows. Sufficiently large separation of vertical (depth) and horizontal length scales can result in a flow with little interaction between 3-D boundary layer turbulence and the 2-D horizontal motions. In this case, the flow can be considered two-dimensional with an approximately uniform eddy viscosity.

The vortex stretching mechanism is very effective in three-dimensional turbulence in transferring energy to eddies of smaller length scales. Since this mechanism is suppressed in shallow flow, horizontal eddies are relatively large and long-lived, as found in two-dimensional turbulence. Generally, 2DCSs extend uniformly over the full water depth. Some examples of 2DCSs can be seen in shallow jets, shallow wakes and the flows over a solid cylinder. 2DCS can be distinguished from their generation mechanisms (Jirka 2001). The most frequent mechanism to form 2DCS is topographical forcing from islands, headlands, and groynes, causing an intense transverse shear layer classified as Type A. Type B usually emanates from internal transverse shear instabilities caused by lateral velocity variations. Examples are shallow jets, shallow mixing layers and shallow wakes. Figure 1.2 shows an example of 2DCS formed in a SML and visualized by a dye.

In contrast to Type A, Type C is the weakest type of mechanism of 2DCS generation, which is related to secondary instabilities in the base flow produced by the shear effect at the solid bottom. The base flow contains 3D turbulence structures or burst events. If any slight imbalance happens in this flow, such as spatial or temporal tidal oscillations or submerged islands, it leads to 2DCS-type C.

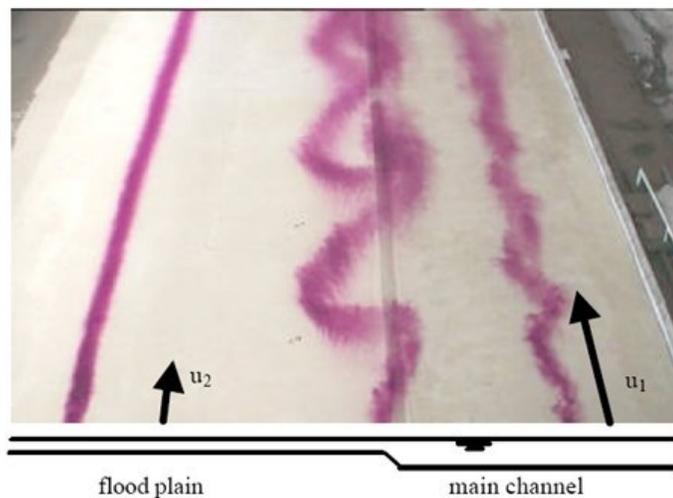


Figure 1.2. Large coherent structures in the SML are made visible by injecting the dye (from van Prooijen et al. 2000)

Horizontal coherent structures with dimensions of the order of the width of the mixing layer develop in the mixing layer due to instabilities in the transverse shear flow. In a flow with two parallel flows of different velocity magnitude, momentum, travelling with the flow downstream, is transferred from the high-velocity side to the low-velocity side, resulting in the broader mixing region and a smaller velocity gradient.

1.2 Motivation

When two parallel shallow streams with different velocities flow side-by-side, a mixing layer is developed. Shallow mixing layers occur naturally at river confluences (Figure 1.1), sudden channel expansions, harbour entrances, and groynes. The present study is motivated by several impacts of SML on the riverine environment, river ecosystems and hydraulic engineering design. Investigating shallow mixing layer characteristics is considered eminent since river confluences offer favorable living conditions to fish and other aquatic organisms.

A better understanding of the complexity of turbulent flow structures in shallow mixing layers will help understand the contribution of the mixing process in the transverse transport of momentum and mass affecting river morphology, sediment transport and transverse pollutant in river confluences.

The high level of energy of 2DCS makes it crucial to study their dynamics to understand processes such as momentum and mass exchange and the dynamic loads exerted by flow over hydraulics infrastructure. Therefore, the description of their spatial characteristics and temporal dynamics is paramount in environmental processes and hydraulic engineering.

Various aspects of SML have made them a popular topic of study in the last decades. Investigations of shallow mixing layer characteristics can be divided into two main categories of Eulerian and Lagrangian approaches. The Eulerian approach looks at fluid motion at specific locations in the space through which the fluid particle flows as time passes. The Lagrangian approach looks at fluid particle motion where the observer follows an individual fluid particle as it moves through space and time. Most past studies have an Eulerian basis. The main reason for this is the technical difficulties of Lagrangian approach in both laboratory and numerical setups. Overall, both Lagrangian and Eulerian analyses are necessary to create a full picture of mixing processes in SML. In this study, advanced experimental and numerical tools were utilized to provide both Eulerian and Lagrangian analyses of SML.

1.3 Objectives and thesis layout

Previous studies reported the effect of flow depth on the shallow mixing flow characteristics (Uijtewaal and Booij, 2000; van Prooijen et al., 2000). The effect of the velocity ratio is less reported. The velocity ratio contributes to SML development, defined as the ratio of streamwise velocities of two parallel streams. Mehta (1991) investigated the effect of velocity ratio on the plane mixing layer where the vertical confinement has no significant impact on large horizontal coherent structures. However, the effect of the velocity ratio on shallow mixing flow characteristics by considering both Lagrangian and Eulerian perspectives, has not been studied. Furthermore, a systematic understanding of the turbulent flow field development of river confluence can be achieved by using both advanced measurement techniques and numerical simulations.

This study explores how the mixing processes occur in shallow mixing layers by considering various velocity ratios between two parallel streams. The Lagrangian and Eulerian approaches are employed to shed light on fluid particle movements in SML and eddy evolution processes. In addition, the mean flow field is studied to provide information on mixing layer growth, mass and momentum transfer. To examine SML visualization based on the Lagrangian and Eulerian

perspectives, this research used the particle tracking velocimetry (PTV) technique in laboratory experiments and smoothed particle hydrodynamics (SPH) for numerical simulation.

The main questions of the present study are:

- How does SML behave under different ratios of incoming flow velocities?
- How the shear caused by the velocity gradient influences the particle trajectories?
- To what extent do the wakes downstream the splitter plate affect the vortical motions?
- Are transverse profiles of streamwise velocities self-similar?
- To what extent previous relations proposed to determine mixing layer width are practical?
- What factors control the pairing of adjacent eddies and the growth of large-scale coherent structures?
- How are the fluid mass and momentum exchange affected by the velocity ratio of two streams?
- This thesis consists of five chapters. Chapter 1 covers the problem definition and the objectives of the research.

In Chapter 2, the background of the study focused on mixing flows is investigated. The flows are categorized into the plane (free) mixing layer and shallow mixing layer.

Chapter 3 consists of the experimental portion of this study. SML visualization using dye is analyzed in this chapter to create visual evidence of SML growth and SML evolutions from the splitter plate to downstream. Shallow mixing flow experiments using PTV in the near field region of the mixing layer (0-30 cm) for two velocity ratios of 1.14 and 1.5 will be presented in Chapter 3. In addition, potential improvements in PTV data acquisition and data analysis are discussed in this chapter.

In Chapter 4, a description of computational fluid dynamics (CFD) and its necessity in expanding knowledge of SML are given. SPH as a meshfree method to simulate turbulent flow is introduced. This chapter presents a shallow mixing flow simulation using SPH in the near-field and far-field regions of the mixing for various velocity ratios.

In Chapter 5, conclusions of the current research and future work suggestions are presented.

Chapter 2

General background

2.1 Introduction

When two parallel shallow streams with different velocities flow side-by-side, a mixing layer is developed. Mixing layer has been studied in two main categories: Free (plane) mixing layer (FML) and SML. In FML, the 'free' implies no influence of a wall and that the turbulence arises from the mean-flow gradients. However, in shallow flows, the presence of a solid bed boundary induces strong frictional effects and hence losses of energy that reduce the velocity difference, slowing the spreading rate.

The bed confinement influences the mixing processes in a transverse shear flow. In a shallow flow, the effect of bed confinement on mixing layer characteristics is considerable. However, this effect is not significant in a free mixing layer. The following sections provide a better description of the differences between the SML and FML. Studying the FML goes back to decades ago, and those studies have created valuable insights into processes involved in mixing flow (Jimenez et al. 1985; Roshko 1981). Recent studies investigated SML through experiments in the laboratory (Higham et al. 2018; Han et al. 2017; Chu and Baburutsi 1988) and field studies (Sukhodolov et al. 2017). In addition, SML has been studied using stability analysis (Chu et al. 1991) and numerical simulation (Cheng and Constantinescu 2020; Constantinescu et al. 2011).

2.2 Free mixing flow

The free mixing happens where bed friction has no significant effect on large horizontal coherent structures. In FML, the velocity difference and the associated gradient between two streams are a source of turbulent kinetic energy (Uijtewaal and Booij 2000). Self-similarity of transverse profile of velocity deduced that the width of a two-dimensional FML grows at a constant rate.

Self-similarity implies that a profile function can describe the transverse profiles of the streamwise velocity. The mean velocity field is thus determined by the velocity difference across the mixing layer and the width of the mixing layer.

Therefore, the width of the FML is defined as follows:

$$\delta = \frac{U_1 - U_2}{(\partial u / \partial y)_{max}} \quad (2.1)$$

where U_1 and U_2 are the streamwise velocity of two mixing streams. The evolution of a free mixing layer is described by an ordinary differential equation (Brown and Roshko 1974).

$$\frac{d\delta}{dx} = \alpha \frac{\Delta U}{\bar{U}} \quad (2.2)$$

where ΔU is the velocity difference, and $\bar{U} = (U_1 + U_2)/2$. Uijtewaal and Booij (2000) rewrote the growing rate of the FML to be proportional to the characteristic mixing layer parameter $\theta = (U_1 - U_2)/(U_1 + U_2)$.

$$\frac{d\delta}{dx} = 2\alpha\theta \quad (2.3)$$

In Eq. 2.2, α is the entrainment coefficient, which has a value of 0.11 ± 0.03 for FML (Pope 2000). For SML, there are different values of α reported in the literature, such as 0.11 (Uijttewaala and Booij 2000) and 0.18 (Chu and Babarutsi 1988).

For FML, the growing rate is constant from upstream to downstream, and thus the width of the layer increases linearly. However, in SML, the presence of a solid bed boundary induces strong frictional effects and hence losses of energy that reduce the velocity gradient, slowing the spreading rate.

The vertical constraint in shallow flows has been reported to be a contributing factor in developing turbulent structures (Jirka and Uijttewaala 2004; Higham et al. 2018). Another factor affecting the flow is the bed roughness. The roughness plays a role in stabilizing large-scale turbulent motions developing in the flow. It has effect on the development of boundary layer, associated velocity gradient and three-dimensional small-scale turbulent motions.

Important length scales and velocity scales of the shallow mixing layer are the flow depth, d ; the velocity difference across the mixing layer, ΔU ; the average velocity across the mixing layer, \bar{U} ; and the bed-friction length scale, d/c_f used to normalize longitudinal distance.

2.3 Shallow mixing flow (SML)

Chu and Babarutsi (1988) studied the effect of confinement and bed friction on transverse mixing flows. They measured transverse velocity profiles at longitudinal distances $x = 0.01, 0.5, 1, 1.5, 2,$ and 3m from the tip of the splitter plate, at 0.005 m below the water surface. The results showed the presence of the mixing layer, as manifested by the steep velocity gradient in the mean profiles and the peak in the root-mean-squared velocity profiles in the mixing layer.

The entrainment coefficient, α , can be defined as the ratio of the entrainment velocity, $d\delta/dt$, to the velocity gradient, ΔU

$$\alpha = \frac{d\delta/dt}{\Delta U} = \frac{1}{\Delta U} \frac{dx}{dt} \frac{d\delta}{dx} = \frac{\bar{U}}{\Delta U} \frac{d\delta}{dx} \quad (2.4)$$

Defining the dimensionless mixing layer width as $\delta^* = \frac{c_f \delta}{d} \cdot \frac{\bar{U}_0}{\Delta U_0}$ as well as dimensionless downstream distance, $x^* = \frac{c_f x}{d}$, Chu and Babarutsi (1988) showed that the initial entrainment coefficient of the SML was equal to a constant value of 0.18, which is almost twice as large as that of the FML. This result was questioned by later findings claiming that the initial growth rate of the SML is similar to that of the FML (Uijttewaala and Tukker 1998). In the far-field region where x is large compared with d , the influence of the bed friction becomes more critical, resulting in the reduction of the spreading rate of the mixing layer width.

In the definition of dimensionless mixing layer width, c_f is the bed-friction coefficient, which, for a logarithmic velocity profile and a smooth bottom with roughness height ($z_0 = 0.135v / u_*$), is obtained as follows (Uijttewaala and Booij 2000):

$$\frac{1}{\sqrt{c_f}} = \frac{1}{\kappa} (\ln(Re \sqrt{c_f}) + 1.0) \quad (2.5)$$

Here, z_o is roughness height; ν is the kinematic viscosity of water; u^* is shear velocity; κ is the Von Karman constant; Re is the Reynolds number.

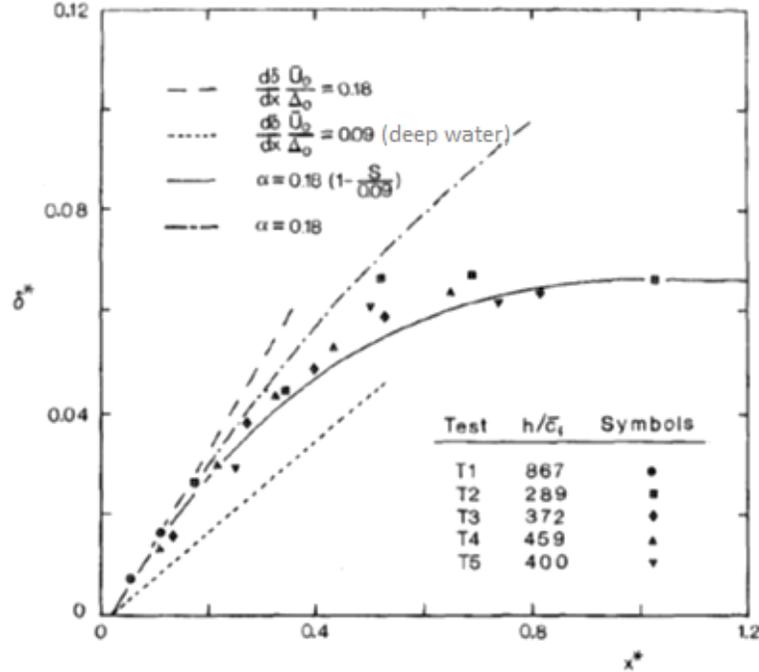


Figure 2.1. Mixing width growing rate curve, fitted to experimental data (from Chu and Babarutsi 1988)

Another characteristic of the SML is the stability parameter or bed friction number that quantifies the stabilizing effect of bottom friction on the shallow mixing layer (Chu and Babarutsi 1988). This number is defined as the ratio of the stabilizing effect of the bed friction to the destabilizing effect of transverse shear resulting in the production of turbulence kinetic energy (Uijttewaal and Booij 2000). The stability parameter is defined as

$$S = \frac{c_f \delta \bar{U}}{d \Delta U} \quad (2.6)$$

A critical value, Sc , between 0.06–0.12 denotes the equilibrium of production to dissipation (van Prooijen and Uijttewaal 2002).

In the case of an analytical approach to studying shallow mixing flows, stability analysis gives an insight into the mechanisms of 2DCS by simplifying momentum equations to linearized depth-averaged equations. Although these equations consider the flow purely parallel rather than expanding and a simple eddy viscosity formulation is used in them, they give a good view of the forms and growth mechanisms of 2DCS.

Using stability analysis, Chu and Babarutsi (1988) showed that the entrainment coefficient becomes negligible when the stability parameter exceeds the critical value of 0.09. This critical value is in the range of values obtained from a stability analysis for parallel flows (Chu et al. 1983;

Alavian and Chu 1985). The linear stability theory indicates when a particular flow becomes unstable, and this theory can be used to describe the structure of critical motions that take place above the threshold.

In another study, van Prooijan and Uijttewaal (2002) used numerical simulation to study the development of coherent structures in SML considering different flow depths. They verified the numerical simulation results using measurements from a series of experiments. They used depth-averaged shallow water equations in the streamwise and transverse directions to model the mean flow field based on self-similarity. van Prooijan and Uijttewaal (2002) applied a hyperbolic tangent function to describe transverse profiles of mean streamwise velocity because the function fitted their experimental data well.

van Prooijan and Uijttewaal (2002) proposed a relation for velocity difference across the mixing layer (Eq. 2.7) and substituted that into mixing layer growth rate relation and proposed a relation for SML width as a function of x (Eq. 2.8).

$$\Delta U(x) = \Delta U_0 \exp\left(-\frac{\bar{c}_f}{d} x\right) \quad (2.7)$$

$$\delta(x) = \delta_0 + \alpha \frac{\Delta U_0 d}{U_{c0} \bar{c}_f} \left(1 - \exp\left(-\bar{c}_f \frac{x}{d}\right)\right) \quad (2.8)$$

Subscript 0 indicates the initial value at the contact point of two streams. δ_0 is the initial mixing layer width introduced as a constant of integration; U_c is the streamwise velocity in the centre of the mixing layer. The initial width, δ_0 , is imposed by the thickness of the boundary layers developed on both sides of the splitter plate and is approximately equal to the flow depth (van Prooijan and Uijttewaal 2002). The mean bed friction coefficient is defined as $\bar{c}_f = (c_{f1} + c_{f2})/2$.

van Prooijan and Uijttewaal (2002) suggested that the effect of shallowness on the mean flow field is mainly governed by the decrease in the velocity difference across the mixing layer due to the bottom friction. Figure 2.2 presents a comparison between Eq. 2.8 and experimental results for two different flow depth cases.

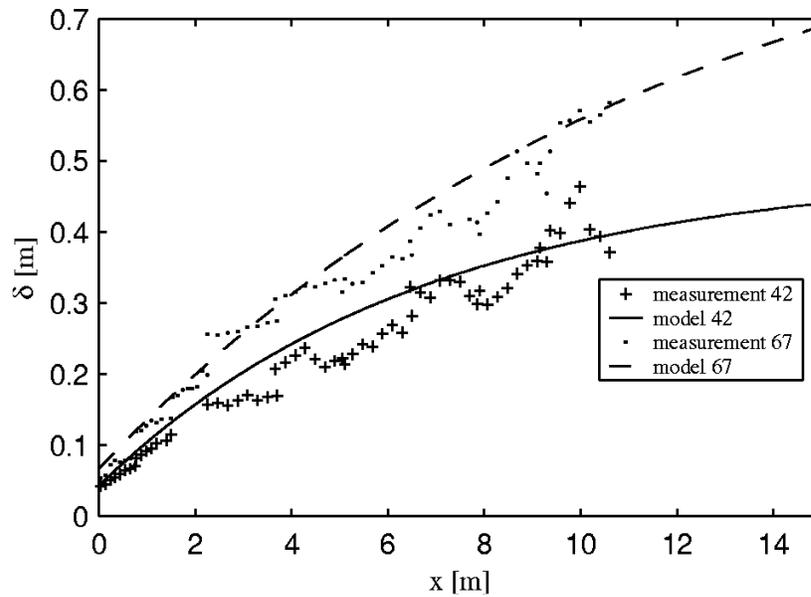


Figure 2.2. The measured and modelled values of the mixing layer width for two cases where the flow depth is 42 and 67 mm. The plus and dot symbols represent the experimental data, and the solid and dash lines represent δ from Eq. 2.8 (from van Prooijan and Uijttewaai 2002)

Sukhodolov et al. (2010) studied a SML in a straight reach of a natural river. They showed that the mixing layer is three-dimensional in contrast to the two-dimensional structures observed in the laboratory.

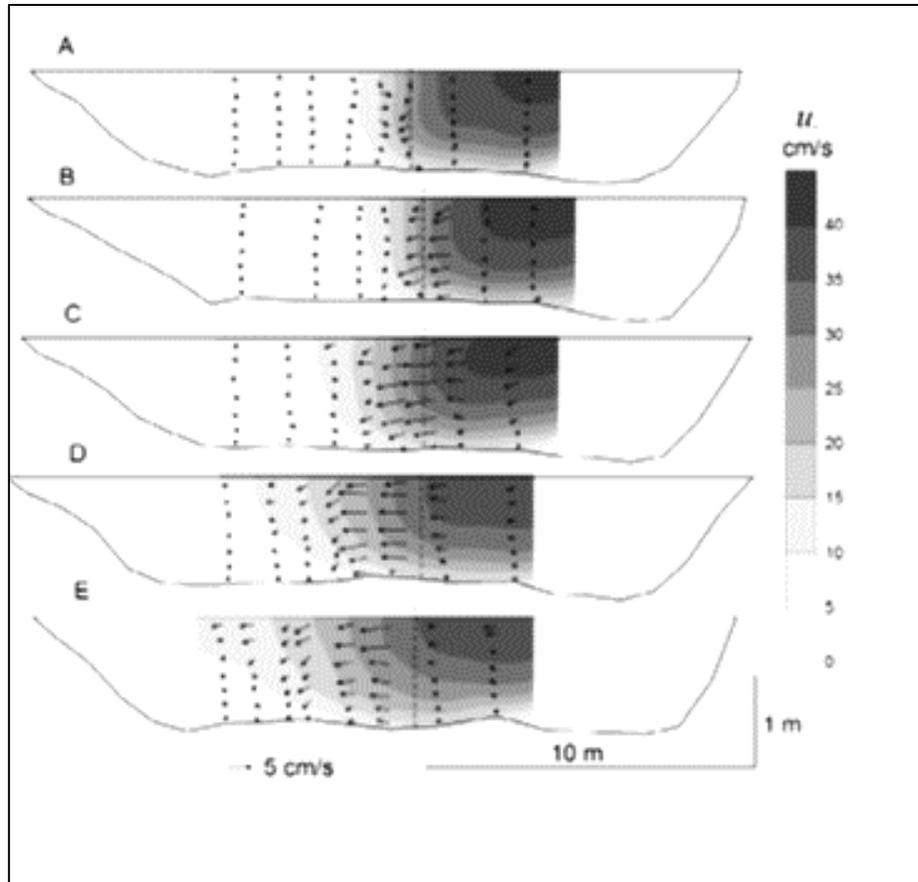


Figure 2.3. Velocity contour line at cross-sections at different longitudinal distances from the splitter plate. From upstream to downstream, the cross-sections are A, B, C, D, and E (from Sukhodolov et al. 2010)

In their study, measurements of three flow velocity components were made by an array of acoustic doppler velocimeters (ADVs). The mean flow depth and width of the river reaches were equal to 1 m and 25 m, respectively, resulting in a shallow flow. Initially strong, lateral shearing, indicated by the verticality and closeness of velocity contours, diminishes at the downstream sections (Figure 2.3).

As Figure 2.3 shows, the high-velocity core shifts laterally toward the slow-flow side of the channel. The local depth-averaged values of the transverse and vertical velocity components depict a pattern characteristic of helical secondary flow. This pattern reveals the development of a weak recirculation cell on the fast-flow side of the shear layer that grows systematically in size and expands toward the slow stream as the distance from the splitter plate increases. This longitudinal evolution of the lateral momentum flux points to the influence of two mechanisms: redistribution of momentum between the two flows, and the frictional influence of the riverbed. Turbulence kinetic energy patterns showed a relatively narrow vertical band of maximal values in the central part of the channel and a horizontal band of increased turbulence stretching near the riverbed toward the fast stream. Sukhodolov et al. (2010) proposed a parabolic relation for mixing layer width growth based on experimental data of the river confluence

$$\delta(x) = \delta_0 + \alpha \frac{\Delta U_0}{U_{c0}} x \left(1 - \frac{\beta}{2\Delta U_0} x\right) \quad (2.9)$$

in which β is a constant coming from the linear decrease of velocity, $\Delta U = \Delta U_0 - \beta x$. Figure 2.4 shows the comparison of SML width determined from Eq. 2.9 and other methods. The thick solid line in Figure 2.4 corresponds to a constant velocity difference and hence is representative of the dynamics of free mixing layers (line *a*). As the graph shows, measured values of the shear layer width are in agreement with both the parabolic function and the theory of free mixing layers at short and intermediate distances ($x \leq 30$ m) from the origin of the layer, while at large distances ($x > 30$ m), the trends deviate from the free mixing layer to a minor extent. The figure reveals that the exponential function (line *c*) overpredicts the effect of bed friction on the width of the mixing layers.

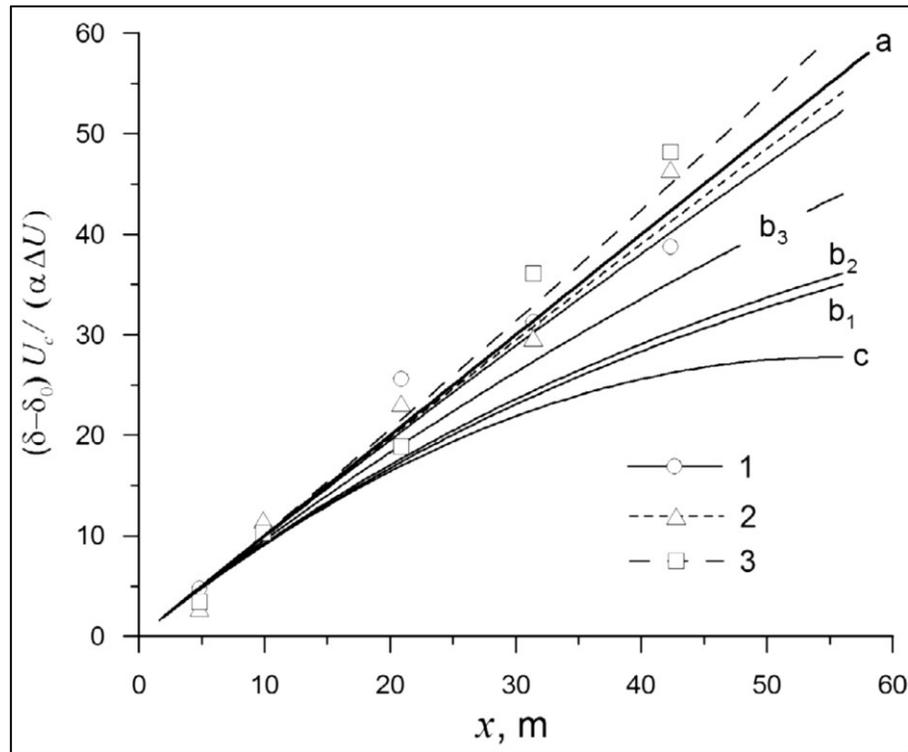


Figure 2.4. Normalized width of SML as a function of x . Solid line **a** represents FML, the solid lines labelled as **b** indicate solutions of Eq. 2.9, the dash lines with symbols show the results of a parabolic function, and line **c** represents the exponential function for the determination of SML width (from Sukhodolov et al. 2010)

Sukhodolov et al. (2017) studied mixing processes in a discordant river. The results support the hypothesis that flows at a discordant alluvial confluence with a velocity ratio greater than two exhibit jet-like characteristics. This jet-like behaviour has important implications for morphodynamical processes at these types of river confluences (Figure 2.5). At this discordant confluence, the jet's highly energetic core is displaced away from the riverbed, thereby inhibiting scour; however, helical motion develops adjacent to the jet, particularly at high flows that may promote scour.

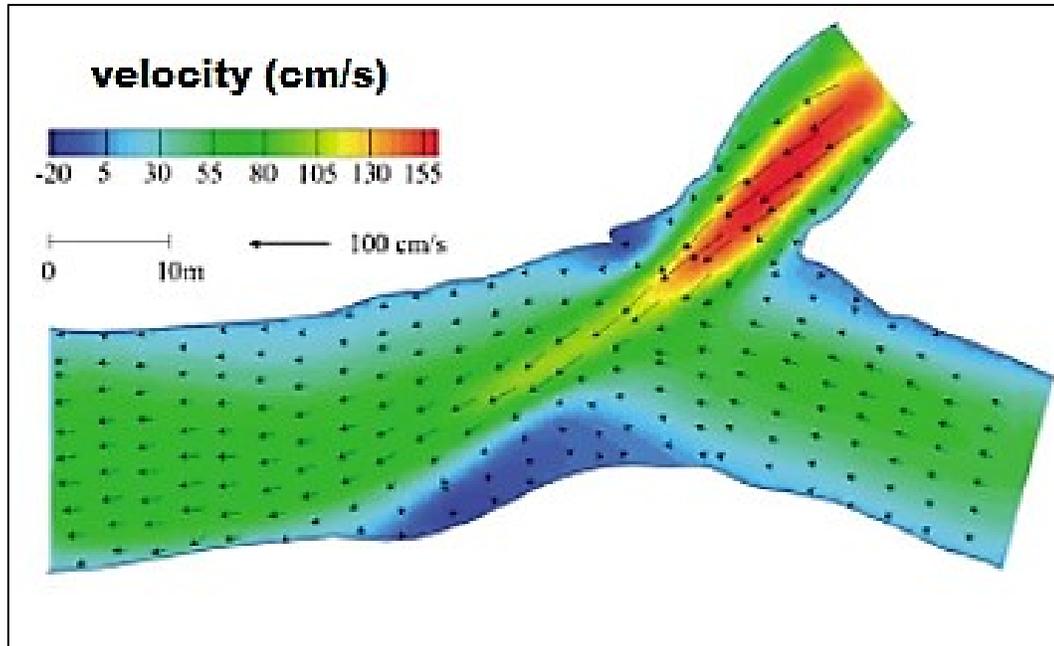


Figure 2.5. Velocity magnitude in streamwise direction and mean velocity vectors (from Sukhodolov et al. 2017)

Cheng and Constantinescu (2020) used detached eddy simulation (DES) to model a SML and compared the results with data from an experiment at which the flow depth was equal to 67 mm and the velocity ratio was equal to 2.2. The scalar concentration of mean flow in Figure 2.6 shows that the mixing layer centerline is pushed toward the low-velocity side due to the gradual equalization of the velocities in the two streams. Cheng and Constantinescu (2020) noticed that decreasing the flow depth to half (33.5 mm) limits the eddy pairing process to $x = 40d$ (Figure 2.6b) compared to $x = 150 - 160d$ in the base case (Figure 2.6a).

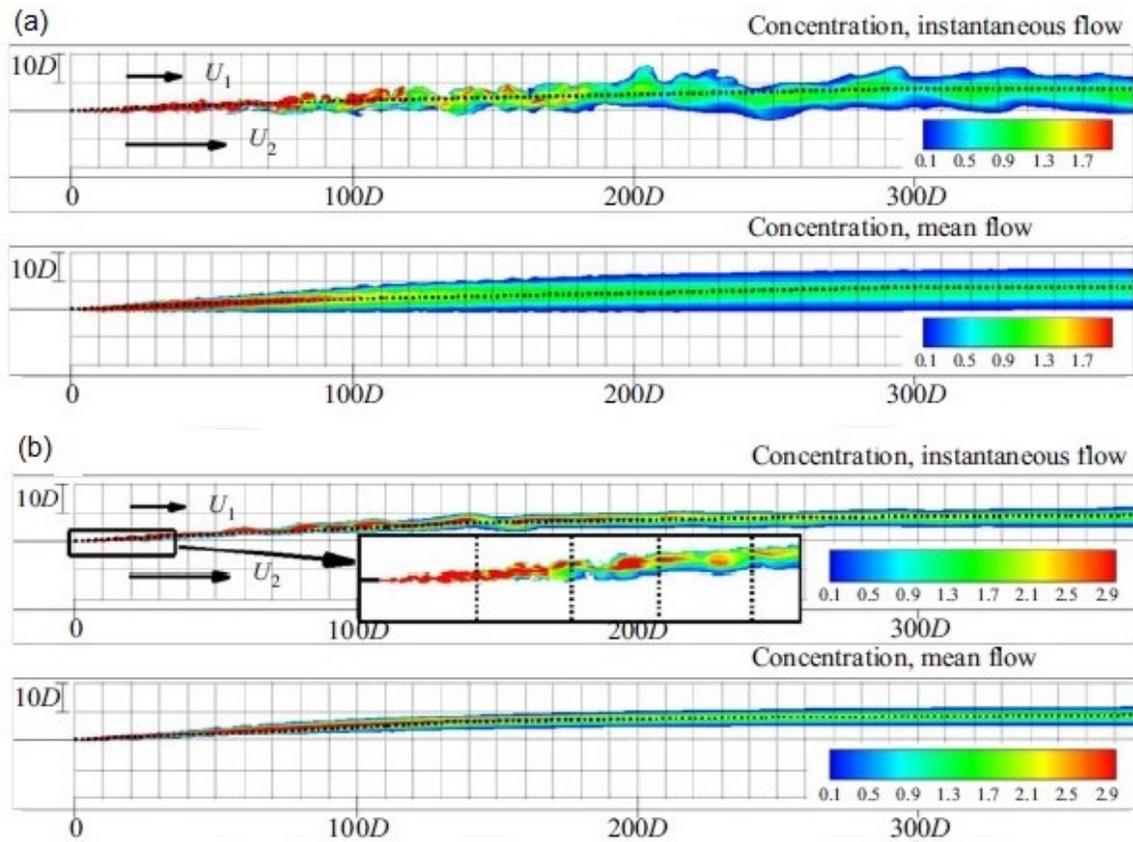


Figure 2.6. Passive scalar concentration in the mean and instantaneous flow: (a) flow depth is equal to 67 mm; (b) flow depth is equal to 33.5 mm (modified from Cheng and Constantinescu 2020)

Cheng and Constantinescu (2021) used DES to model shallow mixing flow between non-parallel streams (Figure 2.7). In this figure, Kelvin-Helmholtz instabilities are shown inside the mixing layer (red circles) and the streamwise-oriented vortical (SOV) cells are located in the vicinity of the mixing layer (black arrow lines). Their results show a strong interaction between SOV cells and Kelvin-Helmholtz instabilities, sharply increasing the mixing.

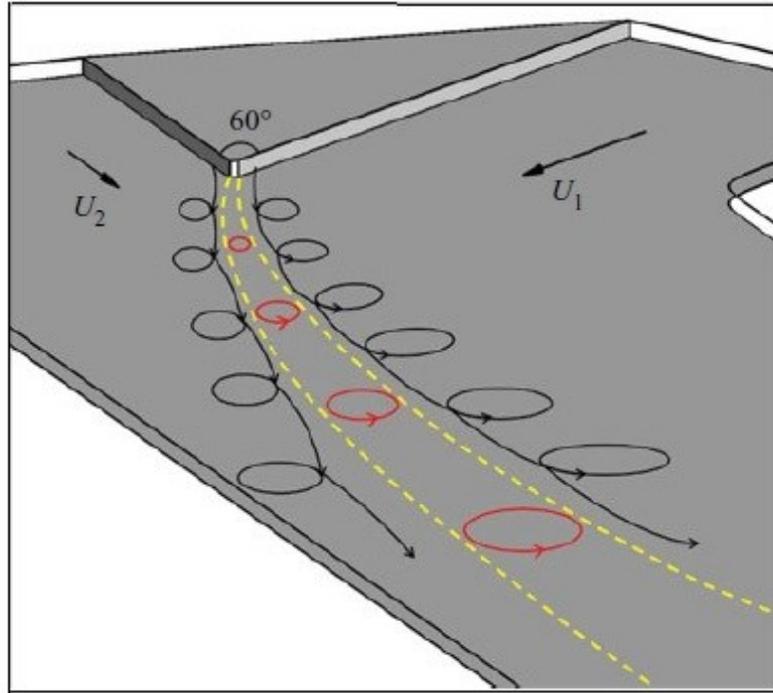


Figure 2.7. Sketch of the computational domain used to study confluent flows between non-parallel streams (from Cheng and Constantinescu 2021)

Previous studies in SML are focused on mean flow field characteristics. However, an investigation of SML from a Lagrangian perspective brings new insights into mixing processes contributing to the mass and momentum exchange. In addition, to the best of our knowledge, the effects of changing velocity ratio on the instantaneous mixing processes and mean flow field characteristics of SML have not been reported in the literature.

Chapter 3

Investigation of SML using particle tracking velocimetry and dye visualization

3.1 Introduction

Particle tracking velocimetry (PTV) is considered an optical flow measurement technique in which neutrally buoyant particles are tracked in consecutive image frames. Recent advances in camera capabilities and computing power for handling large volume data have made PTV a popular technique in turbulent flow measurements. 3D-PTV uses three high-speed cameras that simultaneously capture images from the flow to track seeding particles within the fluid flow and produce particle velocity components in three directions. The 3D-PTV procedure consists of two main stages: data acquisition, and data processing. The data acquisition is the hardware setup where the images are captured through multiple synchronized camera views. The data acquisition step of 3D-PTV is a crucial step of the procedure since it involves many details that the user must take into account. A detailed explanation of the data acquisition procedure is presented in the following sections. The data processing stage includes noise filtration, particle detection, calibration of the multi-camera views, and establishing stereoscopic correspondences, to create 3-dimensional information of flow.

This chapter presents the study of SML using 3D-PTV completed by dye visualization experiments to answer the following questions.

- 3D-PTV has not been applied to obtain the Lagrangian view of the SML so far. This study aims to pave the way for such applications. The main question is how the data acquisition and data process stages can be modified (the changes in the particle size and the illumination setup) to create longer particle trajectories.
- 3D-PTV is an experimental technique with a great challenge to deal with uncertainties, whose most common source is optical distortion (the error that comes into the dataset due to insufficient illumination, flow surface ripples, reflections by the surfaces, etc.). What are the reliable validation techniques to reduce the outlier from the dataset?
- How can the SML be quantified using 3D-PTV, and how are the results compared with previous studies?
- How is the mixing layer width in the near field of SML compared to empirical relation results?

The application of particle tracking methods has become more prevalent in studying turbulent flows. Kim (2015) provided guidelines to accomplish an optimal system set-up of PTV by discussing the hardware configuration, software operation, and post-processing methods.

Arneodo et al. (2008) reviewed challenges in particle tracking techniques, including PTV and particle image velocimetry (PIV), in studying highly turbulent flows. They investigated the intermittency properties of velocity fluctuations during a period of time by comparing data obtained from PTV/PIV techniques and numerical simulations. They stated that the main difficulty of Lagrangian investigations is due to the necessity to resolve the wide range of time scales driving

different particle behaviours, including the large time scales induced by stirring mechanisms to short time scales associated with viscous dissipation.

Several studies utilized particle tracking techniques to investigate shallow flows. Akutina (2015) studied turbulent flow in a shallow embayment using 3D-PTV. The author assessed different components of the velocity error using a small rod with two particles glued onto it; it was moving around in the flow. In the ideal case with no velocity error, the two particles' relative velocity along the vector connecting them must be zero. This method provides a way to measure velocity error in the experiments. Figure 3.1 shows the position error, E_r , for x , y , z location (panels a, b, and c, respectively) determined by subtracting the measured location from the actual location of the particle. As the results show, the position error at the x and y direction is 0.03 mm, while at z -direction, it increases to 0.18 mm. Akutina (2015) concluded that accurate measurements through a free water surface are possible when the average displacement of particles per frame is much smaller than the wavelength of surface oscillations.

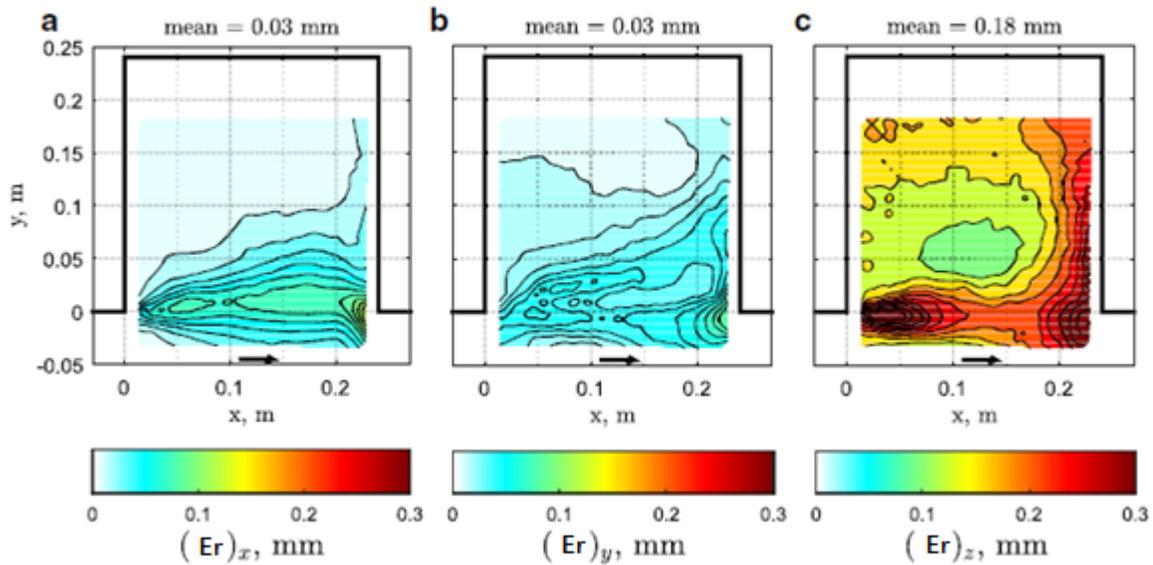


Figure 3.1. Random position error in the x , y , and z -directions (panels a, b, and c, respectively) for flow mean velocity of 0.19 m/s (from Akutina, 2015)

In another study, Oliveira et al. (2013) investigated pipe flow characteristics with 3D-PTV by applying Lagrangian and Eulerian approaches. For Eulerian analysis of 3D-PTV, the velocity vectors were collected in discrete radial bins ('experimental grid' as they stated in their paper). Figure 3.2a shows a schematic of their procedures. Each bin is a cylinder with a radius of $2\Delta r$ where Δr was equal to 0.5 mm. In this figure, $t_{1,50}$ represents the particle at time step 1 of 50 overall time steps. As the figure shows, the particles $t_{1,50}$ to $t_{6,50}$ are collected in bin i .

Figure 3.2b shows the number of velocity vectors collected at each bin from the pipe centerline ($r_i/R = 0$) to the pipe wall ($r_i/R = 1$). As we can see, for $0.2 < r_i/R < 0.8$, the number of velocity vectors is more than 5×10^4 . It is noteworthy noting that in areas close to the pipe wall, the number

of particles recognized from the images is limited, which results in a limited number of velocity vectors collected at bins close to the pipe wall.

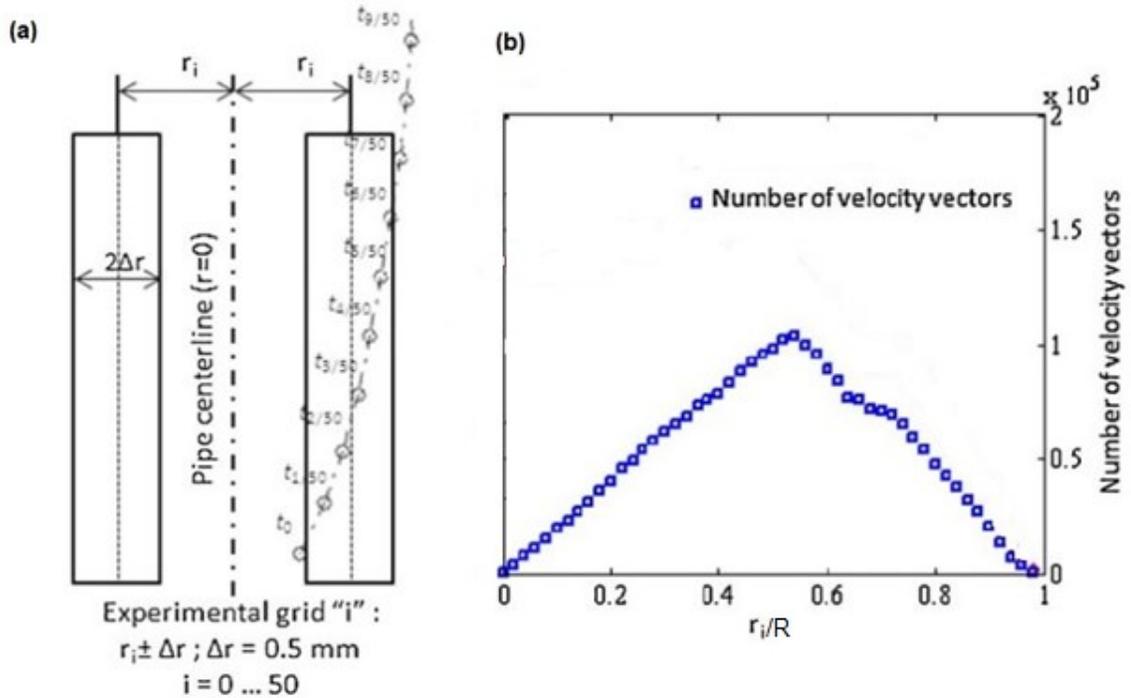


Figure 3.2. Eulerian data from 3D-PTV: (a) Discrete radial bins at a distance of r_i from the pipe centerline; (b) Number of velocity vectors collected at each bin as a function of distance from the wall (modified from Oliveira et al. 2013)

Oliveira et al. (2013) compared the ensemble-averaged velocity profile obtained using the Eulerian approach explained above with direct numerical simulation (DNS) data (Fig 3.3). In Figure 3.3, the red dots represent mean axial velocity (U_z) normalized by the mean centerline of pipe velocity (U_{cp}). The blue dots represent the relative deviation between velocity from 3D-PTV and DNS, expressed as a percentage value of $(U_{3D-PTV} - U_{DNS})/U_{3D-PTV}$. The solid line represents the DNS data. As the figure shows, there is a good agreement between ensemble-averaged velocity data from 3D-PTV and the data from the DNS model.

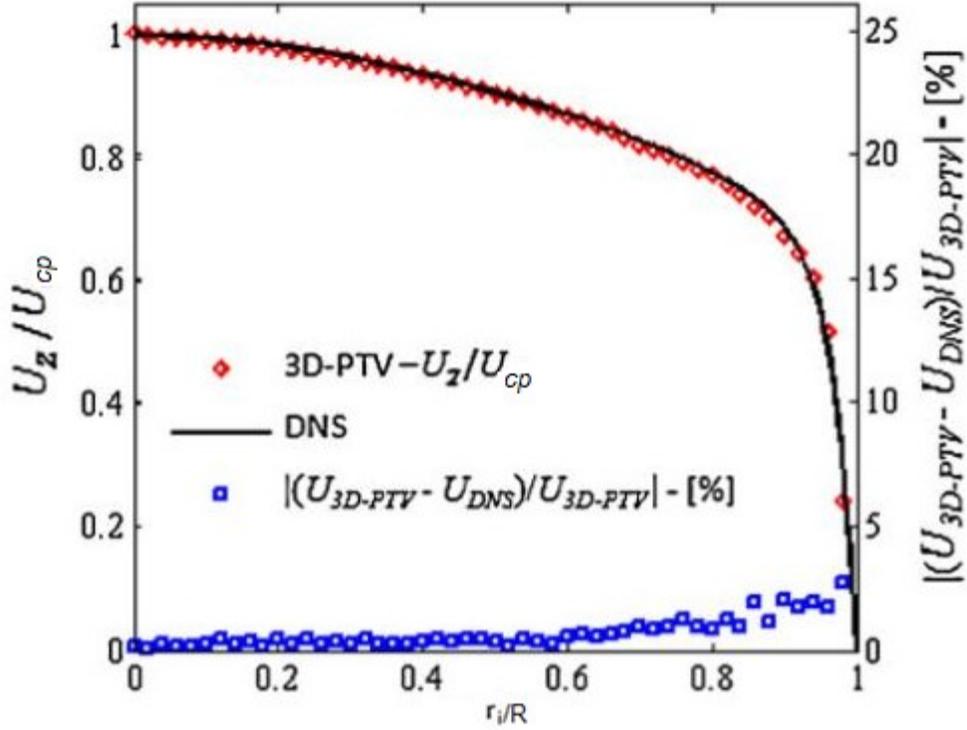


Figure 3.3. Ensemble averaged velocity profile for 3D-PTV and DNS data (from Oliveira et al. 2013)

PTV has been used to study fluid flows in both Lagrangian and Eulerian frameworks. Mulligan et al. (2018) used PTV to compare the time-averaged tangential velocity distribution with the experiments in a free-surface vortex created in a vortex chamber whose geometry took the form of a logarithmic spiral. They used Laser pulses to illuminate the study area and polyethylene microspheres with a mean diameter of 0.8 mm as seeding particles.

In the same vein, Holzner et al. (2008) investigated the small-scale features of turbulent entrainment with the Lagrangian approach using PTV and DNS. By definition, turbulent entrainment is a fluid transition from a non-turbulent to a turbulent state through the boundary between the two regions. Holzner et al. (2008) determined the derivatives of velocity components (e.g., $\partial u/\partial y$) Lagrangian acceleration (e.g., $\partial a/\partial y$) along particle trajectories.

The Laplacian of vorticity, $\nabla^2 \omega$, is obtained indirectly from the local balance equation of vorticity

$$\nabla \times a = \nu \nabla^2 \omega \quad (3.1)$$

where $\nabla \times a$ is the curl of acceleration from the Lagrangian tracking data. The vorticity field and rate of strain are defined in Eq. 3.2 and Eq. 3.3, respectively

$$\boldsymbol{\omega} = \nabla \times \mathbf{V} \quad (3.2)$$

$$S_{xy} = \frac{1}{2} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \quad (3.3)$$

The evolution of enstrophy (the integral of vorticity squared in the study domain) and rate of strain for particles in turbulent entrainment showed that enstrophy is initially very low and increases sharply in a short time when the interface is crossed. In contrast to enstrophy, strain is already significant in the non-turbulent region and increases more gradually.

Previous studies utilize PTV to study turbulent flows. However, no study has been reported to apply the 3D-PTV technique to investigate particle trajectories in a shallow mixing layer. The study of SML using PTV could explain particle behaviour in laterally shear flows, which are common types of flow in river confluence.

3.2 Experimental setup

Experiments were performed in a $9\text{ m} \times 1.5\text{ m} \times 0.25\text{ m}$ (length \times width \times height) laboratory flume with an inlet and outlet reservoir (Figure 3.4). A pump generates recirculating flow at a flowrate of 36 L/s. Figure 3.5 shows the variable flow drive (VFD) to control the discharge. The best practice is to automatically control the pump by setting the VFD control to auto by pressing the auto button on the electronic key until auto appears on the screen. The VFD has an emergency shut-off that stops the pump.

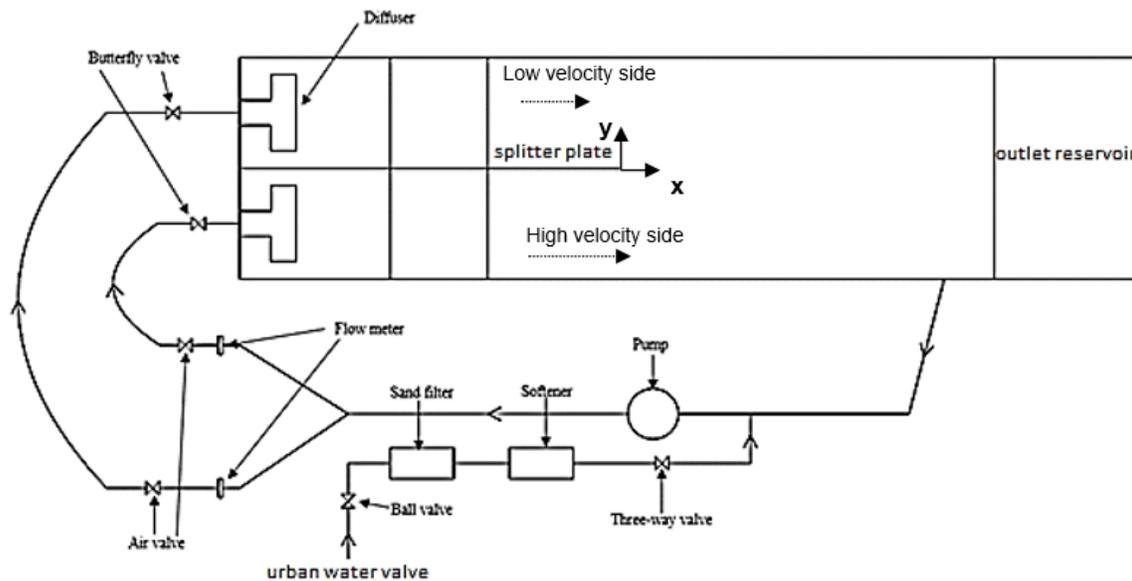


Figure 3.4. Top view of the recirculating flume used for PTV experiments

A blunt-ended splitter plate of 3.5 mm in thickness is installed in the flume channel inlet to create two parallel streams of different velocities.



Figure 3.5. Variable flow drive to control the flow rate in experiments

In order to distribute the flow uniformly over the width of the reservoir, the diffusers inside the inlet reservoir were T-shaped with circular holes uniformly located over the bottom half of their surface area. A uniform flow into the flume was enabled using a transition curve to connect the channel to the inlet reservoir. The previous work of Martinez (2017), using the same flume as this study, verified the effectiveness of the inlet diffuser and the transition curve in producing a uniform flow in the channel. Martinez (2017) showed that the styrofoam sheets on the water surface and honeycombs at the end of the transition curve are sufficient to remove surface waves resulting in a horizontal flow at the end of the transition curve and an almost uniform velocity profile across the flume width.

We used a point gauge to measure the flow depth. One example of a typical point gauge is presented in Figure 3.6. A sand filter and disk filter were used in series (Figure 3.9) to guarantee suitable water quality and lack of external/foreign particles in the flow.

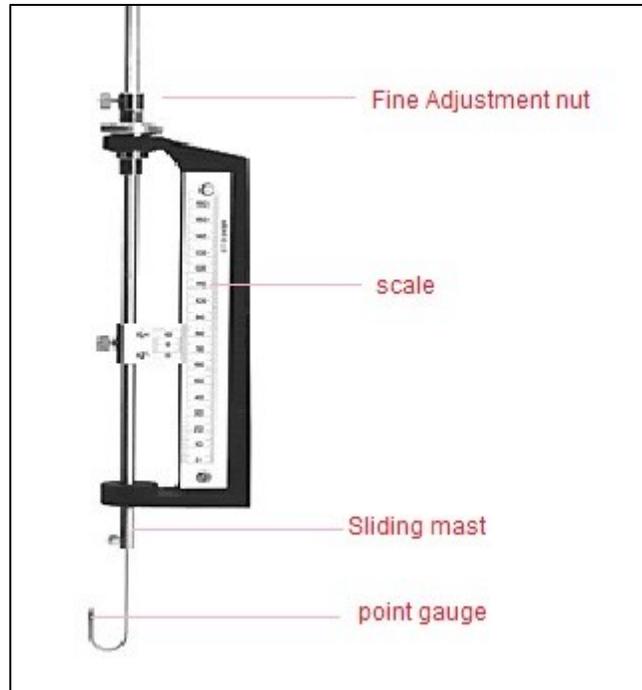


Figure 3.6. Typical point gage for measuring flow depth



Figure 3.7. Filtering apparatus used to remove external particles before recirculating flow into the flume

The experiments were performed for measurements at 0 – 30 cm length range downstream of the splitter plate for three different velocity ratios of $V_r = 1.5$, 1.14, and 1 (as a reference case). The hydraulic conditions of the flow for the experiments are summarized in Table 3.1. The Froude number is a dimensionless number defined as the ratio of inertia to gravity. In this study, the

velocities of two parallel streams were chosen to maintain subcritical flow during the experiments. The Reynolds number, defined as the ratio of inertial forces to viscous forces ($Re = \frac{Ud}{\nu}$), was also maintained to assure a turbulent flow at both sides of the splitter plate. The bottom friction coefficient, c_f , is determined using Eq. 2.5. Flow depth was constant and fixed to 0.065 m for both streams.

Table 3.1. Hydraulic conditions of experiments. Re , Fr and c_f are evaluated at the initial contact location.

V_r	Q (L/s)	Channel stream	U_1 or U_2 (m/s)	Re	Fr	c_f
1.50	14.6	Stream 1	0.300	19500	0.38	0.0115
	21.89	Stream 2	0.450	29250	0.56	0.0106
1.14	18.4	Stream 1	0.377	24505	0.47	0.0110
	20.9	Stream 2	0.430	27950	0.54	0.0107
1	18.1	Stream 1	0.373	24245	0.46	0.0110
	18.1	Stream 2	0.373	24245	0.46	0.0110

3.3 PTV data acquisition

PTV is based on the coordinate measurement and tracking of individual particles, requiring reliable particle identification, multi-image matching, coordinate determination, and tracking each particle in consecutive frames. Figure 3.8 shows a flowchart of the stages for PTV data collection and processing. Each of the steps is explained in the following sections.

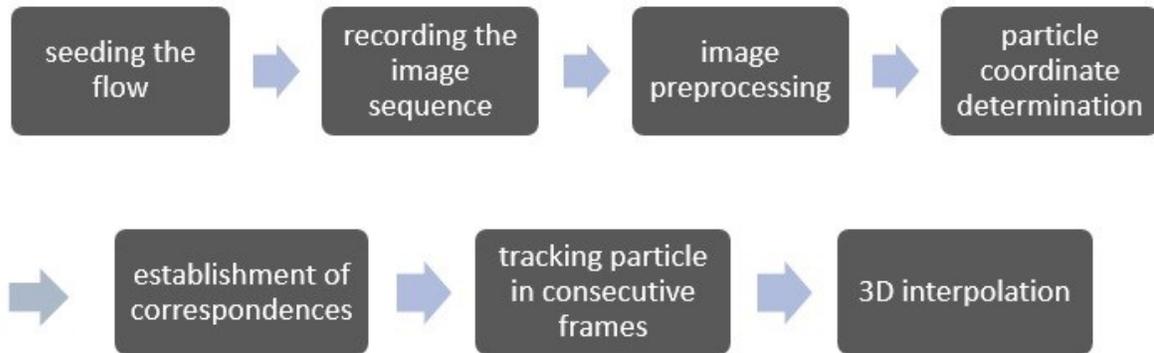


Figure 3.8. Flowchart of sequential stages of data acquisition and processing using 3D PTV

The requirement of particle size for seeding the flow is that the minimum particle size should be equal to the pixel size of the image, with particles sizes typically in the range of 50–100 μm for 3D-PTV experiments (Kim 2015). A study evaluating digital PIV proposed an optimum particle diameter larger than 1.5 pixels and in the range of 1.5 to 3 pixels (Raffel 2007). For the present

study, the pixel size is equal to $0.18\text{ mm}\times 0.18\text{ mm}$ ($180\text{ }\mu\text{m}$). Therefore, the rice bran wax microspheres (Florabeads RBW) with a size range of $250\text{--}600\text{ }\mu\text{m}$ are used as flow markers (Figure 3.11). The particles have a specific gravity of about 1 at 25°C and natural colour that guarantees a good contrast with the black background for image recognition purposes. A sieving process was implemented as Florabeads RBW particles come in a size range of 250 to 600 microns.



Figure 3.9. Rice bran wax microspheres sample used as seeding particle in PTV experiments (from Martinez 2017)

We used three high-speed cameras to acquire the image sequences necessary for 3D-PTV measurements (pco.dimax). At the specified frame rate of 600 Hz, a maximum of 6,298 images are recorded in 10.5 s and stored in the camera's memory. Once the images are recorded, they are locally transferred to the computer operating the camera's software called 'CamWare'. The images stored on the local computer are then transferred to an external drive for analysis using a 3D-PTV image processing tool called 'OpenPTV'.

Camera calibration is a crucial step for 3D-PTV. Using the calibration, the camera's view is correlated to the actual space coordinates being imaged. By calibration, the different camera parameters required for image processing are defined. The quality of the calibration determines the accuracy with which the particle position is observed. Typically, the calibration object is a three-dimensional target-block, in which points with known coordinates are located over the range of x , y and z positions of the imaging volume (Figure 3.10). These point coordinates are then used in the 3D-PTV image processing tool to reconstruct the 3D positions of the detected particles in each recorded frame or in each imaging time step (Kim 2015).

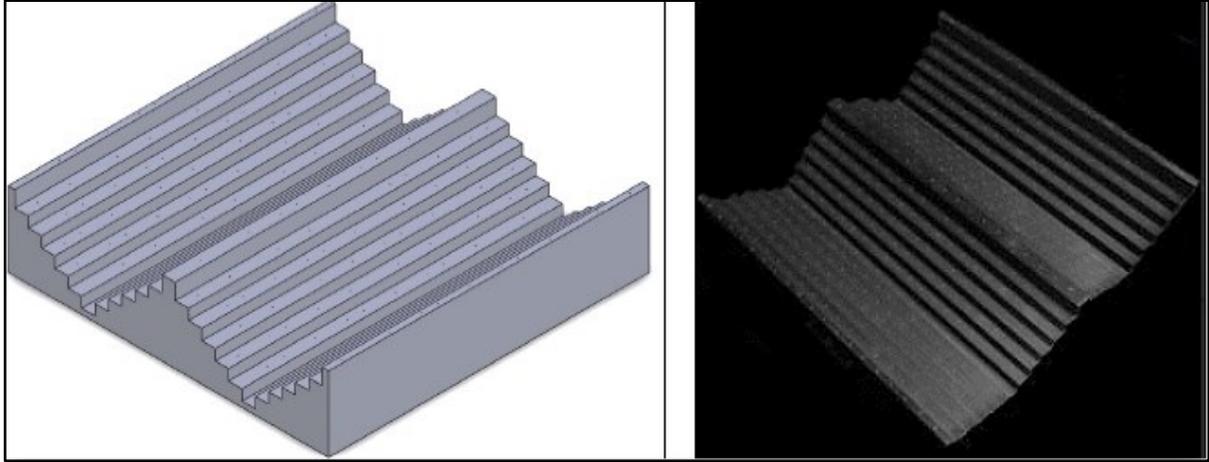


Figure 3.10. Calibration object for 3D-PTV used in this study (left: design; right: machined)

Four sets of halogen lamps (8 light bulbs, 4 of 500W and 4 of 300W) were utilized to illuminate the study area. Two halogen lamps were installed on each side of the flume, perpendicular to the flow direction. Figure 3.11 shows the camera and illumination setup. To ensure that no ambient particles (dust) interfered with the experiments, black tarpaulins were used to cover the top and sides of the flume along its entire length.



Figure 3.11. Camera setup (left) and illumination setup (right) for PTV experiments

In the next step, the images were processed to extract the velocity data. This stage is called ‘image preprocessing’. Measurements of particle positions are affected by noise caused by external factors affecting the image quality. The lower the spatial resolution of the measurement in terms of particle spacing, the more the position signal is affected by noise, and the higher is the required

over-sampling in time (Holzner et al. 2008). In image preprocessing, a high pass filter was applied to all images.

A high pass filter is a technique in image processing mainly used for sharpening the image. The high pass filter only allows high frequencies to pass through. Each pixel in an image has a value in terms of color. High frequencies in images refer to pixel values that are changing dramatically. For PTV experiments that the images are recorded in black and white, the high pass filter sharpens the particles as white dots on a black background.

3.3.1 Image processing

PTV image processing consists of the last four steps of the flowchart shown in Figure 3.8. In this study, we used Open PTV to perform image processing. OpenPTV is a 3D-PTV software based on the core algorithms developed at Swiss Federal Institute of Technology in Zürich. The branches of the Zurich code have been developed independently by the Eindhoven University of Technology (TU/e) group of Turbulence and Vortex Dynamics and by the Turbulence Structure Laboratory at Tel Aviv University (OpenPTV). Figure 3.14 shows the OpenPTV user interface.

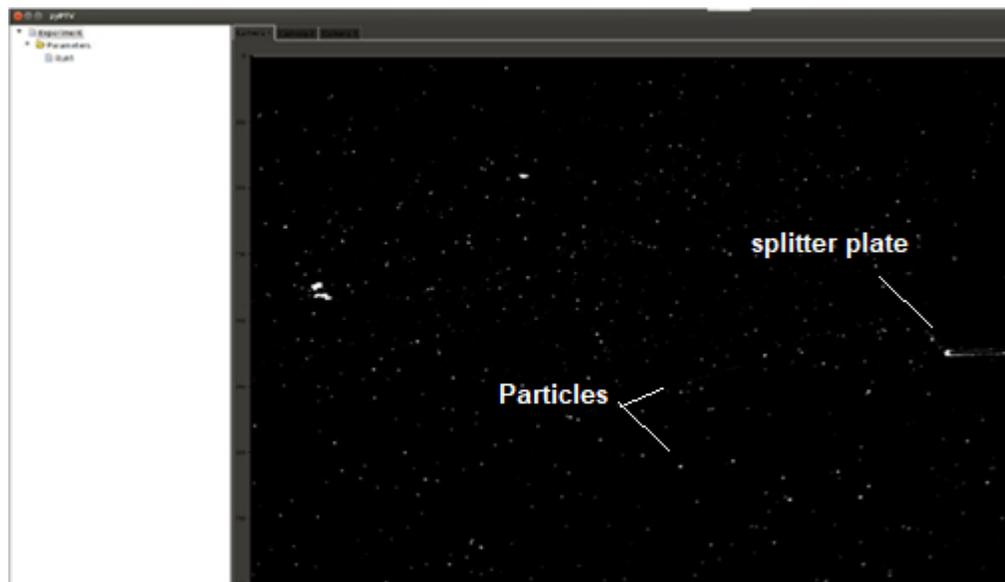


Figure 3.12. Example of PTV images uploaded in OpenPTV environment

Before uploading the images into OpenPTV, we need to prepare them in an acceptable format for the code. OpenPTV needs four folders for each experiment to process the images properly. The name of the folders must be selected as 'cal', 'img', 'parameters', and 'res'.

The calibration images of each camera in format .tif must be in a folder named 'cal'. The calibration images must have a high resolution as well as high quality. Therefore, the unwanted stains and spots must be removed. In addition, the unwanted area around the calibration object should be clipped.

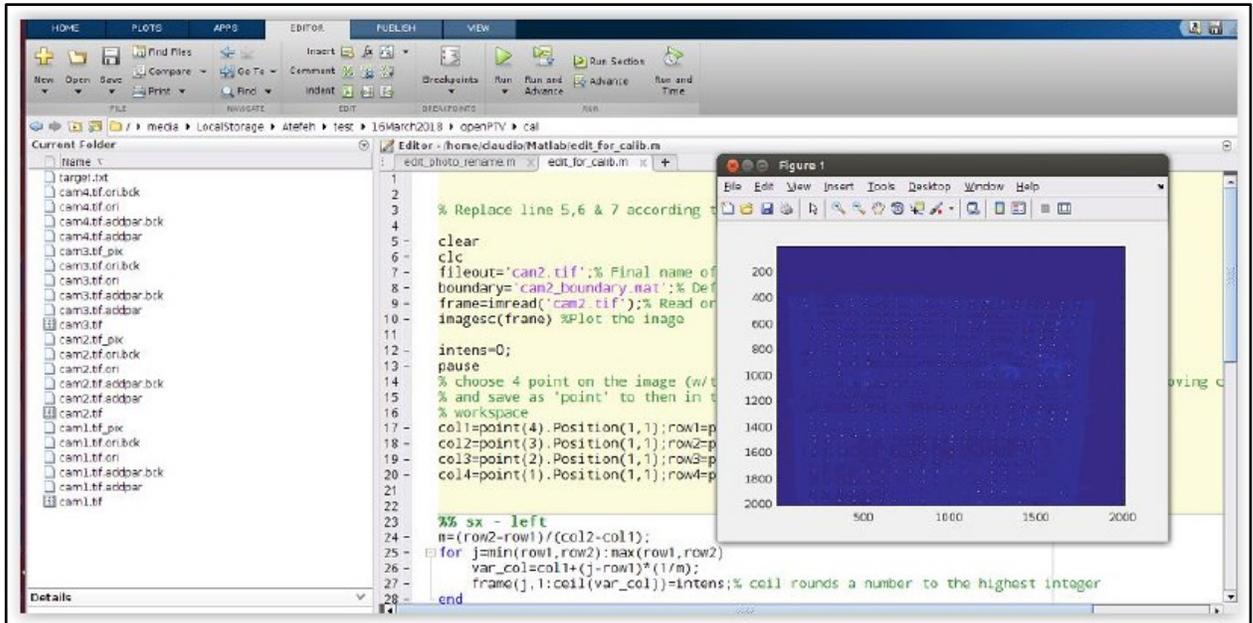


Figure 3.13. Preparation of calibration images using a MATLAB script

The camera orientation files, and parameter files must be located in the folder *cal* in which the processing parameters are defined. Next, the images are renamed and stored in the acceptable format as 'cam10000.tif'.

The parameters directory contains the following:

- cal_ori.par - calibration plate, images, orientation files
- criteria.par - object volume and correspondence
- detect_plate.par - parameters for control point detection
- man_ori.par - point number for manual pre-orientation
- orient.par - flag for camera parameter usage
- pft_version - set flag for peak fitting version
- ptv.par - main parameter file
- sequence.par - sequence parameters
- sortgrid.par - set distance between image coordinate and projected control point in pixel
- targ_rec.par - parameters for particle detection
- track.par - tracking parameters

The parameters mentioned above must be adjusted by the user based on the calibration process. Figure 3.14 demonstrates these parameters that are defined below row by row.

1. First row: location of the camera sensor respect to the calibration target origin (0, 0, 0)

2. Second row: the angles (in radians), the first is around the x -axis, then y -axis, and the third is the angle of rotation around z -direction which coincides with the imaging axis of the camera (the line that connects the sensor and the target)

3. The next three rows are the rotation matrix

4. The next two parameters are the x_p and y_p positions of the pinhole in respect to the image centre in millimetres, e.g., if the camera imaging axis is at 90° to the sensor, then $x_p = y_p = 0.05$ mm. Next parameter is the back-focal distance, typically called F . For example, if we have a ratio of world image to chip image of 500 mm to 65 mm (384 pixels corresponds to 17 microns), i.e., 1:8. The distance from the lens to the calibration target is about 800 mm, so the focal distance is about 100 mm.

6. The last row with the three parameters is the position of the glass (bottom of the flume) with respect to the origin and in the coordinate system of the calibration target. If the glass bottom is perpendicular to the imaging axis and parallel to the calibration target, and the distance in water is about 100 mm, the last row should be written as 0.0 0.0 100.0. Since division by zero is not recommended, the manual suggests using a tiny deviation from 0.0, for example, 0.0001.

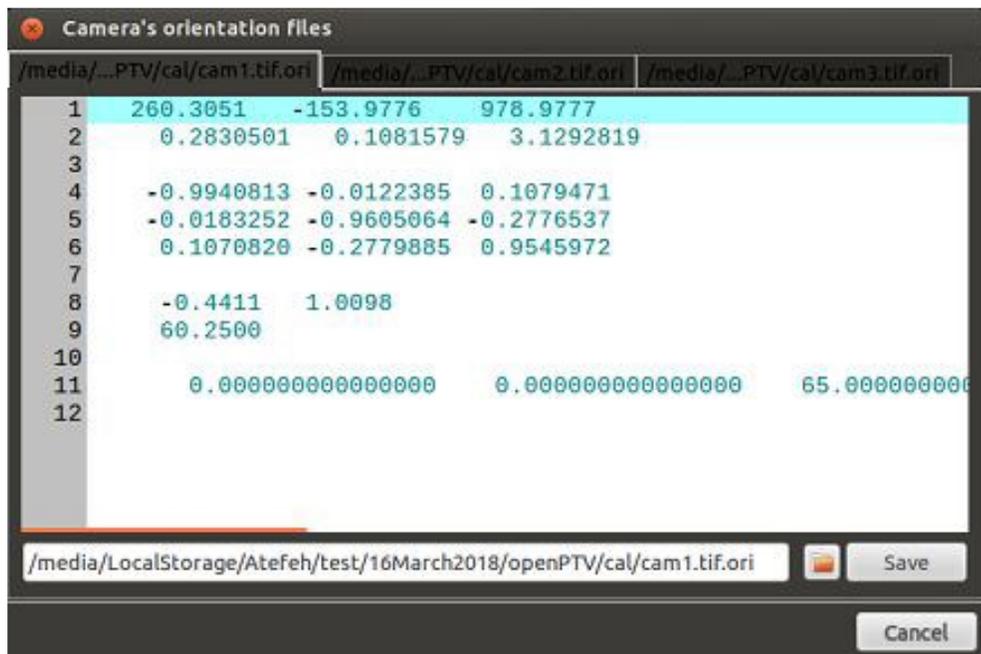


Figure 3.14. Parameter setting for calibration for OpenPTV

To be able to set the correct parameters, knowledge of high-speed camera image recording is necessary. Maas et al. (1993) discussed hardware components for 3D-PTV systems and proposed a mathematical model of photogrammetric 3D coordinate determination, taking into account the different refractive indices in the optical path. The fundamental mathematical theorem of photogrammetric 3D coordinate determination is the co-linearity condition, which states the object point, camera projective centre, and image point lie on a straight line (Figure 3.15).

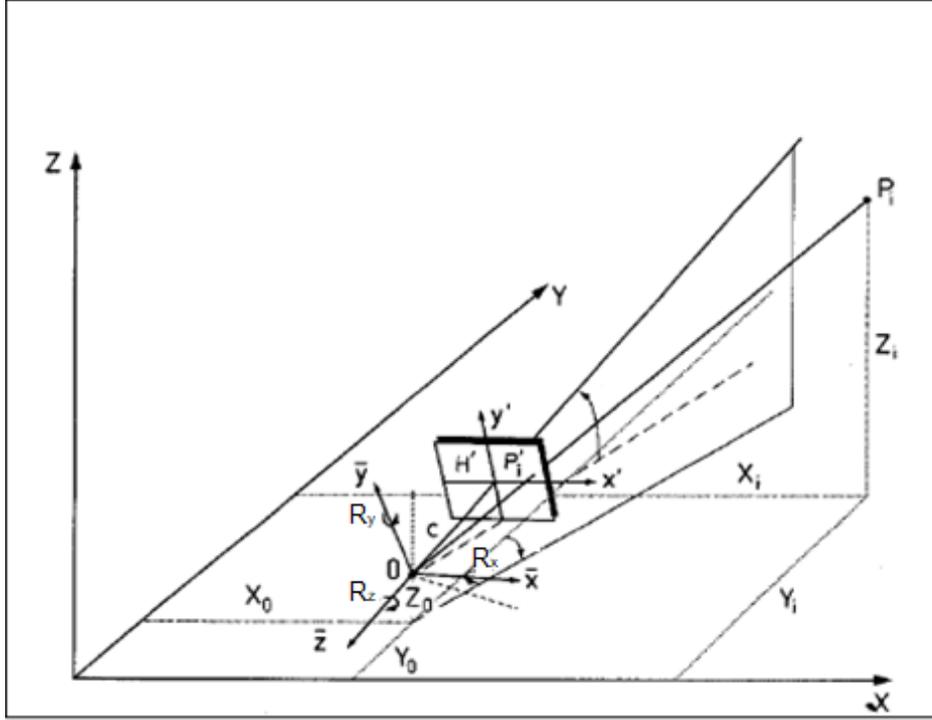


Figure 3.15. Co-linearity condition for image recording (Maas et al. 1993)

The collinearity condition results in a matrix equation as follows:

$$\begin{pmatrix} x'_i - x_h \\ y'_i - y_h \\ -c \end{pmatrix} = \lambda_i \cdot R_{PTV} \begin{pmatrix} X_i - X_0 \\ Y_i - Y_0 \\ Z_i - Z_0 \end{pmatrix} \quad (3.4)$$

In Eq. 3.4, R_{PTV} is defined as a 3×3 rotation matrix with angles R_x, R_y, R_z and λ_i as the scaling factor; X_0, Y_0, Z_0 are the camera projective center coordinates; R_{PTV} forms the camera exterior orientation; (x'_i, y'_i) are projection of (x_i, y_i) , (X_i, Y_i, Z_i) are object point coordinates; (x_h, y_h) is principle coordinates; c is the principal distance.

Maas et al. (1993) extended Eq. 3.4 to consider the broken optical beam due to passing the three media water, glass (flume bottom) and air (between the camera lens and the flume bottom) with different refractive indices from particle to the camera sensor in addition to the effect of lens distortion and effects of the digitization on the image coordinates. In OpenPTV, the solution presented by Maas et al. (1993) is used to determine the coordinates of particles in the image.

Once the orientation parameters are set, all the dots on the calibration object are recognized by Open PTV (Figure 3.16). The next step is the pre-processing of the images.

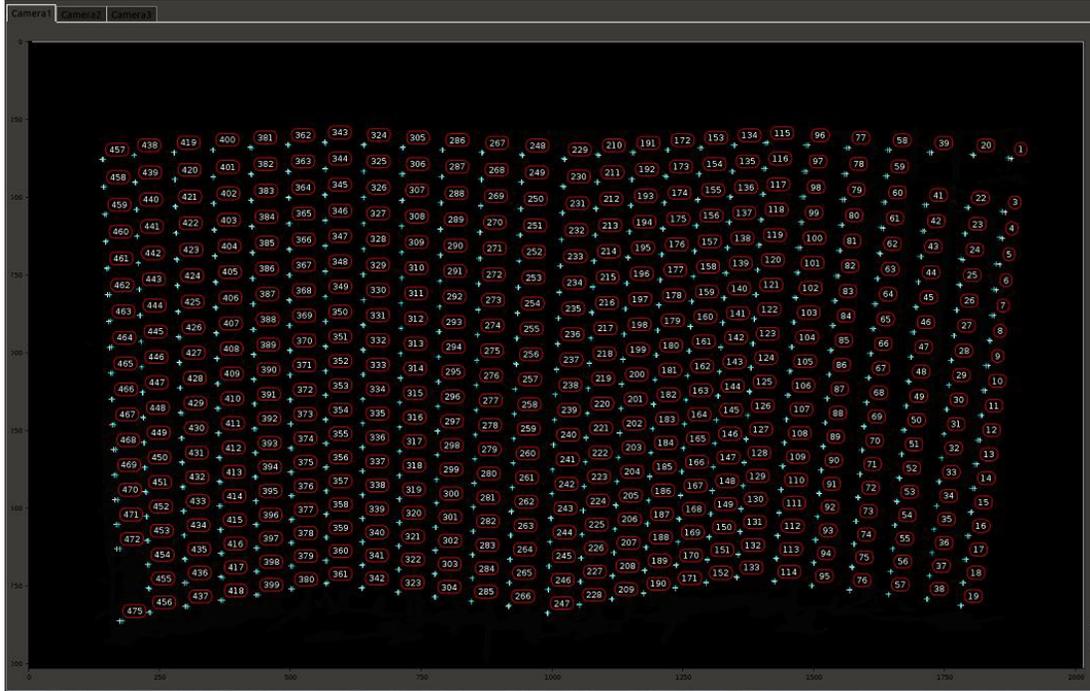


Figure 3.16. The final stage of calibration, dots on the calibration object are all recognized by OpenPTV

Digital images processing includes essential tasks of particle identification and particle image coordinate determination, as the particles appear as bright stains on the dark background of each image. The first step of pre-processing is to remove non-uniformities of the background intensity level due to reflections by using a low-pass filter, which is performed in the spatial domain by subtracting an unsharp mask from the original image.

Using *Image coord*, which performs the detection algorithm, the coordinates of particles are determined in each of the three cameras. The option *Correspondences* runs the epipolar matching. Particles that were matched across all three cameras are recognized. In the next step, *3D positions* and then *sequence without display* are performed to track the recognized particles in the image sequence.

After performing the steps mentioned above, the results folder contains the output files, including *par_is* files used for post-processing.

3.3.2 Dataset analysis

The post-processing stage involves tracking particles frame by frame to produce trajectories. Various tracking algorithms have been developed by researchers such as Malik et al. (1993).

Malik et al. (1993) presented a tracking scheme to determine a sequence of velocity vectors within a three-dimensional observation volume of a fluid flow and reconstructed the long-time particle trajectories. In this study, the study volume was $0.3 \times 0.3 \times 0.065 \text{ m}^3$. They used information

from four consecutive frames to establish links between particles in the middle two of the four frames. Based on Malik et al. (1993), when one of the links in the current frame [f] coincides with the continuation of a track that had already been established at frame [$f-1$], then the track which contains this link is favoured as the true track.

The three velocity components of a particle are obtained by identifying and tracking the position of the particles as they move through the observation volume Eq.3.5

$$\mathbf{V} = \lim_{\Delta t \rightarrow 0} \frac{\Delta S}{\Delta t} \quad (3.5)$$

where ΔS is the displacement determined by the movement of a seeded particle in time Δt .

A C++ script developed by Luthi et al. (2007) based on Malik et al. (1993) was utilized in this study. The script creates *xuag*, *xuap* and *trajpoint* files. One can edit the input file to configure parameters. The output, *xuap* files contain particle ID in the previous frame in the image sequence, particle ID in the current frame, x , y , z coordinates, high pass filtered velocities, accelerations in x , y , z directions, and a successful indication value which takes zero or one.

To obtain the ensemble average flow characteristics, we used an approach similar to the one used by Oliveira et al. (2013). In this method, PTV particle velocity data should be organized in a grid for Eulerian analysis. To achieve this goal, we developed a Python script to define a virtual grid with a $0.001 \times 0.001 \times 0.001 \text{ m}^3$ bin size and iterate to each frame to capture instances in a specific node of the grid. The code provides the node ID that the user can call. The user can perform the statistical analysis on each node and visualise the results. Due to a large number of nodes, especially in a finer mesh, it is essential to develop an automatic method to detect and remove outliers from data sets.

3.3.3 Methods for multivariate outlier detection

In studying SML, considering both streamwise and transverse velocity components is essential. This section explains why in SML studies, multivariate outlier detection is necessary, and the methods to perform multivariate outlier detection are investigated.

The coordinates of particles and their velocity obtained from post-processing contain outliers. Removing spikes and outliers from Eulerian data sets like velocity time series collected by ADV (Goring and Nikora, 2002, Thorne and Hanes, 2002) and particle imaging velocimetry (PIV) (Westerweel and Scarano, 2005) has been investigated before. However, no velocity outlier detection study for the PTV dataset is reported. PTV and PIV methods have in common that they determine the velocity from the displacement of particles in a moving fluid during a prescribed time interval. In contrast to PIV, in which the mean displacement of a small group of particles is sought, PTV tracks individual particles' trajectories in three-dimensional space (Virant and Dracos, 1997).

Previous studies regarding removing outliers from the PIV dataset mainly focused on the interpolation techniques with neighbouring data. Westerweel and Scarano (2005) presented an algorithm to identify spurious data from PIV based on a threshold value of the normalized residual fluctuation of the velocity of one data point. However, this method cannot be applied to PTV data set because of two reasons: first, identifying neighbours for PTV data points that are not gridded

like PIV cannot be done, and the second, PTV data points are not equally distributed in space, so they do not have equal influence in determining the viability of a vector in question.

Duncan et al. (2010) modified the Westerweel and Scarano algorithm to be used for PTV data as well. They tackled the first issue using Delaunay tessellation, allowing a convenient method of defining neighbours as the data points that share triangles. For the second problem, they considered the distance to the point in question as weights of data points in the neighbour.

Past studies of particle tracking velocimetry have been devoted to increasing trajectory yield. On the other hand, in studying shallow mixing flows with lateral shear, both streamwise and spanwise velocity datasets play an essential role in flow characteristics. While single variate outlier detection methods are simple to use, applying multivariate filters that remove outliers from velocity vector components to study shallow mixing flow is essential.

At the first step of multivariate outlier detection, we consider two velocity components (u, v) of each velocity observation. On a u - v plane, u on the horizontal axis, and v on the vertical axis, we can assume one observation is an outlier if it is extreme in every coordinate of the u - v plane (Figure 3.17). On the other hand, not every outlier has an extreme velocity limit. In this figure, d_1 and d_2 are the univariate threshold of an outlier for u and v , respectively. They can be defined as twice the standard deviation of data. As demonstrated in Figure 3.17, point A is a univariate outlier for u and v individually. However, this observation is inside the ellipse where most observations are (cloud of observations); therefore, point A is not considered a multivariate outlier. On the other hand, observation B is not a univariate outlier for both u and v data sets, but it is considered a multivariate outlier since it is separated from the cloud of observations.

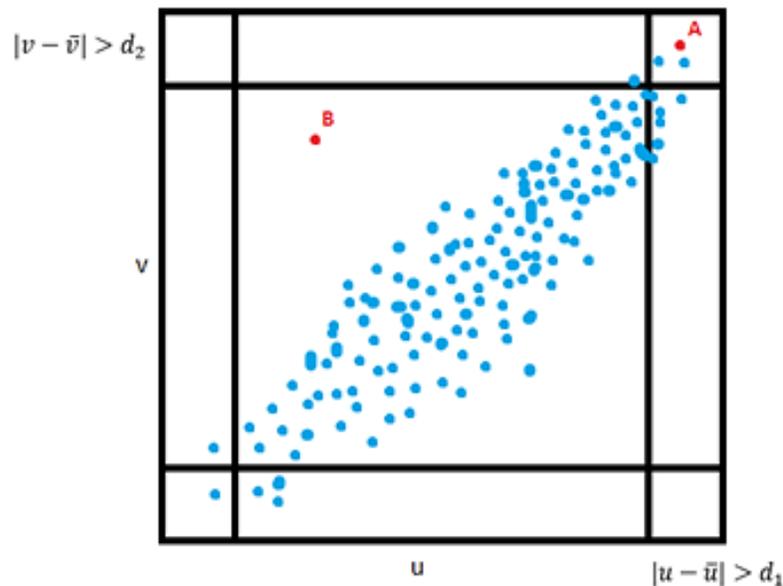


Figure 3.17. Schematic definition of the multivariate outlier on the u - v plane. Point A is not defined as a multivariate outlier, whereas point B is defined as a multivariate outlier

There are many types of outlier detection methods, and researchers, according to different criteria, have divided them into many categories. Choosing the appropriate outlier detection method is highly domain-oriented (depending on the applications). Nazari and Kang (2018) evaluated multivariate outlier detection methods with medical data. Chen et al. (2010) compared several algorithms of outlier detection for traffic datasets in transportation engineering.

Classifying methods that categorize the observations into two classes as “regular” and “outlier” are widely used as outlier detection methods such as support vector machine (SVM) algorithms. While most classifying-based and statistical-based methods construct a profile of regular instances and then identify instances that do not conform to the regular profile, some methods explicitly isolate outliers, such as IsolationForest. In this study, the SVM and IsolationForest algorithms were selected to be tested on PTV data sets. A description of each method is presented in the following.

The SVM algorithm computes a binary function capturing regions in the data space where the probability density lives (Scholkopf and Smola 2005). The algorithm maps the data into a feature space using an appropriate kernel function and then finds the hyperplane that separates classes of data with maximum margin. A simple, widely used kernel is the linear kernel of features of data. In cases that data is not linearly separable, a kernel function is used.

Figure 3.18 illustrates the fundamentals of the SVM method. According to this figure, several classifying hyperplanes can be proposed. A hyperplane in an n -dimensional Euclidean space is a $(n-1)$ dimensional subset of that space that divides the space into two parts, e.g., for a two-dimensional plane and the hyperplane is a line. The algorithm first finds the closest points to the hyperplane to find the optimal classifier line or hyperplane. Those points are called support vectors. Then the distance between the hyperplane and the points is calculated. The goal is to maximize this margin, and the optimal hyperplane is a line for which the margin is maximum. The margin, M is defined as $M = (X_1 - X_2) \cdot \frac{N}{|N|}$, where X_1 and X_2 are the support vectors, and N is the normal vector to the hyperplane.

Therefore, maximizing M is equivalent to minimizing the norm of vector N ($|N|$) which for simplicity of calculations; we consider minimizing the $\frac{1}{2}|N|^2$. The SVM decision is based on the following three criteria:

1. $\min \frac{1}{2}|N|^2$
2. $p^i \cdot |N| \geq 1$ For the label of class +1
3. $p^i \cdot |N| \leq -1$ For the label of class -1

where p^i is the projection of X_1 onto the vector N . Instances are classified into two classes, with +1 and -1 as labels of the class. As illustrated in Figure 3.18, the hyperplane in panel a is not the optimum hyperplane since the values of p^i is small, therefore for SVM decision criteria to be satisfied, the value of $|N|$ must be large, which does not satisfy the optimization function (condition 1). The optimized hyperplane is proposed in Figure 3.18b.

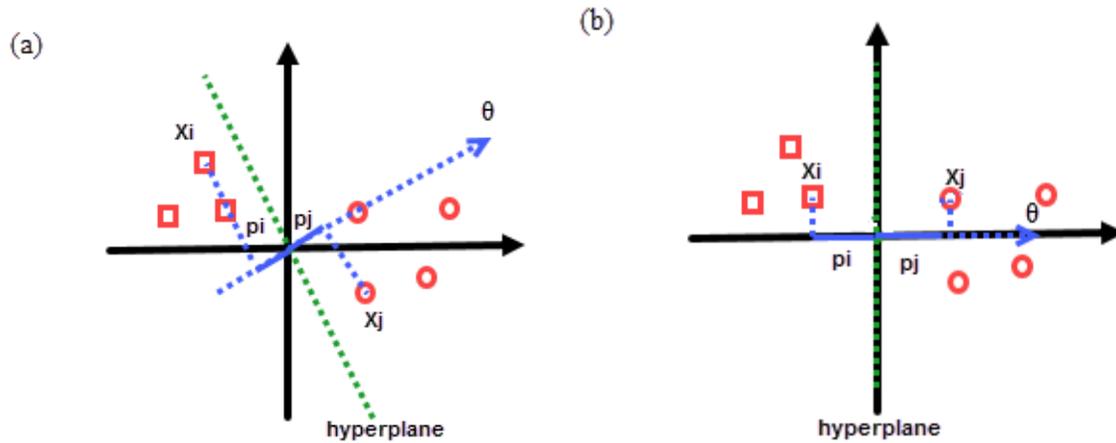


Figure 3.18. Illustration of SVM method for classifying data: (a) non-optimal hyperplane; (b) optimal hyperplane

Another algorithm that was tested in this study was the IsolationForest. The IsolationForest consists of a series of isolation trees (itree) that randomly partition instances and repeat this until all the instances are isolated (i.e., recognized by the itree). Random partitioning produces shorter paths for outliers since the examples with extreme attributes/values are more likely to be separated in early partitioning. Therefore, when a “forest” of random trees collectively produces a shorter path length for some particular points, they are most likely to be outliers. Considering the non-outlier data as ‘regular’ data, Figure3.19 shows the outlier, n_o , detected in shorter trees paths, while regular data, n_i takes more branching to be detected.

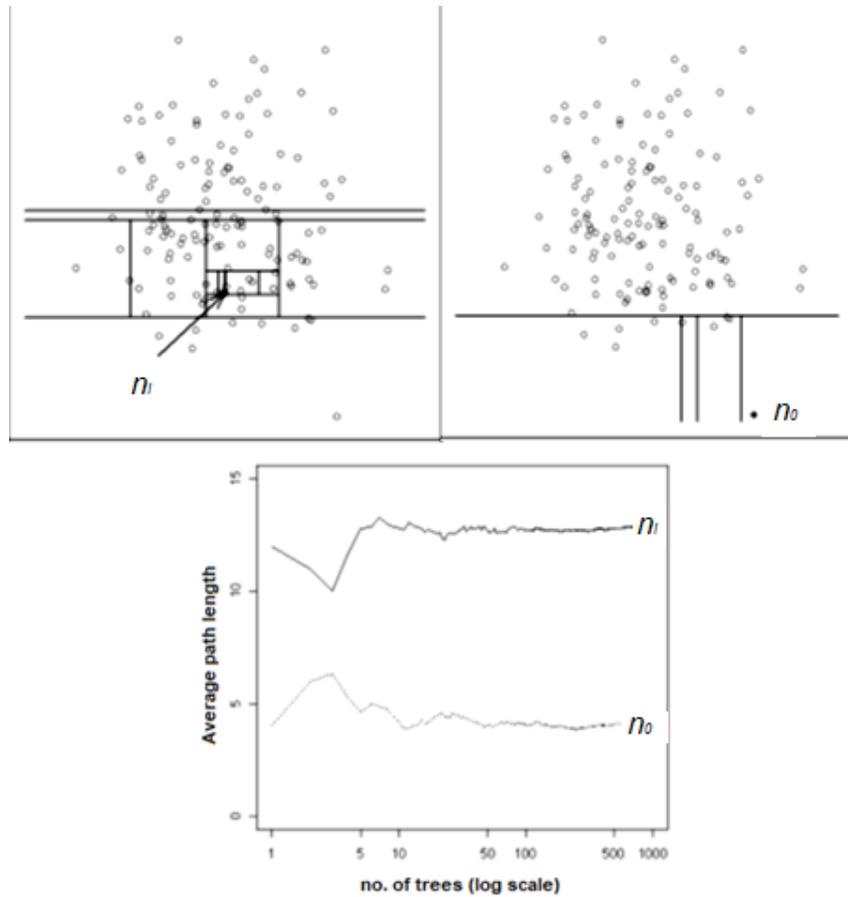


Figure 3. 19. IsolationForest illustration to detect outlier. Outliers are more susceptible to isolation and hence have short path lengths: (a) data recognized by tree; (b) outlier detected by tree; (c) comparison of the average path length of data and outlier (Liu et al. 2008)

The performances of SVM and IsolationForest methods are shown in the scatter plot of $u-v$ in Figures 3.20 and 3.21, respectively. The presence of outliers in each sample is evident with a dispersion of data (isolated from the cloud of data points) and extremely low-velocity values recorded. Four samples are selected in the high-velocity stream far from the splitter plate (Figure 3.4) and have a similar distance to the flume bottom to ensure the physical processes that control velocity data in the samples are similar. The results show that SVM tends to detect some of the instances in the cloud of data in most samples. In the PTV velocity dataset, it appears that most of the outliers are recorded as very low values of u and v , and therefore, the cloud of data is tilted toward the high-value side. In addition, there is no significant difference between the data point distribution density of regular data and outliers in the dataset. This can affect the performance of SVM in terms of creating a non-symmetrical hyperplane around regular data points. On the other hand, IsolationForest utilizes no distance or density measures to detect outliers; therefore, it does not suffer from this issue (dependent on the density of data points) and performs more acceptable on PTV data.

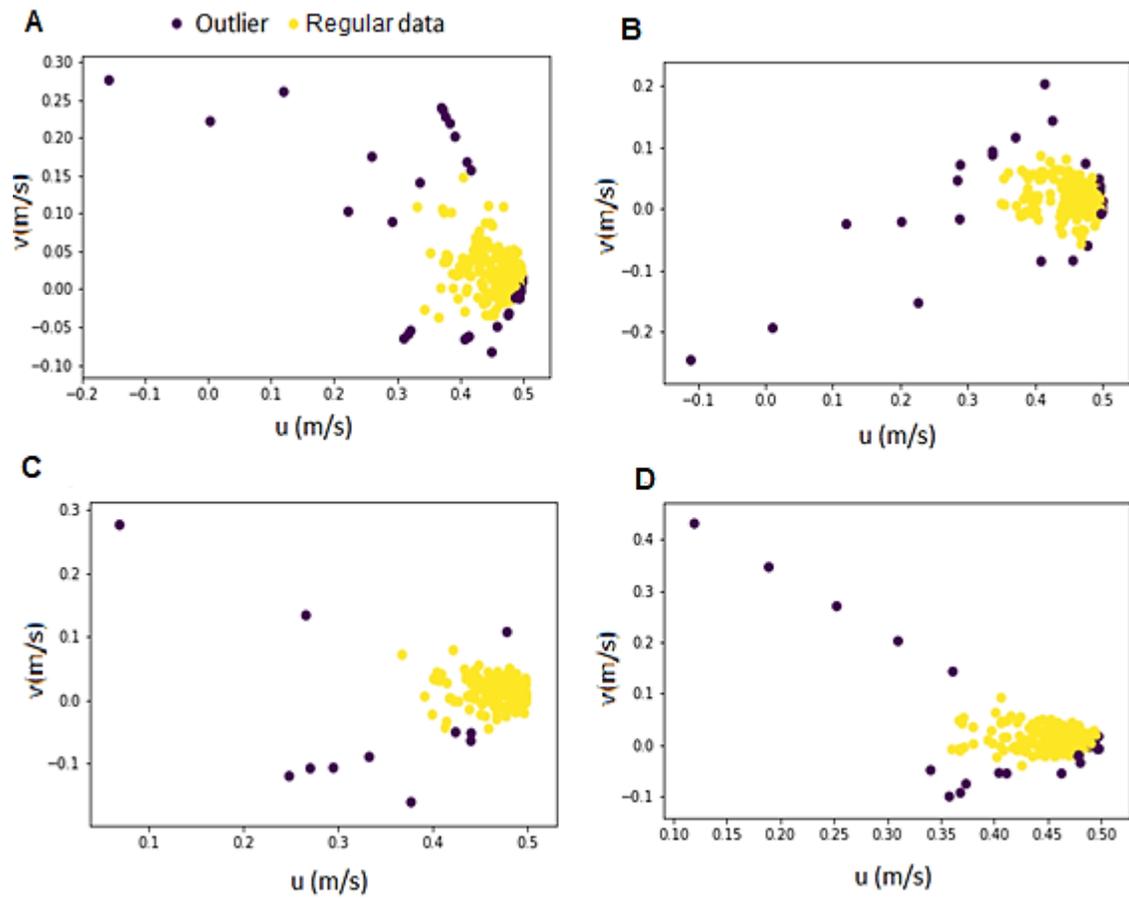


Figure 3.20. u - v scatter plot of four samples filtered by SVM method. Yellow dots are the regular, and purple dots are the outliers. (A) sample A, (B) sample B, (C) sample C, and (D) sample D

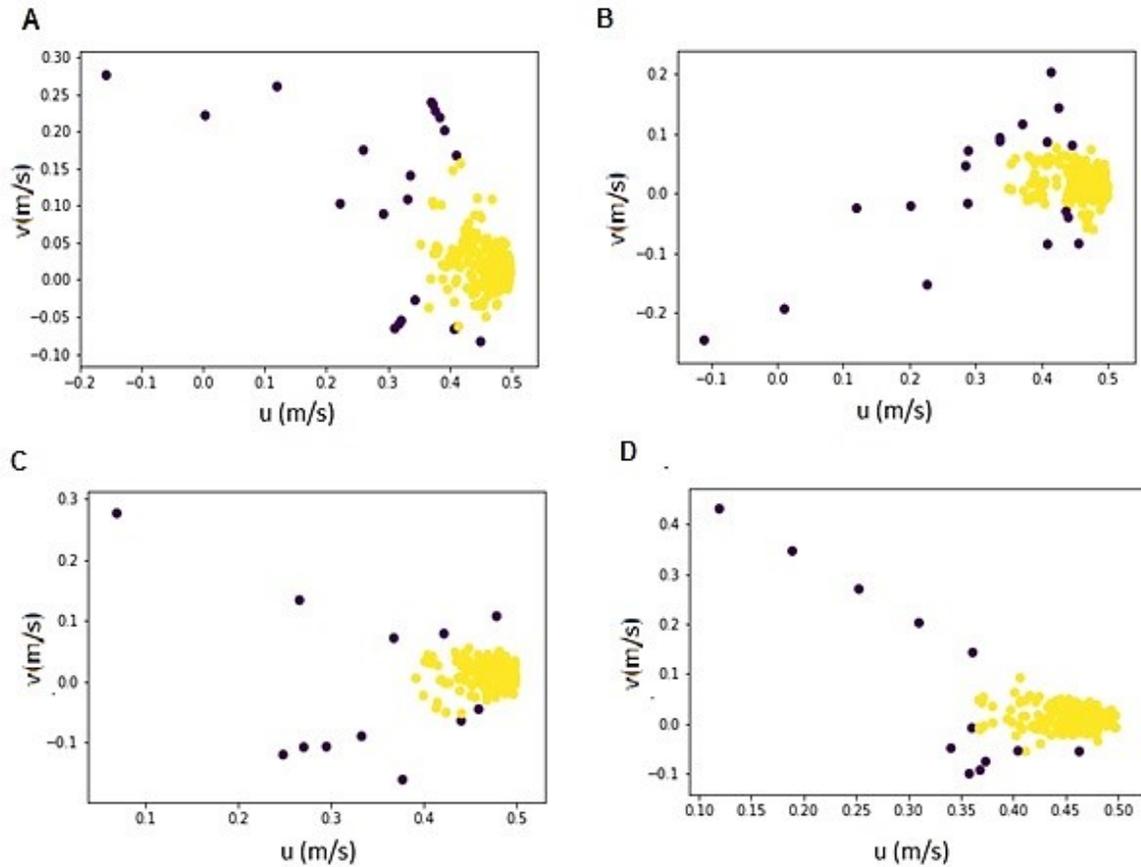


Figure 3.21. u - v scatter plot of four samples filtered by IsolationForest method, (A) sample A, (B) sample B, (C) sample C, and (D) sample D

The box plot of streamwise velocity data shown in Figure 3.23 evaluates how the outlier detection methods affect data distribution. The whiskers indicate the minimum and maximum of data, and the first and the third quartile form the sides of the box. A schematic box plot is shown in Figure 3.22.

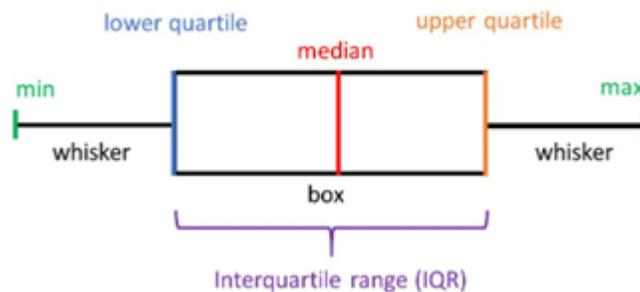


Figure 3.22. Schematic of a box plot components

Figure 3.23 shows the comparison between raw data statistics (before outlier detection) and the data after applying SVM and IsolationForest filters to detect outliers. In most cases, the filtering methods have not changed the quartiles and median significantly. However, they removed the outliers specially located on the lower whisker properly. Comparing to SVM, IsolationForest performs slightly better for samples *A* and *C*. The efficiency of SVM and IsolationForest is similar for samples *B* and *D*.

A comparison between dataset statistics before and after filtering is made in Table 3.2. The data variance without filtering outliers is significant and almost one order of magnitude larger than filtered data. RandomForest reduced the variance of data more than SVM in most of the samples for both streamwise and spanwise velocity. In conclusion, the RandomForest method is a better choice for filtering outliers from PTV datasets.

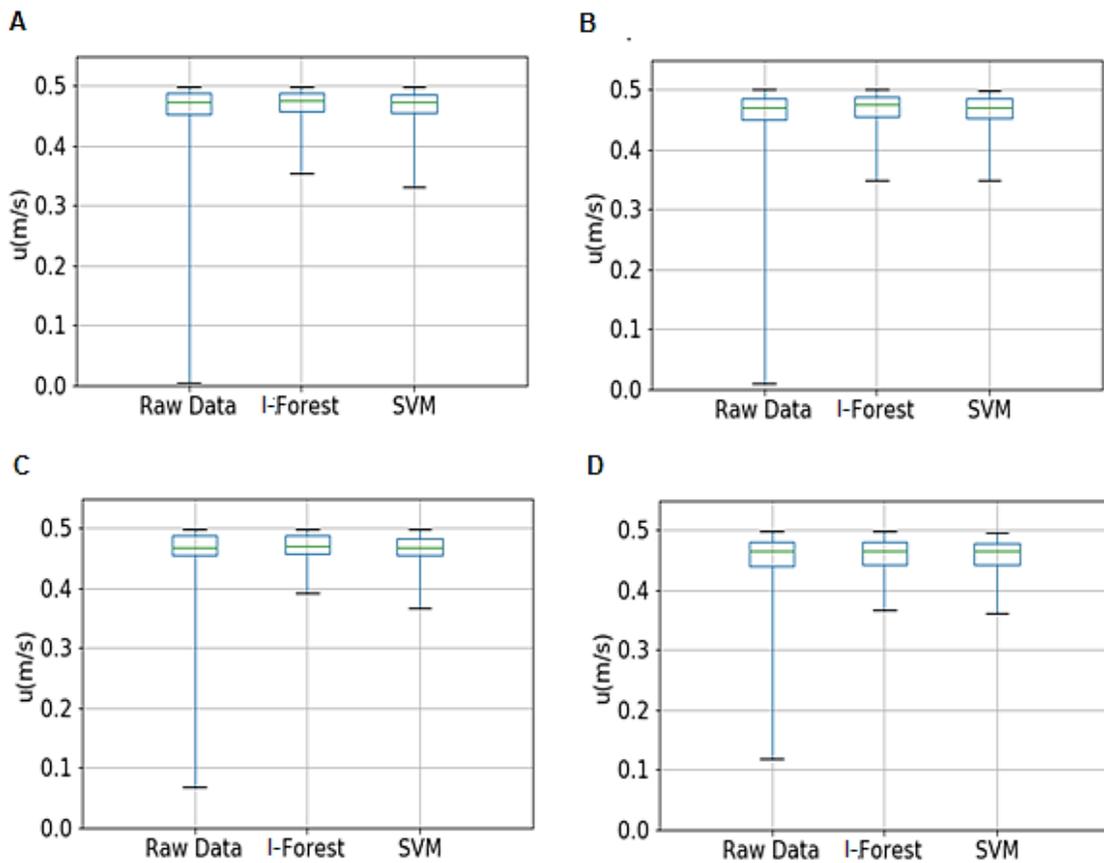


Figure 3.23. Data distribution for data before (raw data) and after filtering methods (Isolationforest and SVM) for four samples: (A) sample A; (B) sample B; (C) sample C; (D) sample D

Table 3.2. The variance of streamwise (u) and spanwise velocity (v) for raw data (without filtering) and two studied filtering methods

		Raw data	IsolationForest	SVM
σ_u (m/s) ²	Sample A	0.0032	0.00085	0.00093
	Sample B	0.0033	0.00096	0.00096
	Sample C	0.00207	0.00056	0.00059
	Sample D	0.0021	0.00087	0.00088
σ_v (m/s) ²	Sample A	0.0018	0.0006	0.00054
	Sample B	0.0011	0.00043	0.00045
	Sample C	0.0012	0.00033	0.00041
	Sample D	0.0024	0.00042	0.00040

PTV requires intensive efforts from the early steps of image acquisition to data refining before producing datasets used for further analysis. In the next section, the refined and cleared PTV datasets are analyzed to investigate mixing flow characteristics.

3.4 Ensemble average flow velocity

The ensemble average streamwise velocity (\bar{u}) profiles in the transverse direction for $V_r = 1.14$ and $V_r = 1.5$ have been presented in Figure 3.24 and Figure 3.25. The profiles are considered at three streamwise locations from the splitter plate ($x = 0.01, 0.1,$ and 0.25 m) and a vertical distance of $z = 3$ cm from the flume bed. The ensemble average considers particles passing through a cube of $0.001 \times 0.001 \times 0.001$ m³. The selection of the cube dimensions should allow a sufficient number of particles, but at the same time, it should not be too large to avoid smearing out velocity gradient.

In Figures 3.24 and 3.25, $y/d = 0$ represents the initial interface location of two streams. The presence of a mixing layer is evident with a significant velocity gradient around $y/d = 0$. In Figure 3.24, the wake effect caused by the splitter plate on the velocity distribution is visible at $x = 0.01$ m. This effect gradually disappears at $x = 0.25$ m. The development of the boundary layers growing on the splitter plate walls and the wake effect cause a velocity deficit in the transverse velocity profiles. In this study, the velocity deficit is defined as the velocity values smaller than the velocity of the slow side of the shallow mixing flow.

When the velocity ratio between two streams decreases to 1.14 (Figure 3.25), the velocity deficit persists for a longer streamwise distance compared to that of the $V_r = 1.5$. One explanation would be the larger value of $\frac{\partial u}{\partial y}$ in the latter case that dominates the velocity deficit in spatially short distances. The velocity deficit modifies the velocity profile in the transverse direction such that it does not follow a hyperbolic tangent function as suggested by van Prooijan and Uijttewaal (2002).

The wake effect and the development of boundary layers on the splitter plate walls cause the traditional relations of mixing layer width determination unpractical. The velocity profile at $x = 0.01$ m shows that the boundary layer growing on the splitter plate walls and the wake effect can overpower the transverse shear caused by the velocity gradient. Therefore, a new method to determine mixing layer width is proposed based on boundary layer definition. Based on this definition, the mixing layer edge at the low-velocity stream is located where the velocity is equal to $\bar{u} = 0.99U_1$ at the high-velocity stream, the mixing layer edge is located where the velocity is equal to $\bar{u} = 0.99U_2$. In Figs 3.24 and 3.25, $\bar{U} = \frac{U_1+U_2}{2}$.

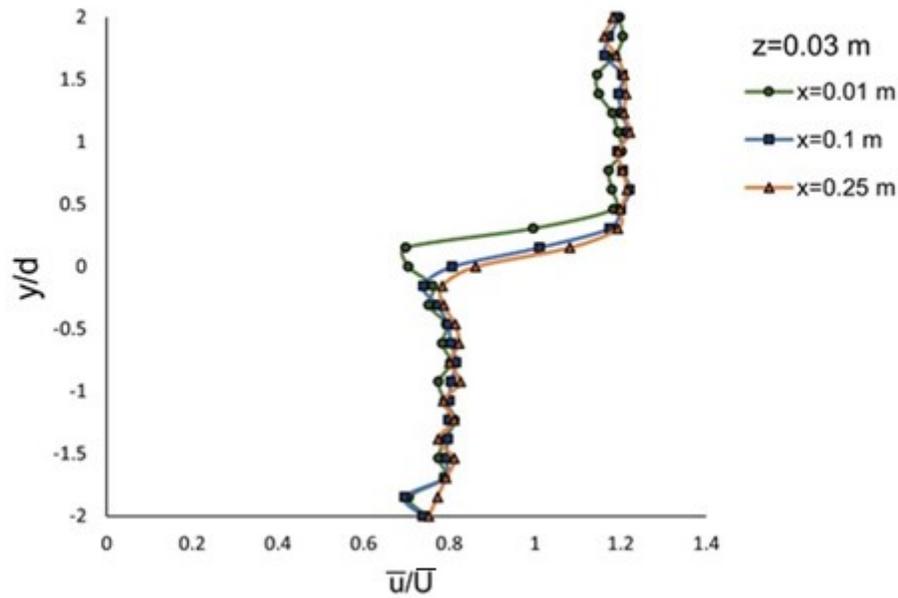


Figure 3.24. Ensemble average velocity profile for $V_r = 1.5$ at three distances from the splitter plate (0.01, 0.1, 0.25)

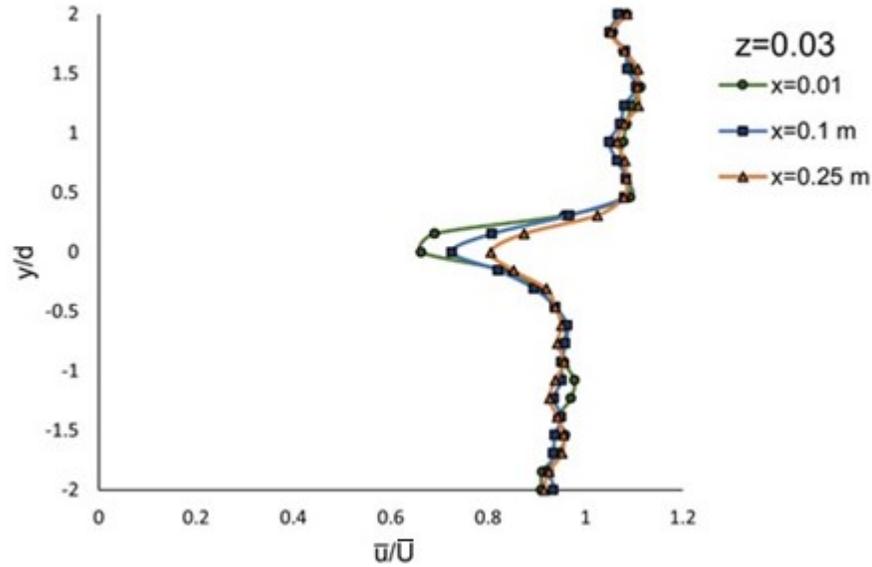


Figure 3.25. Ensemble average velocity profile for $V_r = 1.14$ at three distances from the splitter plate (0.01, 0.1, 0.25)

The values of δ determined from boundary layer definition are compared with the empirical relations of δ presented in Eqs 2.8 and 2.9, and the results are presented in Figure 3.26. In Eq.2.9, the constant α has been considered equal to 0.11 based on Uijtewaal and Booij (2000). Figure 3.26 shows that the mixing width determined from boundary layer definition estimates smaller values than empirical relations for $V_r = 1.14$. As shown in Figure 3.25, the velocity profiles are strongly influenced by the wake effect and the growing boundary layer on splitter plate walls. Therefore, the mixing layer development is constrained, producing smaller mixing width.

In the case of $V_r = 1.5$, the boundary layer development and the wake effect are limited by the larger lateral shear caused by the velocity ratio. Hence, the mixing width values determined from the boundary layer definition are compared well with Eqs 2.8 and 2.9.

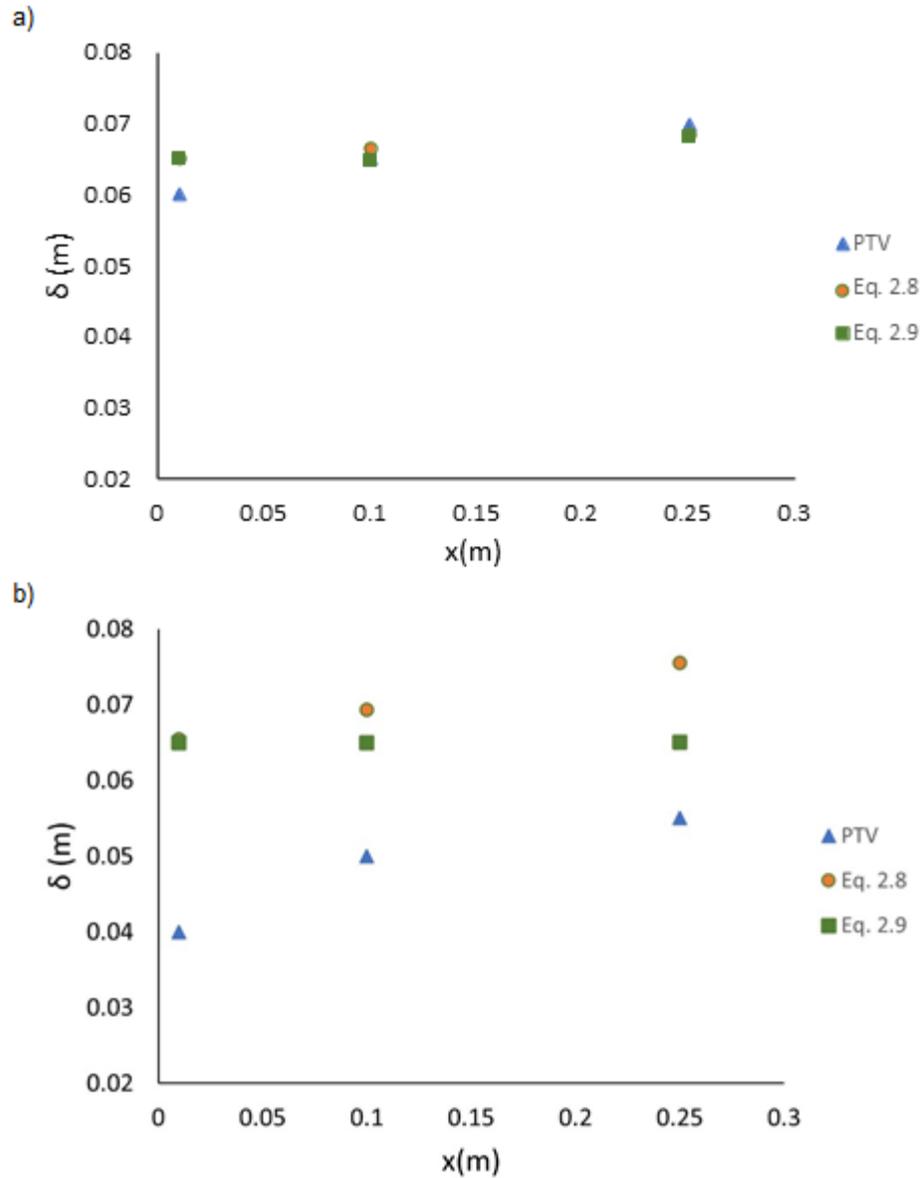


Figure 3.26. Comparison of mixing layer width between PTV measurements and empirical equations 2.8 and 2.9: (a) $V_r = 1.5$; (b) $V_r = 1.14$

3.5 Particle trajectory

One essential feature of PTV is the capability to provide particle trajectories. In order to capture the behaviour of large-scale coherent structures in SML, particle trajectories should be tracked for long time.

Uncertainties in the results from PTV experiments have various sources, including the accuracy levels of measurement tools. For example, the accuracy of the flowmeters (Model: Siemens Sitrans FM MAG 5000) used for measuring the discharge of each stream had an accuracy

of $0.4\% \pm 1$ mm/s. The accuracy of depth gage for measuring flow depth was 1 mm. The particle position determination using the same high-speed cameras as in this study gave positions in the x - and y -directions with an accuracy of ± 0.03 mm and in the z -direction with an accuracy of 0.18 mm (Akutina 2015). The frame rate in this study was set to 600 fps. We investigated the effect of frame rate (or equivalently the time interval) on particle velocities in the x -, y - and z -directions. The time interval did not have a significant impact on the velocity in the x -direction, as illustrated by a comparison of velocities between time intervals of $1/600$ s and $2/600$ s in Figure 3.27a for one particle. However, the velocity components in the y - and z -directions were significantly impacted by an increase in the time interval (Figures 3.27b,c). Figures 3.27d–f show the velocities in the x -, y -, z -direction for another particle trajectory with a different initial position [at $(x, y, z) = (0.016, 0.168, 0.014)$ m]. Figure 3.27d shows that u was less affected by increasing time interval, compared to v and w . Although the increase in the time interval by skipping one frame from every set of three consecutive frames had a small effect in terms of deriving the particle velocity in the x -direction, each of the three frames helped accurately track the particles. In other words, lowering the frame rate or increasing the time interval would result in losing track of particles and hence shortening particle trajectories.

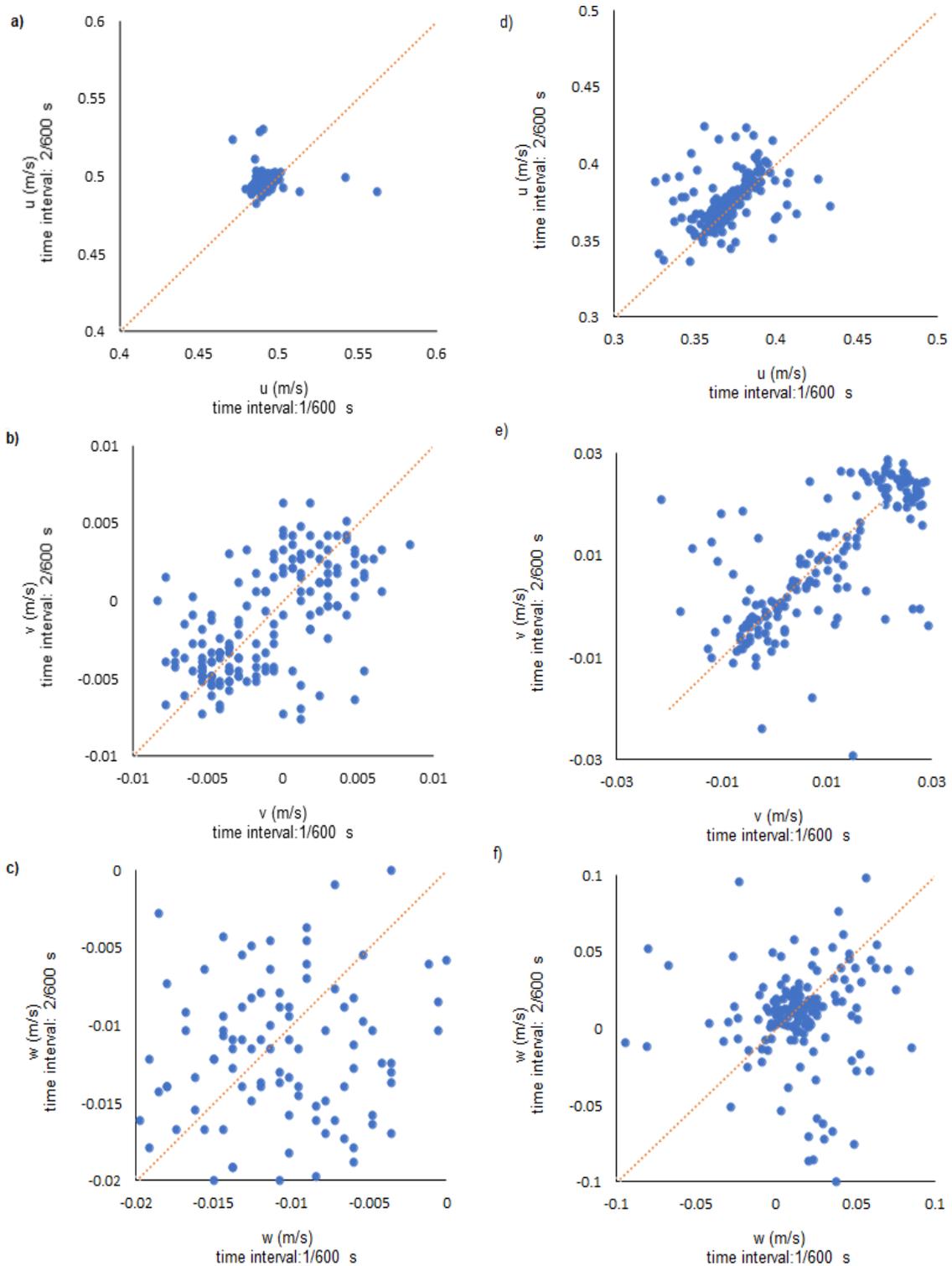


Figure 3.27. Comparison of velocities for $\Delta t = 1/600$ s and $\Delta t = 2/600$ s: a, d) velocity in the x -direction; b, e) velocity in the y -direction; c, f) velocity in the z -direction. In panels (a) to (c), the initial position of the particle was $(x, y, z) = (0.167, 0.024, 0.008)$ and in panel (d) to (f), the initial position of the particle was $(x, y, z) = (0.016, 0.168, 0.014)$ m

Producing particle trajectories in the mixing layer and its vicinity was important in this study because it enables further analysis of particle statistics. Previous experiments with the same setup as this study show that the length of the trajectories is 2 cm at maximum (Martinez 2017). Therefore, one of the primary objectives of the current study was to improve the experimental setup to increase the length of the particle trajectories. Figure 3.28 shows one example particle trajectory on an x - y plane for the case of a velocity ratio equals to 1.14.

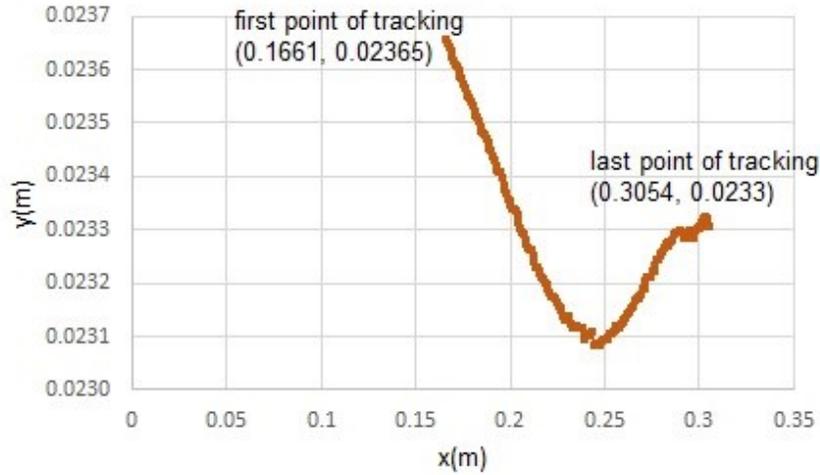


Figure 3.28. Sample trajectory of a single particle for $V_r = 1.14$

This study analyzed the particle tracking datasets using Python code called ‘FlowTracks’ developed at the University of Tel Aviv. Meller and Liberzon (2016) provided a flexible and efficient database management solution tailored for obtaining particle tracking information from Hierarchical Data Format (HDF) files.

One of the challenges in the experimental setup has been the short length of the trajectories identified by PTV in previous studies performed with a similar setup (Figs. 3.29 and 3.30). Luthi et al. (2005) pointed out that long particle trajectories are desirable for a better accuracy of the velocity calculations. One part of this study was to obtain long trajectories with the available setup without reducing seeding density, resulting in lower spatial resolution and losing flow information.

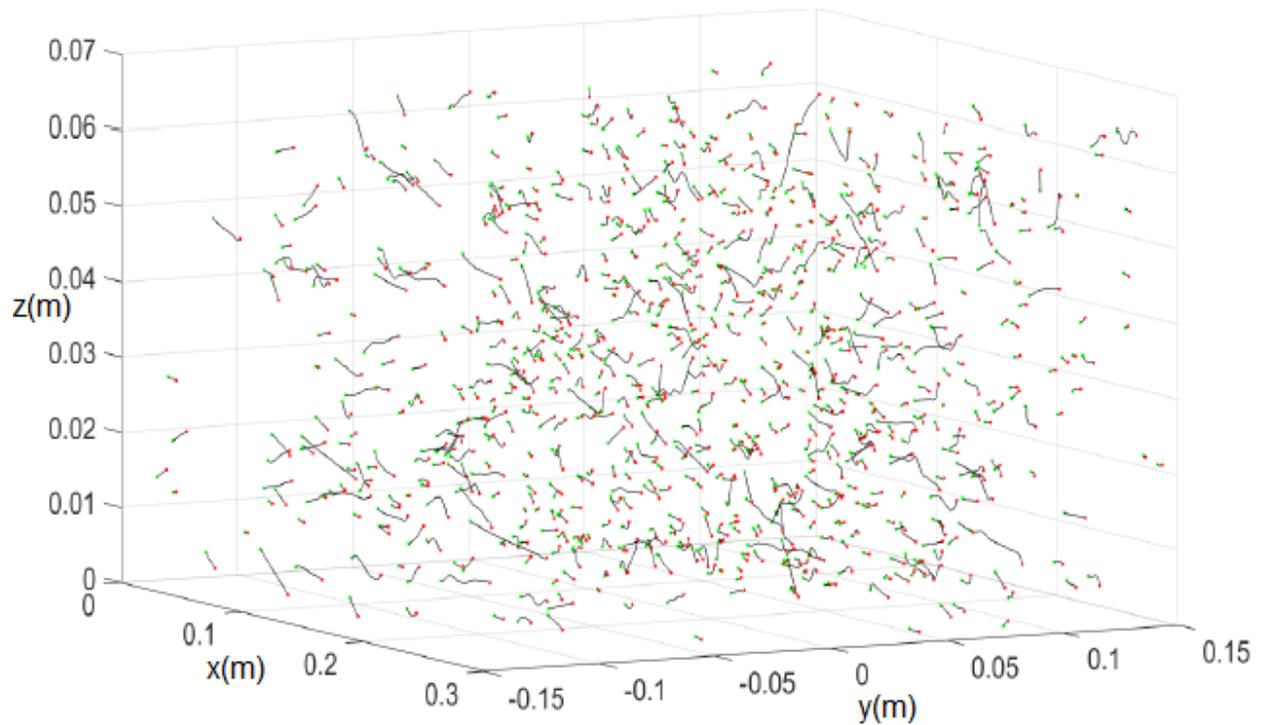


Figure 3.29. Particle trajectories identified by PTV in a SML collected from the same setup as this study (Martinez 2017)

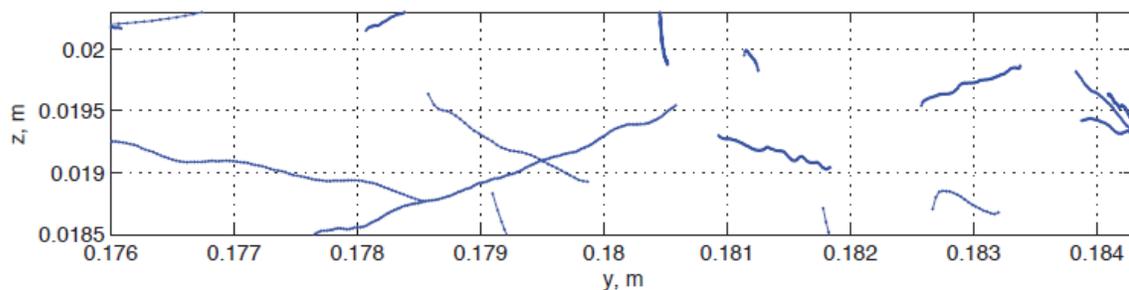


Figure 3.30. Particle trajectories identified by PTV in a shallow embayment (from Akutina 2015)

To increase the length of the particle trajectories, the particle indication in consecutive frames must increase. Following measures have been taken to increase the successful indication of particles in both data acquisition and postprocessing procedures of PTV

- Seeding flow with particles with smaller size range

In previous studies (Martinez 2017), a size range of 355 to 425 μm was utilized for seeding the flow. The optimum size of the particles should be 1.5 times the pixel size (Raffel 2007). In this study, the pixel size was 0.18mm, and a particle size range of 250–355 μm was selected. The

difficulty with using a smaller particle size range was that only a small number of particles after sieving are in the range of 250–355 μm . Considering that the price of the seeding particles is relatively high, we have taken new measures to increase the particle recovery after each experiment efficiently. Therefore, a filtration procedure was designed and implemented at the end of each experiment and before turning off the pump. The filter consisted of 10 nets, was installed downstream of the flume to collect the particles. In addition, when the pump was off and the tanks were full of water, particle recovery was made using a manual net. The recovered particles must be dried and sieved for the next experiments. It is noteworthy that after multiple applications, the colour and density of rice bran wax microspheres were changed, and they could not be used anymore.

- Using black tapes on the flume wall

In the previous study (Martinez 2017), the black tapes were installed on the flume walls to decrease the light reflection and direct the light to the study area. However, the quality of the images showed that the location of black tapes was reversely affecting the amount of light reaching the particles. When the illumination source is Laser-based, the black tapes can be used to cover unwanted areas on the flume wall. However, due to the wide-angle of spread for halogen lamps (35° – 55°), using black tapes on the flume wall results in an insufficient light beam reaching the flume centerline. Therefore, the black tapes were used only in places where a reflection was observed (such as the flume iron joints and the walls of the splitter plate).

- Applying Python code for incorporating data format based on the HDF5 format with a modular framework for querying the database.

The output file of most particle tracking algorithms is in ASCII format. This format makes the post-processing analysis of the PTV database difficult and causes the PTV data not to be used at its highest potential. In the current study, the python code enabled to use of the HDF5 format. This code also let various inquiries from the dataset, which helped develop another Python code for Lagrangian and Eulerian analyses. Figure 3.31 shows one example of the Python code developed in this study.

A PTV experiment was done with a velocity ratio equal to 1 as a reference case (two streams have the same velocity) to evaluate how the measures explained above have addressed the issue with the length of particle trajectories. Figure 3.31 shows the performance of the tracking in a reference case (Table 3.1). Figure 3.31 shows that the experimental measures have successfully increased the length of particle trajectories moving upstream to downstream of the study area (length of the study area in the x -direction was 30 cm).

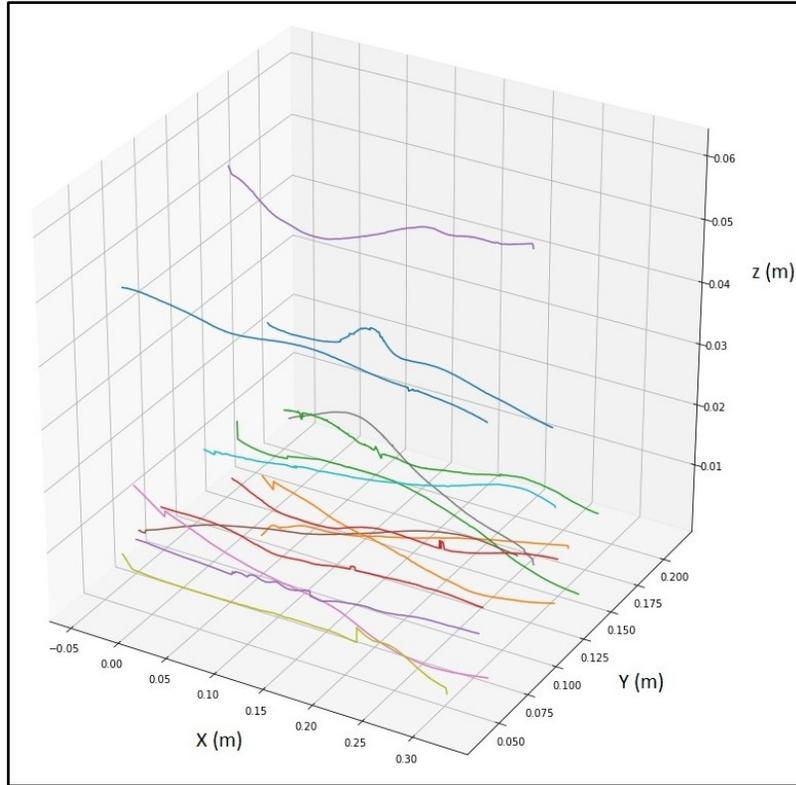


Figure 3.31. Trajectories of particles successfully tracked in 150 frames or more for $V_r = 1$ as an example output of a script developed in this study

Figure 3.32 shows the particle trajectories successfully were tracked in more than 150 consecutive frames for $V_r = 1.14$. One of the tools researchers have applied to study the Lagrangian behaviour of the mixing process is the finite-time Lyapunov exponent (FTLE).

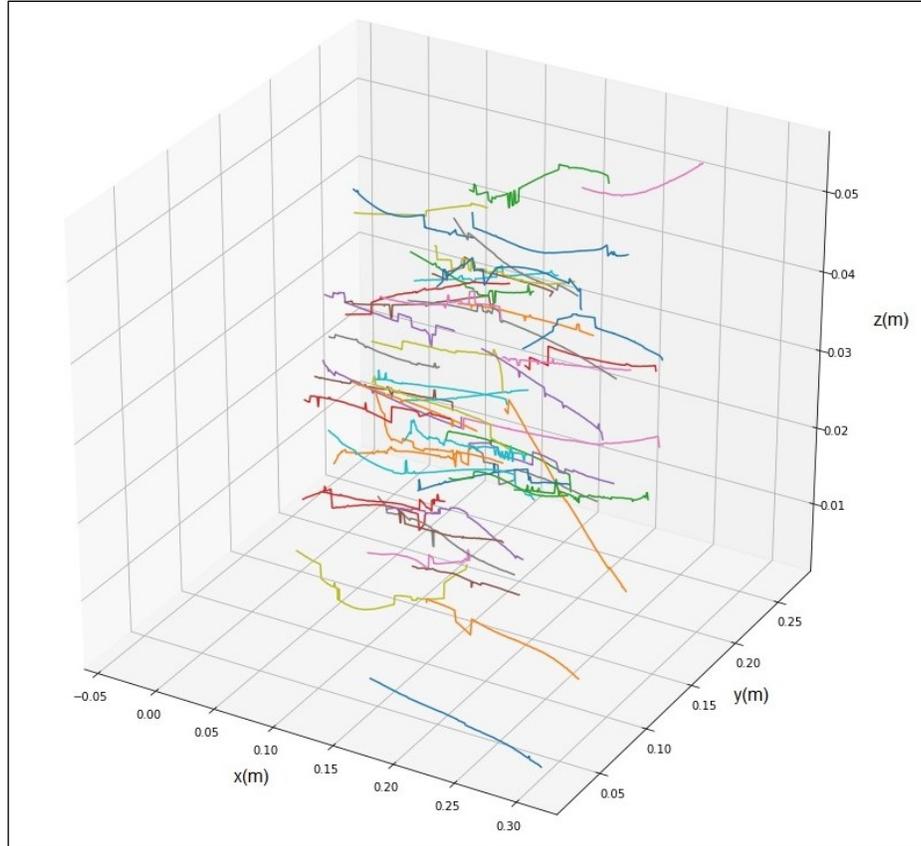


Figure 3.32. Trajectories of particles successfully tracked in 150 frames or more for $V_r = 1.14$

The concept of nearby trajectories diverging or growing away from each other plays an important role in studying the mixing process. Lyapunov exponent quantifies the average exponential rate of divergence of neighboring trajectories in space. Due to Lyapunov exponent's capability to measure a system's predictability and sensitivity to changes in its initial conditions, it has been used for stability analysis (Sprott 2003).

Lyapunov exponents can be considered the average logarithmic separation rate or convergence of two nearby points of two trajectories. To calculate Lyapunov exponents, we consider two points, in the trajectories i and j as x_i and x_j , in which the difference of initial coordinates is small. That means the system reaches near the same state at the i^{th} and j^{th} iterations. Each of the two trajectories represents a time series as $x_i, x_{i+1}, x_{i+2} \dots$ and x_j, x_{j+1}, x_{j+2} , etc. The fundamental of Lyapunov exponent is to know how these two trajectories diverge or converge to one another. The distance between the two trajectories after k steps is defined as $d(k) = |x_{i+k} - x_{j+k}|$. In a mechanical concept, if the system is chaotic, $d(k)$ initially increases exponentially with k . Therefore, one can plot $\ln d(k)$ versus k and apply a linear fit. The slope is an estimate for the Lyapunov exponent.

Since the system in this study is bounded by flume walls and bottom, the two nearby trajectories will not diverge indefinitely in a chaotic system, and exponential divergence stops after

some length. Therefore, instead of the Lyapunov exponent, we used the term ‘finite-time Lyapunov exponent’ (FTLE).

In the explanations above, $\ln d(k)$ is calculated for a single pair of trajectories. The Lyapunov exponent is the average divergence exponent over all nearby initial pairs, x_i and x_j such that the initial distance $d(0) = |x_i - x_j|$ is less than some fixed small value. This value in this study was equal to 0.001. To calculate the Lyapunov exponent, we have used a Python script that creates a dataset with $\langle \ln d(k) \rangle$ and k as the output.

In a chaotic system, at least one Lyapunov exponent must be positive, quantifying the ‘sensitivity to initial conditions’ (Schouten et al. 1996). If the system is dissipative, the sum of Lyapunov exponents is negative. However, the exponential growth of the $d(k)$ is impossible in a bounded system like the case of the experiments in this study, and the largest Lyapunov exponent captures the long-term average of the stretch as we move along the trajectory.

Exponential particle divergence is implied if the slope of the line is a positive value, and particle convergence is implied if the slope of the line is negative.

Figure 3.33 shows the xy -plane view of the particle trajectories for $V_r = 1.14$. Several particle trajectories were selected to perform FTLE analysis. The goal is to compare the FTLE for trajectories inside and out of the mixing layer. Figure 3.34 shows the FTLE determined for a pair of trajectories located at (0.16, 0.17, 0.014) m (in the mixing layer). Figure 3.35 shows the FTLE determined for a pair of trajectories located at (0.06, 0.27, 0.021) m (out of mixing layer). The tip of the splitter plate is located at $x = 0$ m and $y = 0.15$ m in Figure 3.33.

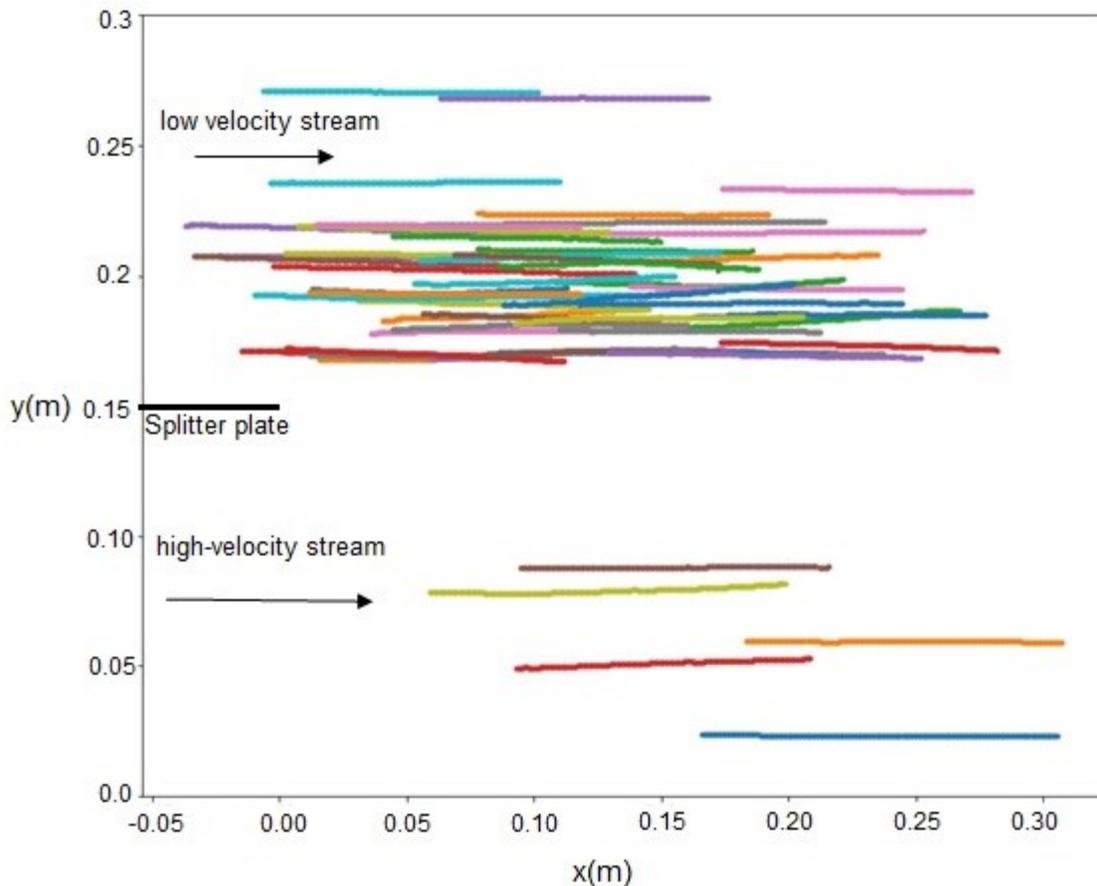


Figure 3.33. Particle trajectories in plane view for $V_r = 1.14$

In Figure 3.34, the FTLE is determined as 0.011. The slightly positive value of FTLE demonstrates the growth of the mixing layer gradually. This observation is in close agreement with PTV Eulerian results, where the mixing growth was observed to increase gradually. Figure 3.34 shows the FTLE calculated for a particle pair located at the farthest y from the splitter plate captured in PTV experiments. Considering that the initial width of the mixing layer is considered 0.065 m (Figure 3.26), the selected particle is located out of the mixing layer and is less affected by the mixing process. As Figure 3.35 suggests, a negative FTLE (slope of the line) at $k < 100$ represents a convergence process for these particles. Although for $k > 100$, the trend is changed, and divergence of the particles is evident with a positive FTLE. It is noteworthy that the negative value of FTLE does not contradict the existence of turbulent flow out of the mixing layer as the particle pairs start to diverge for $k > 100$.

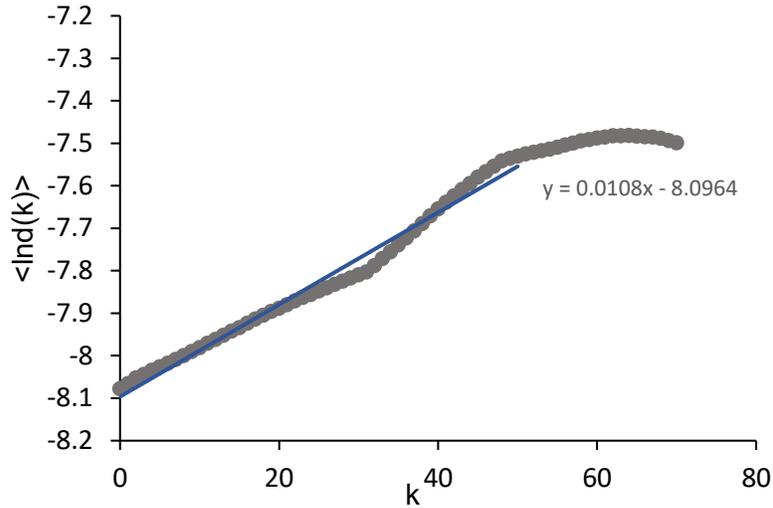


Figure 3.34. FTLE for particle pairs initially located in the mixing layer for $V_r = 1.14$

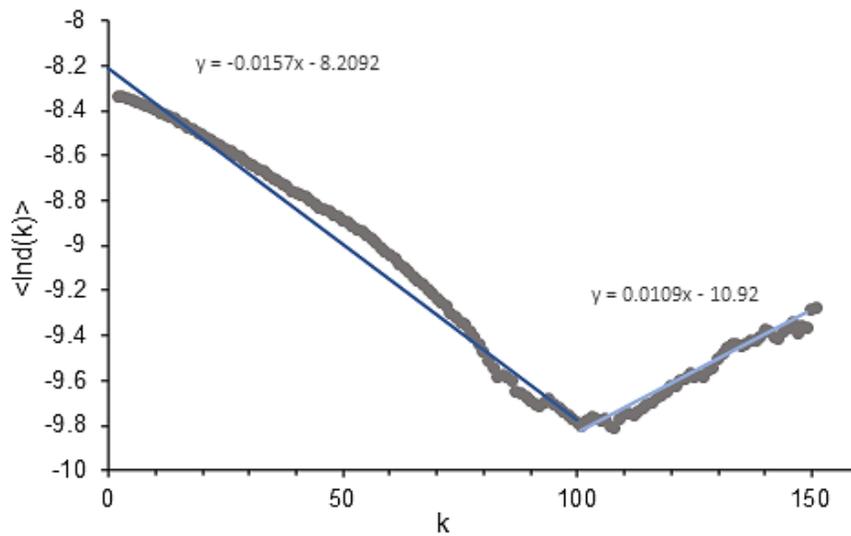


Figure 3.35. FTLE for particle pairs initially located out of the mixing layer for $V_r = 1.14$

In the PTV experiments in this study, some challenges were recognized. These challenges are elaborated in the next section, and the potential solutions for these challenges are proposed. One of the most important challenges was that the particle tracking did not produce sufficient particle trajectories in the mixing region and high-velocity side of the mixing layer for $V_r = 1.5$. Some measures can be considered to address this issue which is discussed in the next section.

3.6 Limitations of experiments

PTV involves a long chain of measurements, image processing, coordinate determination, and particle tracking before producing the desired temporal and spatial particle velocities. The complexity of the procedure could cause the improvement of the technique complicated since many contributing factors are involved in the PTV procedure. The application of PTV to study large-scale coherent structures in SML from the Lagrangian perspective was firstly investigated in this study. Therefore, this study provides its experiences in the challenges of applying PTV to study SML and suggests the solutions for data acquisition and image processing stages.

In data acquisition, some considerations can potentially improve the quality of the images and hence, the resulting particle trajectories. Some of these considerations that were feasible in the laboratory were discussed in Section 3.5. In this section, the limitations of PTV when studying SML are discussed.

Although high-speed cameras with a high capacity of image frequency (fps) are needed to ensure data acquisition quality, illumination is one of the crucial components that could affect the results. The amount of light received by the particles depends on the light source wavelength, beam angle, transmitting distance, light flicker, and power of the illumination source. The beam angle is defined as the angle at which the light is distributed or emitted. Transmitting distance defines how far the light beam can travel. Light flicker refers to quick repeating changes in the brightness of light over time.

In this study, LED lamps were tested as the illumination source. Although, at first glance, the slow transmitting speed caused some of the images to be dark and not usable. The wide beam angle of LED was another reason to select the halogen lamps as the illumination source. Halogen lamps at the sides of the flume were installed to illuminate the study area. Although the images collected in the reference test with the same velocity of two streams had successfully created particle trajectories, for larger velocity ratio cases ($V_r = 1.5$), where a significant shear is affecting particle movement (in the mixing layer), more intensive light could improve quality particle recognition and tracking.

Luthi (2002) used two different light sources for the same PTV setup. One light source consisted of two 400 Watt metal, and another consisted of 5 Watt Argon Ion Laser. Luthi (2002) performed PTV experiments in a small fluid volume $120 \times 120 \times 140 \text{ mm}^3$ and suggested that the Ion laser provides better particle recognition than halide lamps.

An Ion Laser provides a faster-transmitting light flicker and a longer transmitting distance than both LED and Halogen lamps. The transmitting distance is of great importance in studying SML because the flume walls are supposed to be far from the channel centerline to minimize the effect of walls on the mixing process. Therefore, studying SML in a wide channel demands a light source with a longer transmitting distance.

In PTV and PIV, the point velocity is estimated using the fundamental definition of velocity, the displacement of the particle over time. PIV has some limitations that have resulted in more efforts in the development of PTV in past years. One limitation of PIV is that it is not suited to track individual particles. PTV overcomes this limitation at the expense of sacrificing some degree of spatial resolution (Malik 1993). Therefore, the building block for the determination of flow characteristics using PTV is particle displacement estimation.

One essential step of the PTV tracking method is to estimate the most likely expected location of a particle e_i^{f+1} , at the next time step in the frame $f+1$ of the particle at current frame time x_i^f . e_i^{f+1} is estimated from information derived from the established links between frame $f-1$ and the frame f in the correlation neighborhood of particle x_i^f (Figure 3.36). For example, if the current particle in the frame f is linked to a particle in the frame $f-1$. In that case, the expected location in the frame $f+1$ is taken as the linear continuation of this link, without acceleration. This approach results in having a track of the particle trajectory as long as the particle is moving with a slight change in its direction. Therefore, the current tracking algorithm for searching the particle in consecutive frames is not tolerant to large deformation. Although the image capturing frequency of PTV experiments in this study was 600 fps, the particle trajectories in the channel centerline were lost due to high shear stemming from the velocity gradient between two streams. Extensive research in developing new algorithms capable of tracking the particles in highly shear flows is recommended.

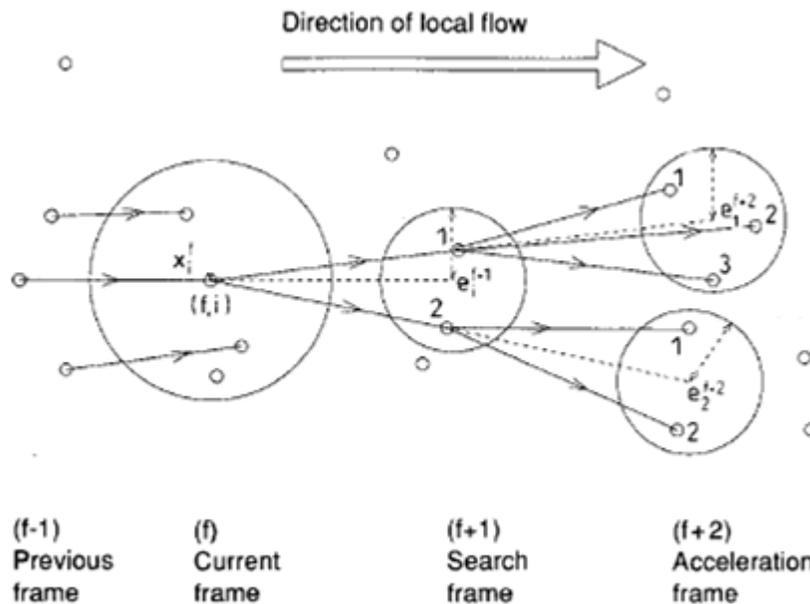


Figure 3.36. Schematic of particle tracking procedure (from Malik et al. 1993)

Limitations in experiments of SML using PTV technique stated in this section and other factors such as being expensive and time-consuming to perform one single test using PTV highlight the necessity of numerical simulation in providing a more complete picture of the SML characteristics. The velocity ratio between two streams that can be tested using PTV is limited and a large velocity ratio results in loss of particle trajectory recognition. This issue can be addressed by performing numerical simulation discussed in the next chapter.

The PTV experiments were performed up to 0.3 m from the splitter plate. In order to have the interpretation of SML in long distances from the splitter plate, we performed a series of dye

visualization experiments. In the following section, the dye visualization experiments are presented.

3.7 Dye visualization experiment

Dye-based flow visualization has been of great importance in studying the dynamics of turbulent flow. By observing a tracer's motion, researchers can understand the formation and transition of flow structures.

The shallow mixing layer between the two parallel flowing streams has been visualized for three cases with velocity ratios $V_r = 1.14, 1.5,$ and 1 (no-velocity gradient as a reference case). Dye has been injected into the flow at the tip of the splitter plate, and the observation area of $2 \times 2 \text{ m}^2$ was recorded by one single camera with a frequency of 30 fps looking parallel to the flume length. The flume bottom was covered with white cardboards, which were measured and scaled to be used in the calibration process.

The frames were analyzed using Matlab with calibrating and measuring properties. The Matlab script enables measuring the reference length for calibrating the number of pixels of the image associated with the reference object's actual length of the reference object.

3.7.1 Mixing mechanisms

Flow visualization for the case of zero velocity gradient between the two streams shows that the instabilities start to grow just at the tip of the splitter plate due to the presence of boundary layers forming on each side of the splitter plate. The velocity deficit in the boundary layers results in a wake forming just downstream of the splitter plate. The instabilities initiated at the tip of the splitter plate grow independently on both sides of the wake, which keeps two streams from interacting. A reasonable assumption is that the turbulent spreading rate tends to zero when the velocity difference between the two streams is zero. In reality, the initial boundary layers formed on both sides of the splitter plate becomes dominant, and the flow becomes a wake flow (Figure 3.37).

Similar phenomena are observed in the wake region when the lateral shear is slightly larger with the velocity ratio $V_r = 1.14$. Just downstream of the wake, the instabilities pair together and form large coherent structures, which, as they are convected downstream, continue to grow in size. Large coherent structures seen in the experiment results are defined as a turbulent flow whose vorticity expression contains orderly components that can be described as instantaneously coherent over the spatial extent of the flow structure (phase-correlated).

In the case of zero velocity gradient, from $x = 0.1 \text{ m}$ ($x/d = 1.54$), the wake effect weakens, and the instabilities start to interact. However, the lack of a velocity gradient leads to a weak pairing process. The non-vortical cores convected downstream and eventually dissipated at $x > 0.9 \text{ m}$ ($x/d > 14$).



Figure 3.37. Flow visualization for streams with the same velocity. The distance between transverse lines is 10 cm

A critical aspect of the process of mixing called ‘pairing’ was first indicated by Brown and Roshko (1974), in which the two (or more) vortical cores entangle and create a larger coherent structure. The results of the dye visualization show the pairing process right downstream of the wake. The convective speed is independent of the size or location of eddies as it depends on the free stream velocity of each stream of the flow. Therefore, at a specific location x , it might seem that the frequency with which eddies pass the location x is constant. However, the requirement of the increasing size of the eddies requires a decreasing frequency.

One can conclude that while eddies are convected downstream, they must amalgamate in some way into larger structures, and this process must continuously recur with increasing x . Vortex pairing is the stage at which the tandem arrangement of two successive vortices becomes unstable, and they rotate around each other and become one. The new structure is then convected downstream until the subsequent encounters (Figure 3.38).

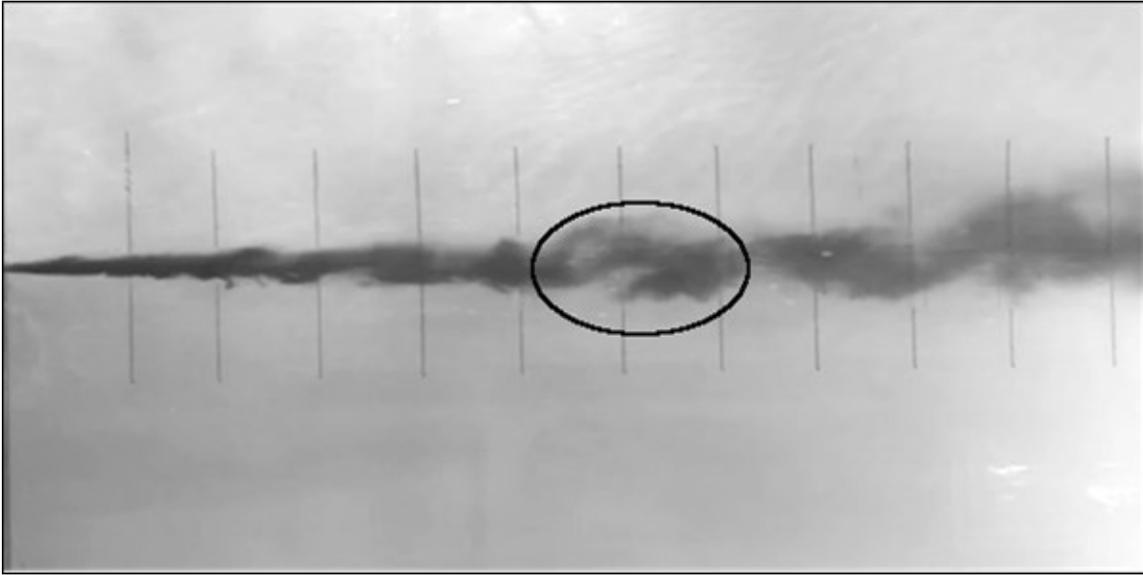


Figure 3.38. Pairing process as a function of downstream distance for $V_r = 1.5$

The lifetime of the entangled eddies is very short in the case of velocity ratio equal to 1.14 since the velocity gradient decreases in the downstream direction, and hence the velocity gradient is not feeding the eddies with energy, and they decay at $x > 1$ m ($x/d > 16$) as shown in Figure 3.39.

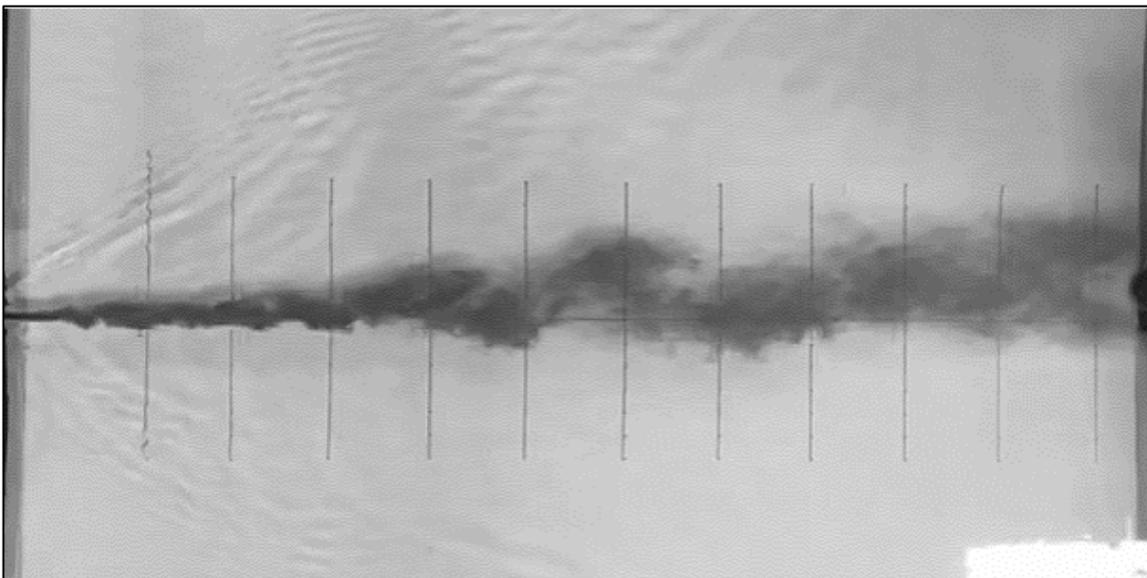


Figure 3.39. Flow visualization for streams with a velocity ratio of 1.14

The strongest eddies are seen for the case of a velocity ratio equal to 1.5, and this case gives a more complete image of the mixing processes. In Figure 3.39, three regions for the evolution of

the mixing layer can be identified. Immediately downstream of the splitter plate, the wake effect is significant, and the vortical cores start to form. After the wake region, the entrainment of the fluid occurs, and the vortex pairing forms elongated two-core coherent structures. This region is located at $7 < x/d < 10$. The third region can be identified by the discrete large-scale eddies formed from paired vortices located at $12 < x/d < 30$. These are the strongest eddies, increasing their size while being convected downstream. This observation agrees with the entrainment process as a function of distance defined by Winant and Browand (1974).

3.7.2 Characteristics of coherent structures

In the mixing layer, the spacing of the eddies increases with increasing distance downstream. The diameter of the eddies also increases. The image of the dyed fluid in the mixing layer taken at random times (Figure 3.40) shows that there does not appear to be a temporally fixed pattern in the vortex arrangement, but there is only a general resemblance in that the scale increases.



Figure 3.40. Random frames of visualization at a specific location ($x = 110\text{--}170$ cm) and different times for velocity ratio equal to 1.5

An interesting aspect of the pairing process is the deformation of the vortices. While they roll around each other, they become elongated in the flow direction. This study identified this process as an approximately elliptic shape of eddies with a major-to-minor axis ratio of 2. This deformation indicates that the pairing is more than a simple solid-body rotation of the vortices around each other: the induced velocities must generate strains, which deform the vorticity distribution. Figure 3.41 shows one example of this deformation with elliptic shape eddies with major to minor axis ratio of 2.

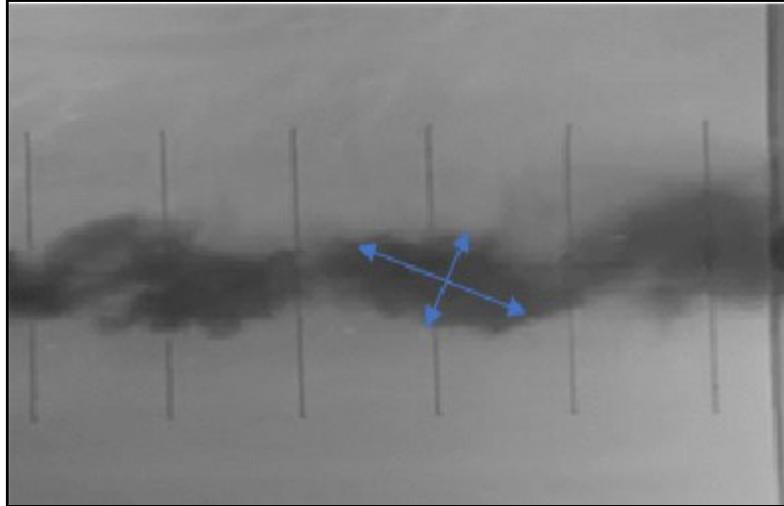


Figure 3.41. Deformation of eddies with elliptic shape for velocity ratio equal to 1.5

Brown and Roshko (1974) measured the eddy spacing identified with the value of x midway between the two consecutive eddies. Their sample size was 3600 eddies, not necessarily independent since some eddies were tracked over many frames. Their results show that the mean value of the spacing is $l_m = 0.31(x - x_0)$, where x_0 is the virtual origin and is defined as the location where the width of the mixing layer is equal to zero. Usually, x_0 is determined operationally, and from the results of this study, the value of x_0 is estimated as zero. Brown and Roshko (1974), using the mixing layer width (δ), found the mean spacing value equals to 2.9δ . In this study, to estimate the eddy spacing in a shallow mixing layer, based on Winant and Browand (1974), the distance between vortex cores has been counted in each frame for 500 frames. If the coherent structure is in a transitional stage and the vortex has not been formed, it is neglected in the counting. A histogram of eddy spacing for shallow mixing layer for $V_r = 1.5$ is presented in Figure 3.42.

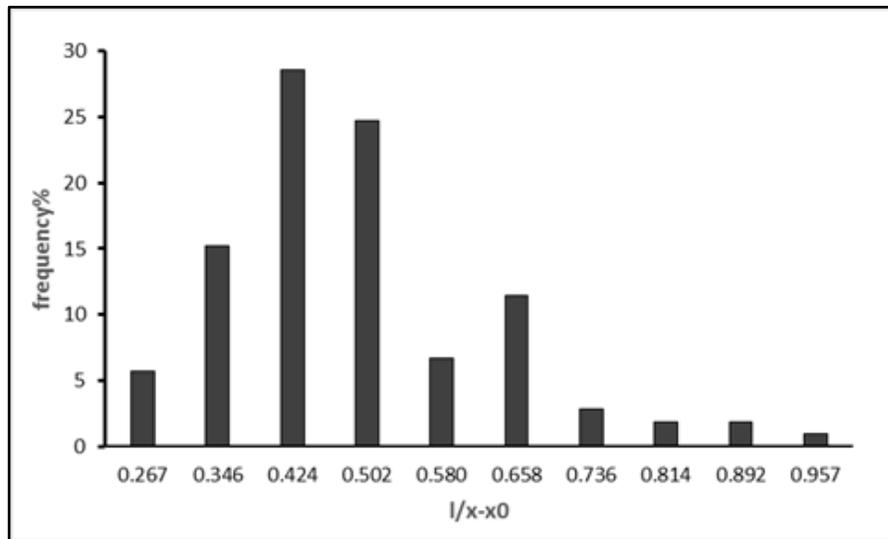


Figure 3.42. Eddy spacing histogram for $0 < x < 2$ m downstream of the splitter plate for $V_r = 1.5$

The distribution of eddy spacing from the tip of the splitter plate up to $x = 2$ m in Figure 3.43 shows a linear relation between eddy spacing and downstream distance with a slope of 4.5 to 5, which agrees with the most frequent eddy spacing in Figure 3.42. However, the tendency of outliers to the low side of the line would emphasize the need to validate the linearity for $x > 2$ m.

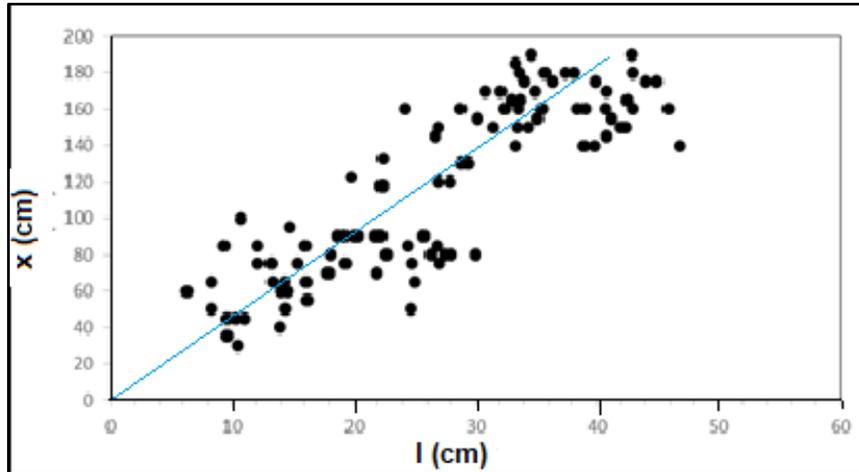


Figure 3.43. Eddy spacing distribution in the downstream distance x for $V_r = 1.5$

3.8 Conclusion

An experimental investigation of SML has been presented in this chapter. The experiments were conducted in two parts: near-field investigation using PTV; near-field and far-field investigation using dye visualization. For PTV experiments, some of the limitations to previous studies have been overcome in the current study. The measures introduced in the data acquisition of PTV, and the Python script developed for particle trajectory analysis improved the PTV techniques in terms of producing long trajectories. An ensemble average method was used to perform the Eulerian analysis of stream velocity. The results show that the boundary layer development on splitter plate walls and the wake effect is limiting the mixing layer growth. However, these effects are weakened by the larger lateral shear for cases with a larger velocity ratio. The effect of boundary layer growth on splitter plate walls on the velocity profiles gradually disappears in downstream distances. A new approach for the determination of mixing layer width was proposed based on the boundary layer definition.

FTLE was determined for particle trajectories to measure divergence and convergence of particles inside or out of the mixing layer. For particle trajectories located inside the mixing layer, a divergence was evident with a positive value of FTLE. However, for the particle trajectories out of the mixing layer, both positive and negative FTLE can be observed.

In part two of the experimental investigations, a series of dye visualization was performed. These experiments shed light on the flow behavior of SML in the far-field. The results show that the turbulent instabilities still form in a lack of velocity gradient ($V_r = 1$). However, in the presence of a velocity gradient, the instabilities persist, and a pairing of eddies is observed. The intermittency of SML is observed with a lack of a temporally fixed pattern in the vortex arrangement at a fixed location.

The experimental investigation of SML using the PTV technique has shown to have several limitations discussed in this chapter. Considering that PTV is an expensive and time-consuming technique to perform, in addition to the limitations discussed before, the necessity of numerical simulation in providing a more complete picture of SML characteristics is highlighted. The velocity ratios between two streams that can be tested using PTV are limited, and a significant velocity ratio results in loss of particle trajectory recognition. This issue can be addressed by performing numerical simulation presented in the next chapter.

Chapter 4

Smoothed particle hydrodynamics simulation of shallow mixing layer

4.1 Introduction

Computational fluid dynamics (CFD) methods numerically solve mass conservation, momentum balance and energy conservation equations as a mathematical model of fluid flow. The conservation laws must be satisfied by fluid motions in a CFD model. The model equations describe relevant physical processes, and the solution methods produce velocity field, pressure field and scalar properties. There are two approaches to describe the motions: One is the Lagrangian approach, in which the equations are formulated directly from the properties of a given fluid element moving in space; the other is the Eulerian approach, in which the conservation principles are applied to distributed properties such as fluid density and velocity on a mesh system. Historically, the Eulerian approach is more commonly used in CFD models, compared to the Lagrangian approach.

The Eulerian approach requires employing mesh, which is faced with significant numerical challenges. For example, inaccuracy in computational results can arise from truncation errors in the discretization of a partial derivative $\partial u/\partial x$:

$$\left. \frac{\partial u}{\partial x} \right|_i = \frac{u_{i+1} - u_i}{\Delta x} + \left[\frac{\Delta x}{2!} \left. \frac{\partial^2 u}{\partial x^2} \right|_i + \frac{(\Delta x)^2}{3!} \left. \frac{\partial^3 u}{\partial x^3} \right|_i + \dots \right] \quad (4.1)$$

where u_i is streamwise velocity at grid node i ; Δx is the grid size. Using the first term on the right-hand side of Equation 4.1 to replace the partial derivative $\partial u/\partial x$ in a CFD model equation gives the first order finite difference approximation. The approximation produces valid results only if the truncation error vanishes at $\Delta x \rightarrow 0$. In practice, $\Delta x \rightarrow 0$ is never attained; therefore, the truncation error depends on the grid size, the order of approximation, and order of derivatives: $\partial f/\partial x$, $\partial^2 f/\partial x^2$, ..., where f is a dependent variable. In general, the numerical accuracy increases with a decreasing grid size, which reduces the errors of approximation.

In the Eulerian approach, the behavior of the approximated function impacts the accuracy of approximations (Figure 4.1). If the function has a strong gradient, as seen from the curve u in Figure 4.1, the amplitudes of derivatives are large, and hence, the truncation error can be significant even when the numerical consistency is satisfied and Δx is not particularly large. In a shallow mixing layer (SML), the flow field features large gradients with respect to space as well as with respect to time. For this reason, it would be difficult to obtain accurate CFD results using the Eulerian approach.

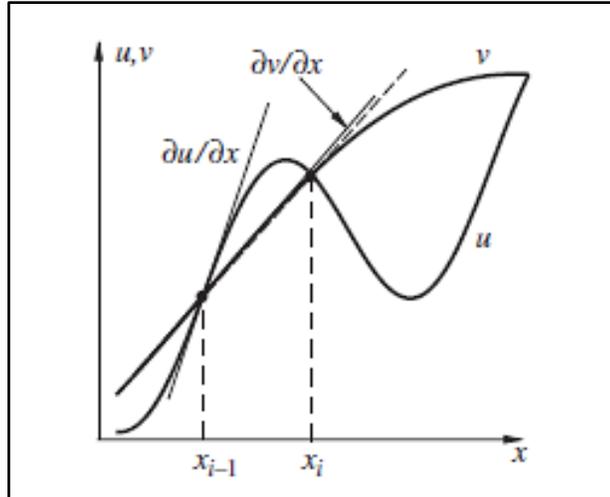


Figure 4.1. Effect of strong gradients of a function on truncation error. The finite-difference approximation (slope of dashed line) accurately reproduces $\partial v / \partial x$, but not $\partial u / \partial x$. (from Zikanov, 2019)

There are several advantages in modeling fluid flow using the Lagrangian approach over the Eulerian approach. For example, in the Lagrangian approach, the conservation of mass is automatically satisfied as there is no mass flow through the Lagrangian element. In addition, remeshing (one characteristic of mesh-based methods) is not required at each numerical iteration, resulting in lower computational costs.

Lagrangian modeling is based on the hypothesis that the CFD model domain can be divided into a finite number of particles, and the center of mass of each particle receives the coordinates that define its position in space. The physical properties of each Lagrangian element are found at each instance of time. These particles move according to the kernel influence of their neighboring particles. The mesh-free Lagrangian particle model is increasingly applied to various physical problem simulations.

Simulation of free surface flows using the mesh-based methods is challenging (Gabriel et al. 2018). For example, in Volume of Fluid (VoF) method, as the free surface moves through a grid the solution region is continuously changing. At the free boundaries of this changing region the proper free surface stress conditions must also be applied and updated by time which is computationally complex. In Lagrangian modeling, free surfaces, interfaces, and moving boundaries can be naturally traced by particles in the simulation process. Furthermore, the history of the particles can be obtained, and their graphical visualization is accessible, allowing a better understanding of the spatial and temporal evolution of the fluid and its properties.

The main objectives of the SML simulation using Smoothed Particle Hydrodynamics (SPH) are to answer the following questions:

- How does SML behave under large ratios between two incoming flow velocities?
- To what extent do the wakes downstream the initial joining point (Figure 4.6b, point O) affect the vortical motions?
- Are transverse profiles of streamwise velocities self-similar?
- What factors control the pairing of adjacent eddies and the grow of large-scale coherent

structures (CSs)?

- How does the fluid mass and momentum exchange is affected by the velocity ratio between the two streams?

4.2 Smoothed particle hydrodynamics theory

Mesh-free methods were developed in early studies of Frazer et al. (1937), and Lanczos (1938). In recent years, the mesh-free Smoothed Particle Hydrodynamics (SPH) technique, firstly introduced by Gingold and Monaghan (1977) to solve astrophysical problems, has been developed and successfully used for flow simulations in a wide range of CFD applications. Examples include multi-phase flow with a sharp material interface (Colagrossi and Landrini 2003; Hu and Adams 2007) and dam-break flow (Gómez-Gesteira and Dalrymple 2004).

Federico et al. (2012) used SPH to study a uniform laminar open channel flow of low Reynolds Number. Later, Meister et al. (2014) used the same numerical technique to study the steady laminar open channel flows with different water viscosities.

The Lagrangian CFD method discretizes a continuum using a set of material points or particles. In the case of fluid dynamics simulation, the discretized Navier-Stokes equations are locally integrated at each of these particles' locations, according to the surrounding particles' physical properties. The set of neighboring particles is determined by a distance-based function usually called a kernel function with an associated characteristic length (or smoothing length), denoted as h (Figure 4.2). At each timestep, new physical quantities are calculated for each particle, and in the next time step they move according to the updated values.

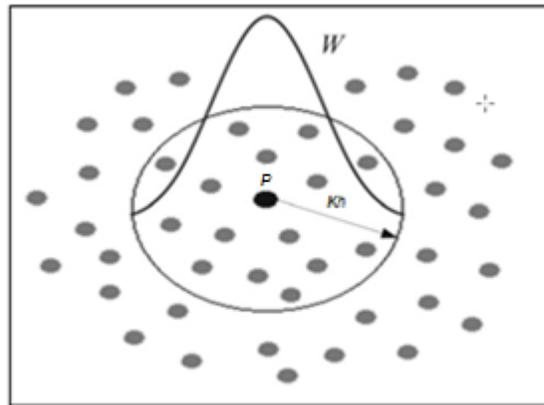


Figure 4.2. Representation of the kernel function, W , and its domain of influence for particle P (from Fraga Filho et al. 2019)

The conservation laws of fluid dynamics are transformed from their partial differential form to a suitable form for particle-based simulation using integral equations based on an interpolation function, which estimates values at a specific point.

The interpolation or weighting function, referred to as the kernel function (W) can take different forms, the most common being cubic or quintic. Let $F(\mathbf{r}')$ denote a function value at point \mathbf{r}' . The kernel function represents $F(\mathbf{r})$ by the integral approximation:

$$F(\mathbf{r}) = \int F(\mathbf{r}')W(\mathbf{r} - \mathbf{r}', h)d\mathbf{r}' \quad (4.2)$$

The kernel function must fulfill several properties, including positivity inside a defined zone of interaction, monotonically decreasing value with distance, and differentiability (Monaghan 1992; Liu and Liu 2003).

The discrete form of Eq. 4.2 is

$$F(\mathbf{r}_a) \approx \sum_b F(\mathbf{r}_b)W(\mathbf{r}_a - \mathbf{r}_b, h)\Delta v_b \quad (4.3)$$

where $F(\mathbf{r}_a)$ is the function value for a particle a ; $F(\mathbf{r}_b)$ is the function value at all particles in the region of support defined by the smoothing length h and the distance between \mathbf{r}_a and \mathbf{r}_b ; v_b is the volume of the neighboring particle b , defined as $v_b = m_b/\rho_b$; m and ρ are the mass and the density of the particle b , respectively. Substituting v_b in Eq.4.3, we obtain

$$F(\mathbf{r}_a) \approx \sum_b F(\mathbf{r}_b) \frac{m_b}{\rho_b} W(\mathbf{r}_a - \mathbf{r}_b, h) \quad (4.4)$$

The classic SPH approximation of the density field, ρ_i , is obtained using

$$\rho_a = \sum_{b=1}^n m_b W(\mathbf{r}_a - \mathbf{r}_b, h) \quad (4.5)$$

In the SPH model of this study, the initial density of particles was set to 1000 kg/m³. The mass of particle is determined by $m_a = \rho_a dp^2$. This is for the case of two dimensions, and in the third dimension the length is considered to be unity. In this study m_a was equal to 0.004 kg. It is noteworthy that the mass of particle is constant during the simulations. The particle density variation is less than 1%. From Eq. 4.5, it can be seen that the unit of kernel function is m⁻³ in 3D simulations. From the mass conservation law, one can obtain

$$\frac{D\rho_a}{Dt} = \sum_{b=1}^n m_b (\mathbf{v}_a - \mathbf{v}_b) \cdot \nabla W(\mathbf{r}_a - \mathbf{r}_b, h) \quad (4.6)$$

where t is time; n is the number of particles in the domain of influence; \mathbf{v}_a and \mathbf{v}_b are the velocity vectors for particles a and b , respectively.

The general form of momentum equation in a continuum can be written as

$$\frac{D\mathbf{v}_a}{Dt} = -\frac{1}{\rho} \nabla P + \mathbf{g} + \Gamma \quad (4.7)$$

where \mathbf{g} is the gravitational acceleration; P is the pressure; Γ is the dissipative term caused by viscosity and turbulence. In this study, the channel was horizontal, and the simulation was 2D. Therefore, the gravitational acceleration components were zero in the x - and y -direction.

In SPH, there are two methods to include the effect of dissipation in the Γ term, including artificial viscosity, and laminar viscosity combined with sub-particle scale turbulence. The artificial viscosity scheme, proposed by Monaghan (1992), is a common method in fluid simulation using SPH due to its simplicity. The artificial viscosity is defined as follows

$$\Pi_{ab} = \begin{cases} -\frac{\alpha_{sph}\bar{c}_{ab}\mu_{ab}}{\bar{\rho}_{ab}} & \text{if } \mathbf{r}_{ab} \cdot \mathbf{v}_{ab} < 0 \\ 0 & \text{if } \mathbf{r}_{ab} \cdot \mathbf{v}_{ab} \geq 0 \end{cases} \quad (4.8)$$

where $\mathbf{r}_{ab} = |\mathbf{r}_a - \mathbf{r}_b|$; $\mathbf{v}_{ab} = \mathbf{v}_a - \mathbf{v}_b$; $\bar{c}_{ab} = \frac{1}{2}(c_a + c_b)$ is the mean speed of sound; $\mu_{ab} = h\mathbf{r}_{ab} \cdot \mathbf{v}_{ab}/(r_{ab}^2 + \eta^2)$; $\eta^2 = 0.01h^2$; α_{sph} is a coefficient that needs to be tuned to introduce the proper dissipation. Altomare et al. (2015) suggested $\alpha_{sph} = 0.01$.

With the artificial viscosity in the dissipative term, the momentum equation is given by

$$\frac{D\mathbf{v}_a}{Dt} = -\sum_b m_b \left(\frac{P_a + P_b}{\rho_a \rho_b} + \Pi_{ab} \right) \nabla_a W_{ab} + g \quad (4.9)$$

Lo and Shao (2002) presented the laminar viscous stress term in the momentum equation as

$$(\nu_0 \nabla^2 \mathbf{v})_a = \sum_b m_a \frac{4\nu_0 \mathbf{r}_{ab} \cdot \nabla_a W_{ab}}{(\rho_a + \rho_b)(r_{ab}^2 + \eta^2)} \mathbf{v}_{ab} \quad (4.10)$$

where ν_0 is the kinematic viscosity of water (equal to 10^{-6} m²/s). Applying the laminar viscous stress into the momentum equation, we obtain

$$\frac{D\mathbf{v}_a}{Dt} = -\sum_b m_b \left(\frac{P_a + P_b}{\rho_a \rho_b} \right) \nabla_a W_{ab} + g + \sum_b m_b \frac{4\nu_0 \mathbf{r}_{ab} \cdot \nabla_a W_{ab}}{(\rho_a + \rho_b)(r_{ab}^2 + \eta^2)} \mathbf{v}_{ab} \quad (4.11)$$

Gotoh et al. (2001) added the sub-particle scale (SPS) turbulence effect to the momentum equation to represent the effects of turbulence in their moving particle semi-implicit (MPS) model, yielding

$$\frac{D\mathbf{v}_a}{Dt} = -\frac{1}{\rho} \nabla P + g + \nu_0 \nabla^2 \mathbf{v}_a + \frac{1}{\rho} \nabla \cdot \tau \quad (4.12)$$

where τ is the SPS stress tensor. As an example, τ has an xy component given by

$$\frac{\tau_{ab}}{\rho} = \nu_t \left(2S_{xy} - \frac{2}{3} S_{kk} \delta_{ab} \right) - \frac{2}{3} C_l (dp)^2 \delta_{ab} |S_{xy}|^2 \quad (4.13)$$

where $\nu_t = [C_{sm}(dp)]^2 |S_{xy}|$ is turbulent eddy viscosity. S_{xy} is the strain rate (Eq. 3.3). S_{kk} is the isotropic part of the strain tensor. C_{sm} is the Smagorinsky constant, which is taken as 0.15; $|S_{xy}| = \sqrt{2S_{xy}S_{xy}}$, $C_l = 0.0066$ is a constant; dp is the particle to particle spacing, δ_{ab} is Kronecker delta (Dominguez et al. 2021).

In this study, laminar viscous stress combined with SPS turbulence effect was used to include the effects of dissipation into the model.

4.3 Methods in DualSPHysics

The DualSPHysics code is an open-source SPH model developed by researchers at the Johns Hopkins University in the U.S., the University of Vigo in Spain, the University of Manchester in the U.K. and the University of Rome La Sapienza in Italy. DualSPHysics is developed to study free-surface flow phenomena where the application of mesh-based methods encounters difficulties. DualSPHysics is a set of C++, CUDA, and Java codes designed to deal with fluid dynamics problems. The fifth version of the code was released in 2020. The new version has the

option to create several layers of particles at boundaries. Figure 4.3 illustrates the parts of the DualSPHysics model.

SPH simulations in DualSPHysics include three main steps:

- Generation of the neighbor list,
- Computation of the forces between particles (solving momentum and continuity equations)
- Update of the physical quantities at the next time step.

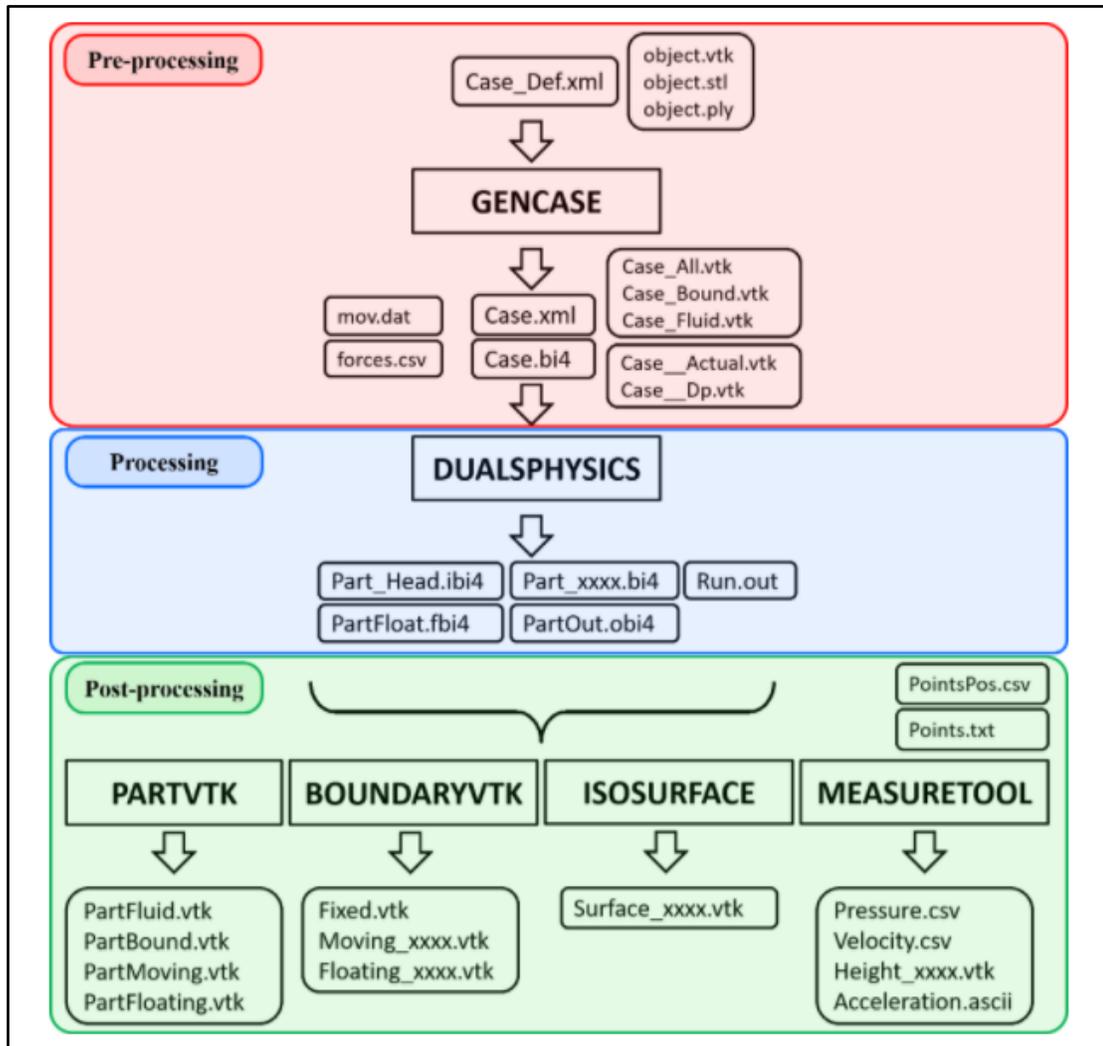


Figure 4.3. Preprocessing, processing, and post-processing stages of DualsSPHysics model (modified from DualsSPHysics manual, <https://github.com/DualSPHysics/DualSPHysics/wiki>)

In Figure 4.3, GenCase employs a 3-D Cartesian mesh to locate particles. The idea is to build any object using particles. These particles are created initially at the nodes of the 3-D Cartesian mesh.

The main code which performs SPH computations is named DualSPHysics. The input files to run DualSPHysics include one XML file and a binary file. The geometry (or flow domain) of the flow problem should be defined in the XML file. The XML file also includes all the system configuration parameters and its execution, such as smoothing length, reference density, gravity, the constant coefficients, the number of particles in the system, and the properties of moving bodies.

The ComputeForces code is employed to compute the force exerted by the fluid onto a boundary object. The forces are calculated as the summation of the acceleration values (by solving the momentum equation) multiplied by the mass of each boundary particle.

The PartVTK code converts the output binary files of DualSPHysics into different formats that can be visualized and analyzed. Thus, the output files of DualSPHysics, the binary files (.bi4), are the input files for the post-processing code called PartVTK.

4.3.1 Boundary treatment

Solid boundaries are treated using the dynamic boundary condition (DBC) method in which the boundary particles satisfy the same equations as fluid particles except that the boundary particles do not move according to the exerted forces, and instead, they were fixed in the position. One advantage of this method is the computational simplicity because the boundary particles' properties are calculated inside the same loops as fluid particles resulting in a considerable saving of computational time. DBC was firstly proposed by Dalrymple and Knio (2001) and used in further studies on the interaction between waves and coastal structures (Gomez-Gesteira et al. 2005; Gomez-Gesteira and Dalrymple 2004; Crespo et al. 2007; Crespo et al. 2015). A modification of DBC (mDBC), presented by English et al. (2021), arranges the boundary particles in the same way as the boundary particles in the original DBC, with the boundary interface located half a particle spacing from the innermost layer of boundary particles.

Other methods of boundary treatment are SPH-DEM coupling (Canelas et al. 2016) and open boundary conditions (Gomez-Gesteira et al. 2012). Open boundary condition (OPC) is used when an inflow velocity needs to be prescribed at an inlet, while other velocity or pressure conditions can be either prescribed or extracted from the fluid domain at an outlet. OPC was firstly presented by Tafuni et al. (2016). Figure 4.4 shows the working principles of the OPC algorithm in a case of a fluid flowing near an open boundary.

In Figure 4.4, the innermost dashed line represents the fluid-buffer interface, followed by a layer of SPH particles used to enforce certain boundary conditions. The position of the ghost nodes (Figure 4.4, green dots) is obtained by mirroring the boundary particles into the fluid along a direction that is normal to the open boundary. A standard particle interpolation would not be suitable to calculate fluid quantities at the ghost nodes due to the lack of full kernel support at boundaries. The buffer width should be equal to or exceed the kernel radius to ensure full kernel support for the fluid particles in the proximity of an inlet or outlet. Buffer regions prevent errors generated by the kernel truncation near the boundaries, and particles inside the buffer areas are created or deleted to prevent the formation of voids. To enforce flow conditions at the boundaries, velocity, and pressure can be assigned to the particles inside the buffer region. In OPC, the physical information of the buffer particles can also be extrapolated from the fluid domain based on ghost nodes located in the fluid domain near the boundary.

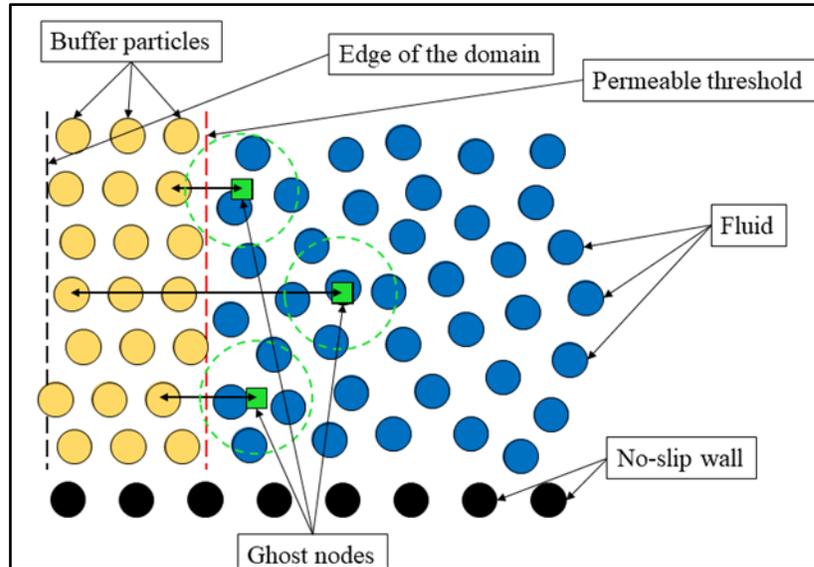


Figure 4.4. Open boundary treatment in DualSPHysics (modified from DualSPHysics manual, <https://github.com/DualSPHysics/DualSPHysics/wiki>)

In this study, the boundary conditions at the inlet for $y < -0.002$ m was $u = U_1$ and $v = 0$; for $y > 0.002$ m was $u = U_2$ and $v = 0$. The density of particles was obtained by extrapolations from the neighboring fluid particles. For more details, refer to Tafuni et al. (2016). The inlet boundary consisted of ten layers of fluid particles. At the outlet, velocity was obtained by extrapolations from the neighboring fluid particles. The outlet boundary consisted of four layers of fluid particles. These layers were buffer layers to allow for extrapolations. At the solid boundaries (sidewalls, channel bottom and splitter plate surfaces), the flow velocity components were $u = 0$ and $v = 0$ (no-slip wall). The dynamic boundary condition was used at the solid boundaries such that the boundary particles satisfied the same equations as fluid particles except that the boundary particles did not move according to the exerted forces and instead, they were fixed in position.

4.3.2 Shifting algorithm

Anisotropic particle spacing is an important stability issue in SPH simulations because in flows with significant shear, particles cannot maintain a uniform distribution. Anisotropic particle spacing produces noise in the velocity and pressure field and causes voids within the water flow (Figure 4.5). Xu et al. (2009) proposed a particle shifting algorithm to prevent the anisotropic particle spacing. The algorithm was firstly created for incompressible SPH simulation. However, it can be extended to the weakly compressible SPH model used in DualSPHysics (Vacondio et al. 2013). Using the shifting algorithm, the particles are moved (or “shifted”) towards areas with fewer particles (lower particle concentration), allowing the domain to maintain a uniform particle distribution and eliminating any voids that may occur due to the noise.

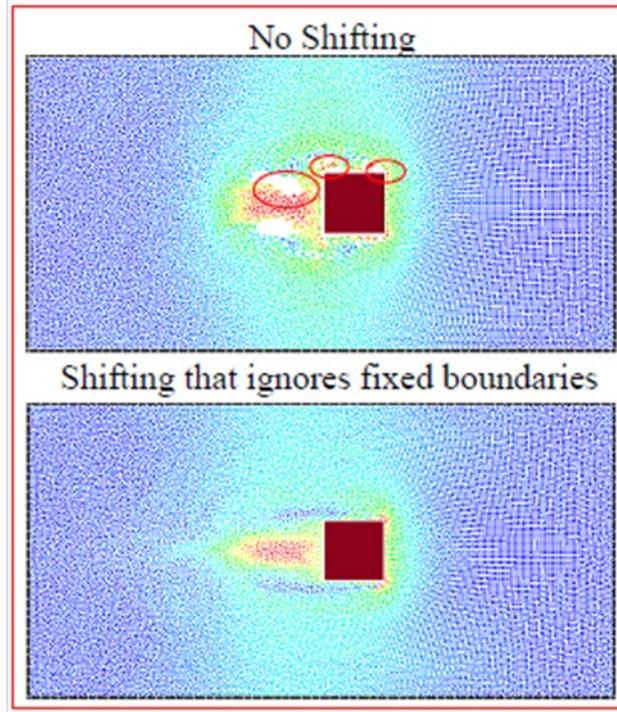


Figure 4.5. Shifting algorithm to avoid particle agglomeration is a SPH simulation(modified from DualSPHysics manual, <https://github.com/DualSPHysics/DualSPHysics/wiki>)

4.3.3 Specifications of model parameters

This section presents the SPH model specifications used to simulate SML using DualSPHysics. SPH runs include three velocity ratios: $V_r = 1.14, 1.5,$ and 3 (Table 4.1). The first two cases matched the PTV experiments discussed in Chapter 3 of this thesis. The third case with $V_r = 3$ allowed us to examine the effect of the doubling velocity ratio on flow structures. Table 4.1 presents a summary of the hydraulic conditions of the runs. Figure 4.6b illustrates the geometry of the SPH model domain. The model domain had segment lengths $L = 1\text{m}$; a width $b = 0.8\text{ m}$, and a depth of flow $d = 0.064\text{ m}$.

The Froude number is given by $Fr = U/\sqrt{gd}$, where U is cross-sectionally averaged streamwise velocity. The velocities of two parallel streams were chosen to maintain a subcritical flow during the simulations. The Reynolds number is defined as $Re = Ud/\nu$, where ν is the kinematic viscosity of water. The Reynolds number was high enough to assure a turbulent flow at both sides of the splitter plate. The runs covered a simulation time period of 7s ; this is long enough because fluid particles initially entering at the upstream inlet have advected to the downstream end of the model domain.

Table 4.1. Hydraulic conditions for SPH runs of shallow mixing layer. The subscripts 1 and 2 refer to the low and high-velocity sides, respectively (Figure 4.6)

Run ID	V_r	U_1 (m/s)	U_2 (m/s)	(Fr ₁ , Fr ₂)	(Re ₁ , Re ₂)
A	1.14	0.377	0.43	(0.47, 0.54)	(24505, 27950)
B	1.5	0.3	0.45	(0.38, 0.56)	(19500, 29250)
C	3	0.187	0.561	(0.23, 0.70)	(12155, 36465)

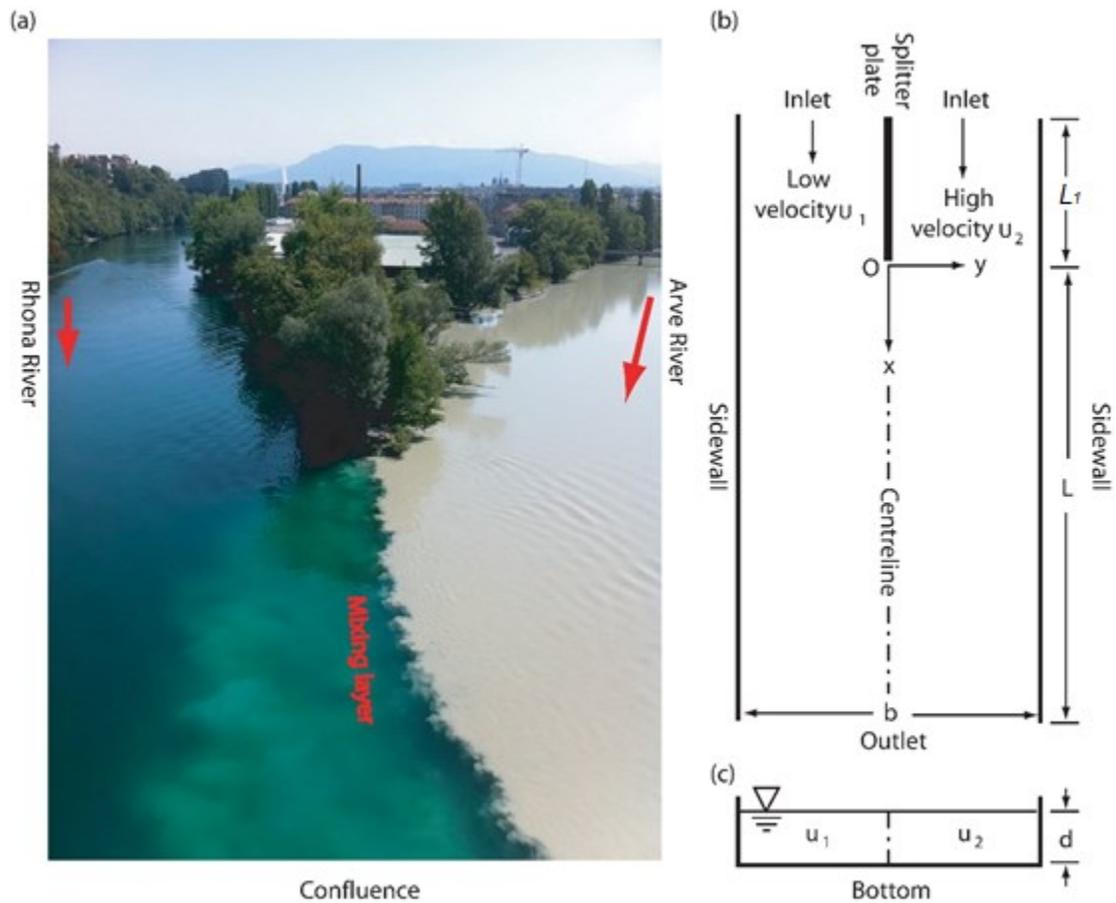


Figure 4.6. (a) Mixing layer at the confluence of the rivers Rhona and Arve in Geneva, Switzerland (modified from World Geography); (b) plane view of a definition diagram of a shallow mixing layer, where the tip of the splitter plate is at point O; (c) elevation view of a channel cross-section

Particle spacing and smoothing length need to be specified. We tested the suitability of particle spacing $dp = 0.005$ m and found that SPH results with $dp = 0.005$ m did not give details about SML flow structures. Therefore, we set the particle spacing to 0.002 m; initially, the total number of particles was 221,518 distributed in the model domain. Wu et al. (2016) tested three different particle spacing ($dp = 0.004, 0.005, 0.006$) m in a dam break application. The results with $dp = 0.004$ m produced the best flow profile and captured the flow structures; it incurred higher computational costs. Therefore, it is necessary to consider a balance between numerical accuracy and computational costs, which is situationally dependent. In this study, we used $dp = 0.002$ m in all three runs.

The smoothing length, h , is a very important parameter in SPH simulations, having effects on both the accuracy of the solution and the efficiency of computations. If h is too small, there may not be enough particles in the support domain of dimension Kh to exert forces on a given particle, which results in a low accuracy. K is a scaling factor equal to 2 (Figure 4.2). If h is too large, all details of local properties may be smoothed out. The particle approximation used in the SPH method depends on having a sufficient and necessary number of particles within the support domain of Kh .

In early implementations of SPH, a global particle smoothing length was used, which depended on the initial average density of particles in the simulation domain. In later implementations, a varying smoothing length with respect to time and space was proposed by Monaghan (1992). Based on this approach, each individual particle is assigned by a smoothing length according to the variation of the local density of the particle

$$\frac{dh}{dt} = -\frac{1}{D} \frac{h}{\rho} \frac{d\rho}{dt} \quad (4.14)$$

where D is the number of dimensions (1D, 2D or 3D). In this study, $D = 2$. Using this formula, the smoothing length became updated so that the number of the neighboring particles remained relatively constant. Liu and Liu (2003) stated that the sufficient number of particles for 2D simulation is 21. Fraga Filho (2019) proposed the same criteria for the smoothing length. Considering an initial smoothing length equal to $1.5dp$ and $K = 2$ in this study, we were able to maintain 21 particles in the supporting domain of initial particles.

A varying smoothing length in time and space means that each particle has its own smoothing length. If h_i is not equal to h_j , the influencing domain of particle P_i may cover particle P_j but not necessarily vice versa. Therefore, particle P_i may exert a force to particle P_j without particle P_j exerting the same reaction on P_i . This is a violation of Newton's third law. In order to overcome this issue, SPH uses the arithmetic mean of smoothing lengths of the pair of interacting particles

$$h_{ij} = \frac{h_i + h_j}{2} \quad (4.15)$$

Wu et al. (2016) evaluated the effect of smoothing length, h , on the flow patterns in a dam break. They showed that the flow patterns are more accurate when the smoothing length is $h = 1.25dp$ to $1.5dp$. They showed that the smoothing length equal to $1.5dp$ means more surrounding particles, and therefore, more accurately represents the fluid profile (Figure 4.7).

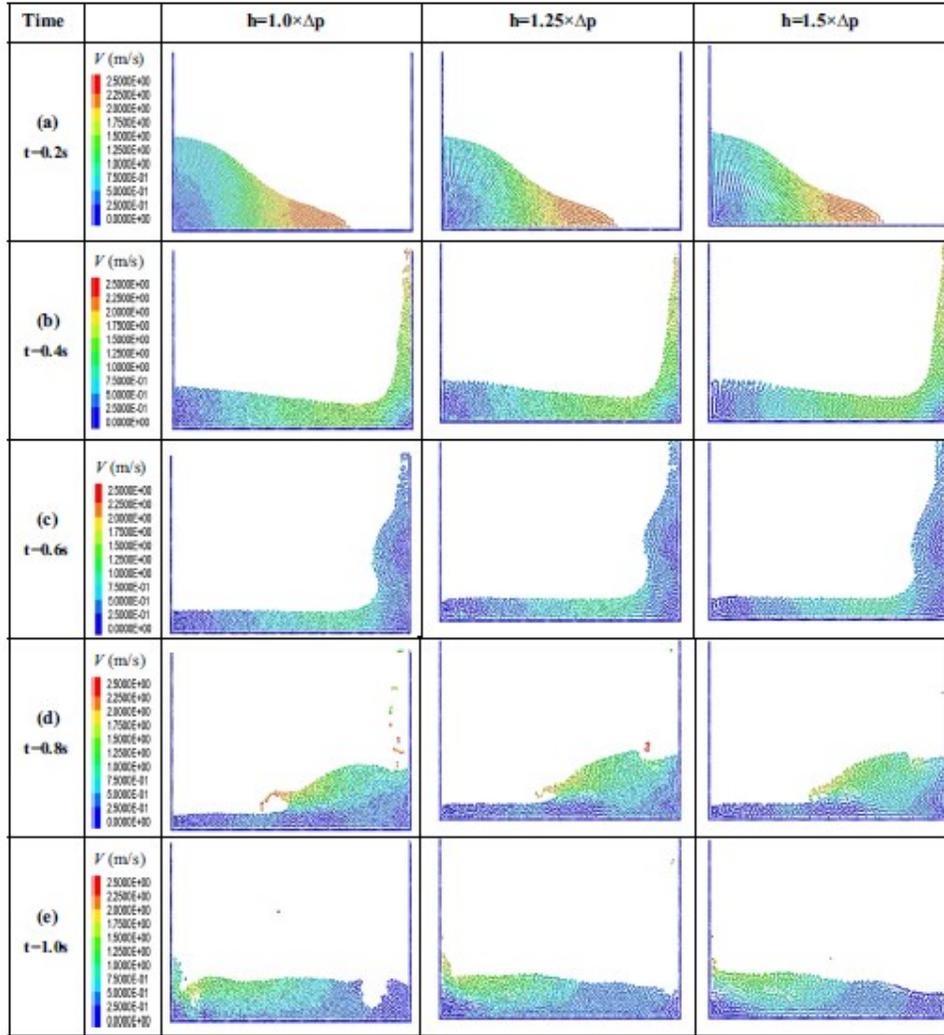


Figure 4.7. Effect of smoothing length on flow patterns in a dam break, at different time steps (from Wu et al. 2016)

With regard to Courant–Friedrichs–Lewy (CFL) stability, we ensured a CFL number of 0.2. The computational time step, Δt , was automatically adjusted to follow the CFL stability requirement (Gomez-Gesteira et al. 2012). The CFL condition can lead to very small time steps, making a SPH simulation a very computationally demanding method. In this study, we set the minimum time step to 1.33×10^{-5} s.

About the Smogorinsky Constant, C_{sm} , in classical examples of SPH simulations, C_{sm} has a value between 0.1 and 0.2. Gabreil et al. (2018) used a range of C_{sm} values (0.6–3.5) to find the best fit of simulated velocity profiles to experimental data. Values of C_{sm} between 0.6 and 3.5 were found to cause turbulent shear stresses significantly smaller than measured values. In this study, we selected $C_{sm} = 0.15$ for the runs of this study.

Regarding the density diffusion term, Molteni and Colagrossi (2009) introduced density diffusion to improve pressure evaluation in SPH simulations. Using the density diffusion term, the

numerical high-frequency noise in the pressure field is automatically removed. In the current study density diffusion option was used.

The quintic function (Wendland, 1995) was selected as the interpolation kernel function (Figure 4.2) in the SPH simulations of this study

$$W(r_{ab}, h) = \alpha_D \left(1 - \frac{q}{2}\right)^4 (2q + 1) \quad 0 < q < 2 \quad (4.16)$$

where α_D is equal to $7/4\pi h^2$ in 2-D simulation; q is the non-dimensional distance between particles given by $q = r_{ab}/h$. Macia et al. (2011) showed that Wendland kernels reproduce turbulence dissipation mechanisms more accurately than those with a renormalized Gaussian kernel in free water surface flow simulations.

In DualSPHysics, two explicit time integration schemes are available: a verlet based scheme, and a symplectic scheme. In this study, the symplectic method was employed for time integration. In this method, the particle density ρ_i is calculated in half-step as

$$\rho_i^{n+1/2} = \rho_i^n + \frac{\Delta t}{2} R_i^n \quad (4.17)$$

$$\rho_i^{n+1} = \rho_i^n \frac{2 - \varepsilon_i^{n+1/2}}{2 + \varepsilon_i^{n+1/2}} \quad (4.18)$$

where $R_i = d\rho_i/dt$; $\varepsilon_i^{n+1/2} = -(R_i^{n+1/2}/\rho_i^{n+1/2})\Delta t$. Note that the verlet method computes the density evolution at time step $n+1$ directly from step n , whereas the symplectic method determines the density evolution in half timesteps.

In order to homogenize the particle distribution, Adami and Adams (2012) introduced a modified transport velocity to advect particles. DualSPHysics uses this approach for stabilizing the numerical scheme with strongly reduced artificial numerical dissipation. To avoid a computationally expensive Poisson solver for the pressure field, DualSPHysics utilizes an equation-of-state that relates the density of water to pressure, P , which treats water as a weakly compressible fluid

$$P = \frac{c_s^2 \rho_0}{\xi} \left[\left(\frac{\rho}{\rho_0}\right)^\xi - 1 \right] \quad (4.19)$$

where ξ is the polytropic index, usually equal to 7 for water; c_s is the speed of sound. The actual speed of sound in water is about 1500 m/s. On the basis of previous studies, c_s in Equation 4.19 was given a value smaller than the actual value; it should be chosen as at least $10U_{max}$ to assure a particle density variation of less than 1%. The use of the actual speed of sound would require a prohibitively small time-step (Crespo et al. 2015). The speed of sound in this study was set to 15 m/s.

4.4 Results

This section analyses the SPH simulation outputs and presented the results. Transverse profiles of the streamwise velocity and the effect of wake on the SML development are discussed. The

discussion covers flow structures, and their temporal and spatial evolutions. One important environmental aspect of SML is the exchange of mass and momentum. This section addresses the effect of velocity ratio on the fluid mass and momentum exchange in the SML and compares the results between this study and previous studies of SML.

4.4.1 Effect of wake

Transverse profiles of the streamwise velocity at $x = 0.1, 0.25, 0.5$ and 0.95 m from the tip of the splitter plate for Run A, B, and C are shown in Figure 4.8. The reason for selecting these x locations was to observe the evolution of transverse profiles of velocity from the near field to the far field. The discussion of profile shapes involves the concept of mixing layer width, δ . In this study, δ is defined using the boundary layer theory. Based on this definition, the mixing layer edge at the low-velocity side is located where the streamwise velocity is equal to $u = 0.99U_1$, and at the high-velocity side, the mixing layer edge is located where the velocity is equal to $u = 0.99U_2$.

The velocity ratio $V_r = 1.14$ results in a smaller value of δ for Run A, compared to that for Run B (Figure 4.8b). The mixing of fluid masses for Run C is subject to the influence of a strong velocity gradient. The result was a larger mixing layer width, compared to those for runs A and B.

It can be seen that the mixing layer width increased with distance toward downstream for all three runs (Figure 4.8). Regardless of the velocity ratio, a velocity deficit due to the effect of wake caused by the splitter plate can be observed at $x/d = 1.5$, and the velocity deficit became less profound in the downstream direction. The velocity deficit is defined as the velocity values smaller than the mean streamwise velocity of the low-speed stream. The velocity deficit is more visible for Run A, where the velocity ratio is lower (Figure 4.8a) than those for runs B and C (Figures 4.8b,c). One explanation would be the larger value of $\partial u/\partial y$ for the latter runs, which dominates the velocity deficit at $x = 0.1$ and 0.25 m.

Due to the effect of wake, the self-similarity of transverse profiles of the streamwise velocity reported by Uijtewaal and Booij (2000) did not appear at $x = 0.1$ m, irrespective of the velocity ratio between the two parallel streams. For Run A, there was no self-similarity at $x < 0.95$ m. Townsend (1980) stated that the velocity profile can be asymmetric around the channel centerline if the ratio $\Delta U/U_1$ is very small, where $\Delta U = U_1 - U_2$. For the conditions investigated in this study, the ratio $\Delta U/U_1$ was 0.05, 0.1, and 0.125 for Runs A, B, and C, respectively.

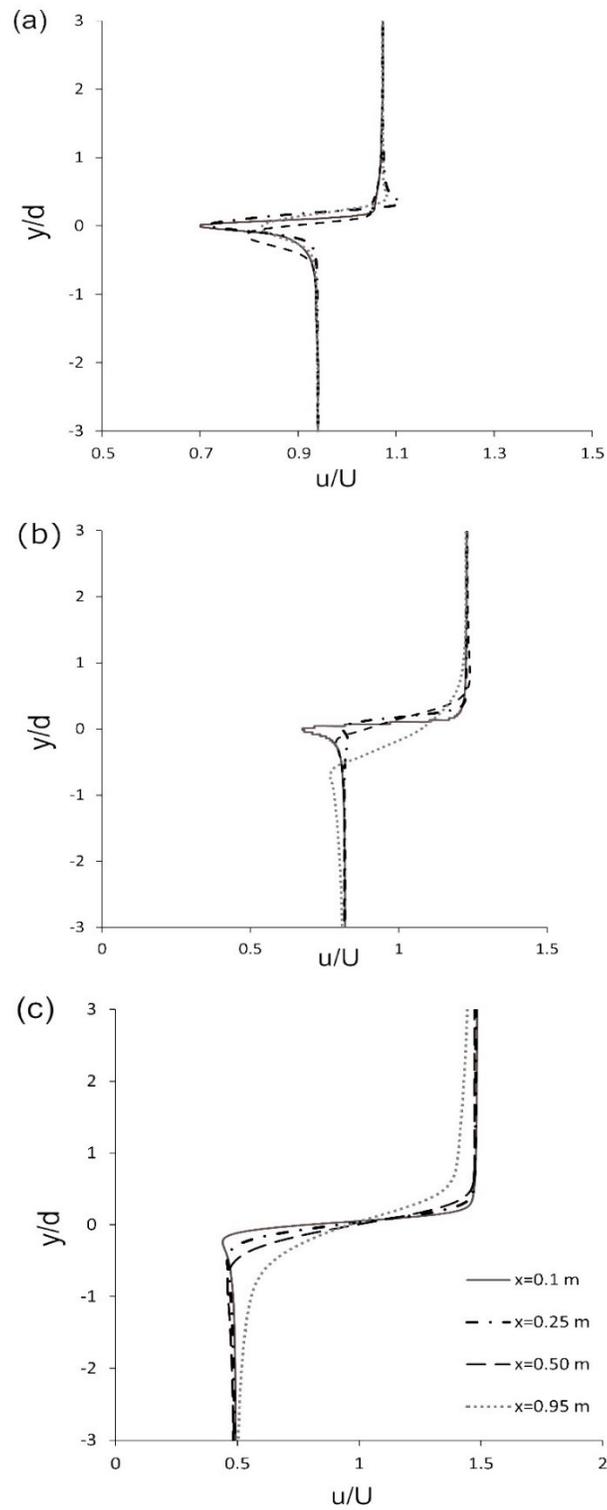


Figure 4.8. Transverse profiles of the streamwise velocity at selected streamwise distances from the tip of splitter plate for: a) Run A with $V_r = 1.14$; b) Run B with $V_r = 1.5$; c) Run C with $V_r = 3$. The model time is $t = 7$ s

4.4.2 Temporal and spatial evolutions of flow structures

Let ω denote vorticity, which is mathematically expressed by

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \quad (4.20)$$

Figure 4.9 provides the vorticity magnitude at model time $t = 7$ s for Run A, B, and C. Such results of vorticity field for different velocity ratios have not been reported in the literature. The vorticity field reveals coherent structures and the process of their formation. Figure 4.9a represents the vorticity magnitude when V_r is as low as 1.14. The vortices decayed in shorter distances, compared to the case of $V_r = 1.5$. Pairing activities of these vortices occur less often for Run A, compared to those for Run B. The strongest eddies are seen for Run C, and this case gives a complete image of fluid mass mixing processes. Although for Runs A and B the pairing process gradually discontinues downstream of $x/d = 6-7.5$, it can be seen that the pairing process continues over the whole length of the model domain for Run C.

Figure 4.9 shows that the instabilities appear immediately at the tip of the splitter plate. In a two-dimensional mixing layer, eddies are advected by the mean streamwise velocity. Hence, the crest of the perturbed interface on the high-velocity side moves temporally faster than the trough of the perturbed interface on the low-velocity side, as illustrated in Figure 4.10. This phenomenon increases the width of the vortex street and later transforms the sheet into a spiral. The model time is $t = 7$ s.

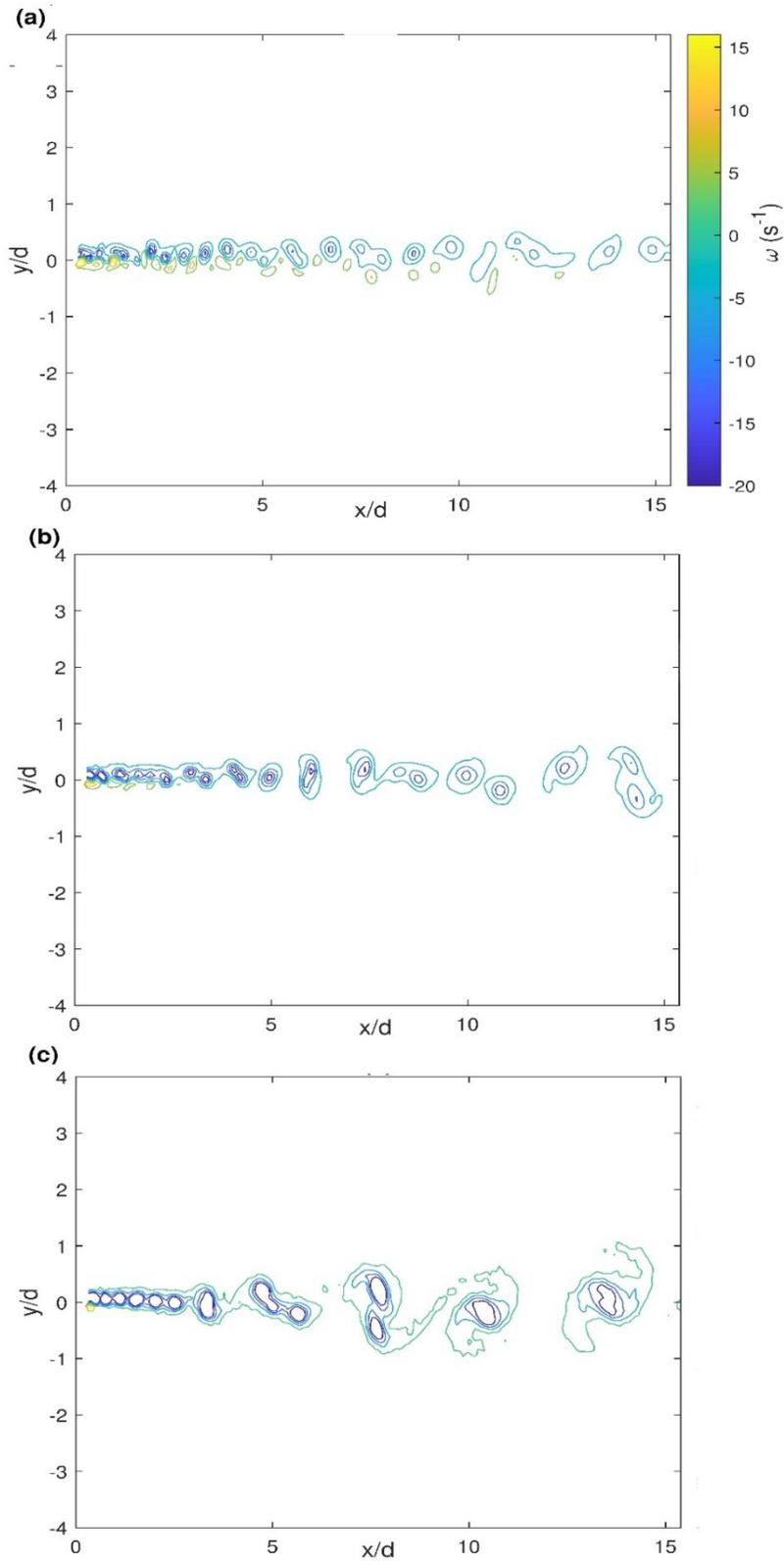


Figure 4.9. Vorticity contour for: a) Run A; b) Run B; c) Run C. The model time is $t = 7$ s

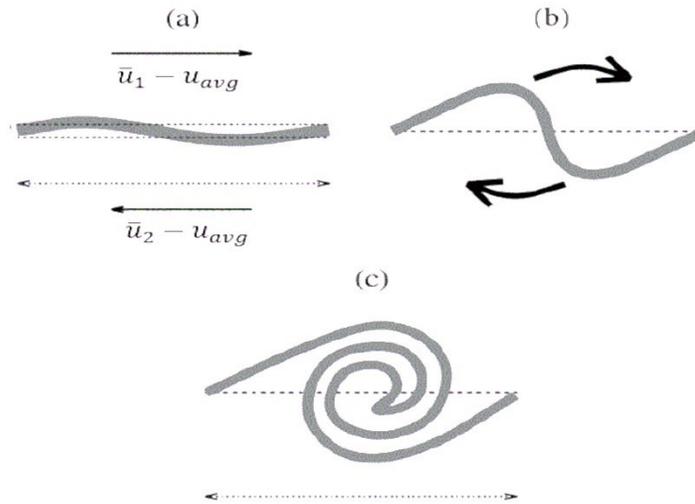


Figure 4.10. Schematic illustration of the roll-up process in mixing layer (from Lesieur 2008)

In Figure 4.11, the flow structures formed between the two streams for Run A, are displayed by individual fluid particles. The length scale of the unstable flow features increased with downstream distance. The roll-up occurred right at the tip of the splitter plate as soon as the two streams came to contact. During this stage, a dominant 'wave' appeared as a temporally fastest-growing perturbation, which inhibited the growth of perturbations outside the mixing region. It can be seen that when the velocity ratio was small ($V_r = 1.14$), the pairing process did not become complete since ω was small.

The Euclidean distance, d_{ij} , from the center of a random eddy i to its nearest neighboring eddy j is measured in consecutive time steps. For the two selected eddies, the mean vorticity magnitude was calculated. For Run A, the results show that pairing did not occur even though the dimensionless distance between the two eddies d_{ij}/d reached values as small as 0.155. This highlights the impact of vorticity magnitude in pairing formation. For Run A, the average vorticity value for the two eddies, ω_{ij} , ranged from -5 s^{-1} to 19 s^{-1} , the value of 19 s^{-1} being for two eddies with $d_{ij} = 0.326$.

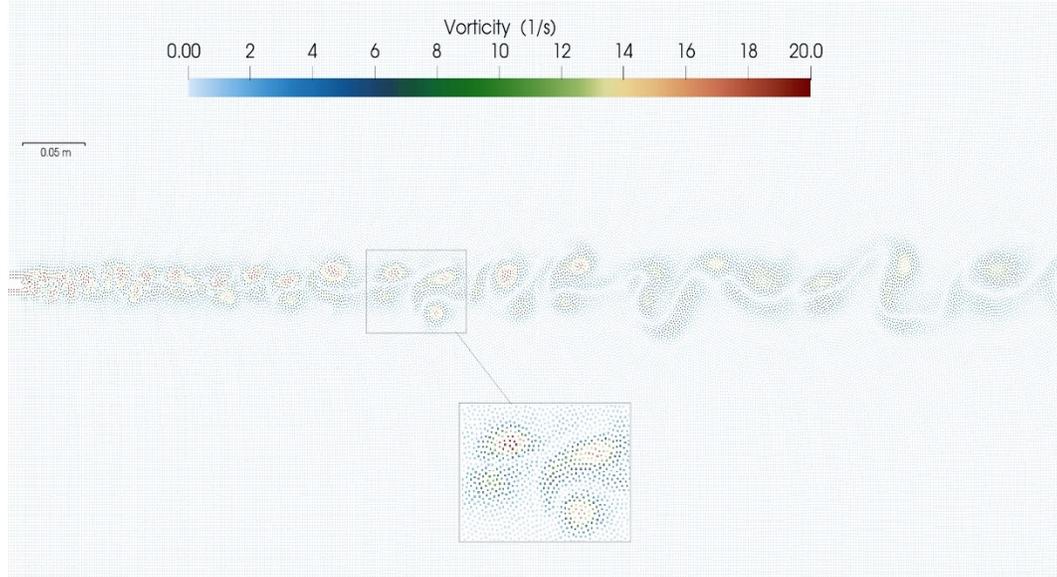


Figure 4.11. Particle vorticity field at $x < 0.3$ m for Run A

The same analysis for Runs B and C proves the impact of ω_{ij} on the pairing process. For Run B, ω_{ij} ranged from 20 to 43.5 s^{-1} while d_{ij}/d varied from 0.155 to 0.463. With an increasing velocity gradient between the two streams for Run C, the values of ω_{ij} increased, ranging from 27.5 to 64.5 s^{-1} . It is noteworthy that the range of d_{ij}/d for Run C increased due to larger sizes of eddies, comparing to those for Runs A and B. The results mentioned above show that the pairing process is affected mainly by the vorticity of two eddies rather than their relative distance. For Run A, although the eddies became close to each other, they did not necessarily pair. Thus, results cannot be obtained from Eulerian simulations, which highlights the importance of the application of mesh-free methods in creating new insights in studying shallow mixing layer.

One example of the pairing process for Run B at time steps 61 to 64 (model time $t = 3.05$ to 3.2 s) is shown in Figure 4.12. Two vortices are selected and tracked in a sequence of time frames. The figure shows that one larger eddy entangles with a neighbour eddy while being advected downstream, resulting in a larger eddy. The SPH simulations capture the pairing process from $x/d = 4.5$ – 5 . The dimensionless duration, $t \times \frac{\Delta U}{d}$, of the pairing process for the vortices in Figure 4.6 was as short as 0.57, and the paired vortex decayed with decreasing vorticity downstream.

Figure 4.12e shows a frame captured from dye visualization experiments where the velocity ratio between the two streams equals 1.5. All the hydraulic conditions, including the flow depth, Fround number and Reynolds number, are the same as the SPH simulations ($V_r = 1.5$). More details of the dye visualization experiment are presented in Section 3.7 of this thesis. Figure 4.12e shows a pairing process for two selected eddies at $x/d = 9$. The experiments show such pairing process aspects as rotating the core of two neighboring eddies around each other and advecting downstream, which support the computed results from SPH. The pairing process in $x/d < 9$ in the dye visualization experiments is not recognizable due to the small size of the eddies and a lack of color spectrum to distinct eddy cores. Considering the oval shape of eddies (Figure 4.12), we examined the flow patterns in the region bounded by $x/d = 9.2$ and $x/d = 12.3$ and determined the

dimensions of eddies. The minor and major axes of the oval shapes (eddies) were $0.8d$ and $1.4d$, respectively from the experiments, compared to $1.1d$ and $1.4d$ from the SPH computations.

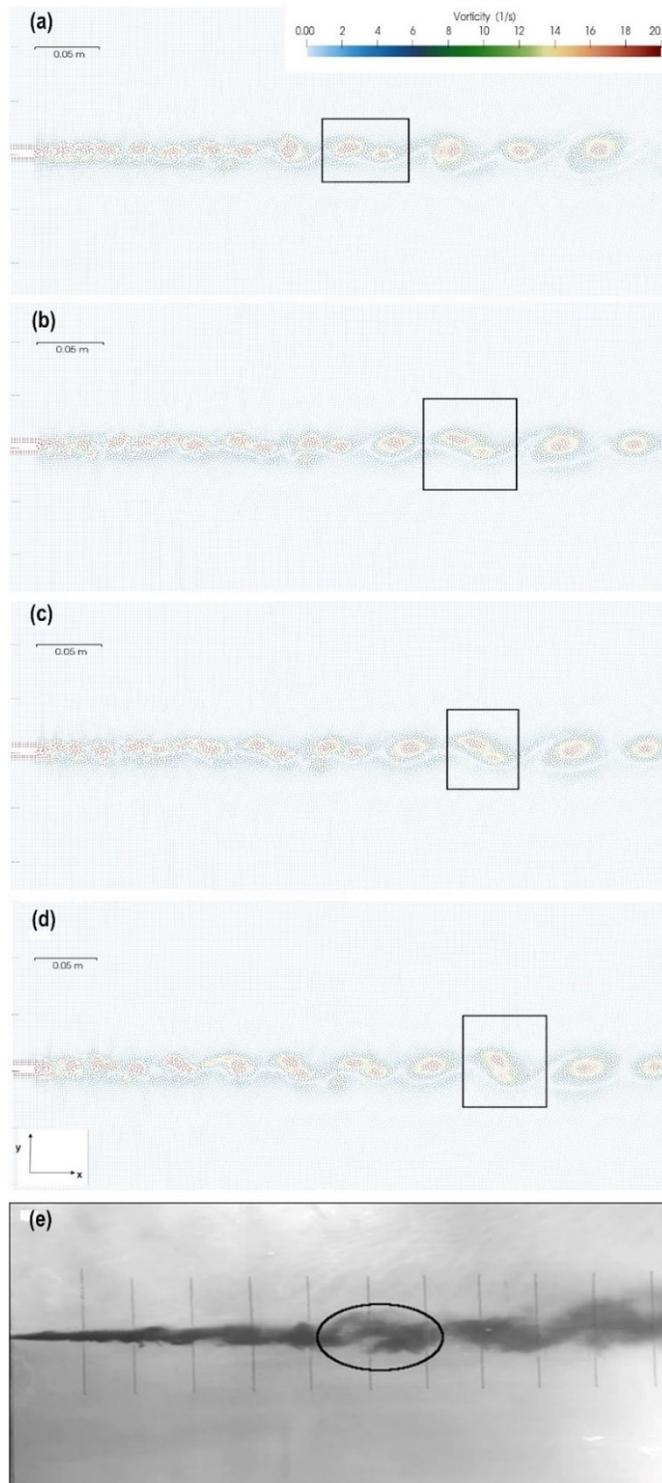


Figure 4.12. The pairing process for two selected vortices at $V_r = 1.5$ from $t = 3.05$ s (panel a) to $t = 3.2$ s (panel d), and one example frame of the dye visualization experiment (panel e)

For Run C, the pairing process at time steps 120 to 131 (model time $t = 6$ to 6.5 s) is presented in Figure 4.13. It can be seen that the pairing process occurred in a larger flow area, compared to that for Run B shown in Figure 4.12. The formation of an entangling process for the first three time steps was similar to that for Run B but was slower as the dimensionless duration was equal to 1.73 for Run C and 0.57 for Run B. In Figure 4.13, the paired eddy contacted a neighbor eddy at $t = 6.3$ s, and a new pairing process occurred. Examining the evolution of eddies from the tip ($x = 0$) of the splitter plate up to $x = 15.4d$ for Runs B and C would help understand the processes involved in the near-field and far-field regions. Figures 4.12 and 4.9 show that the pairing of eddies was the primary process that enlarged the horizontal dimension of coherent structures to be larger than the flow depth. Jirka (2001) mentioned a similar criterion for the transition to the far-field region where the horizontal dimension of two-dimensional coherent structures increases to values larger than the flow depth. The proposed criterion by Jirka (2001) is based on the mean flow field and the growth rate of the mixing layer. This study shows that the pairing process could be considered the criterion for demarcating the transition from near-field to far-field.

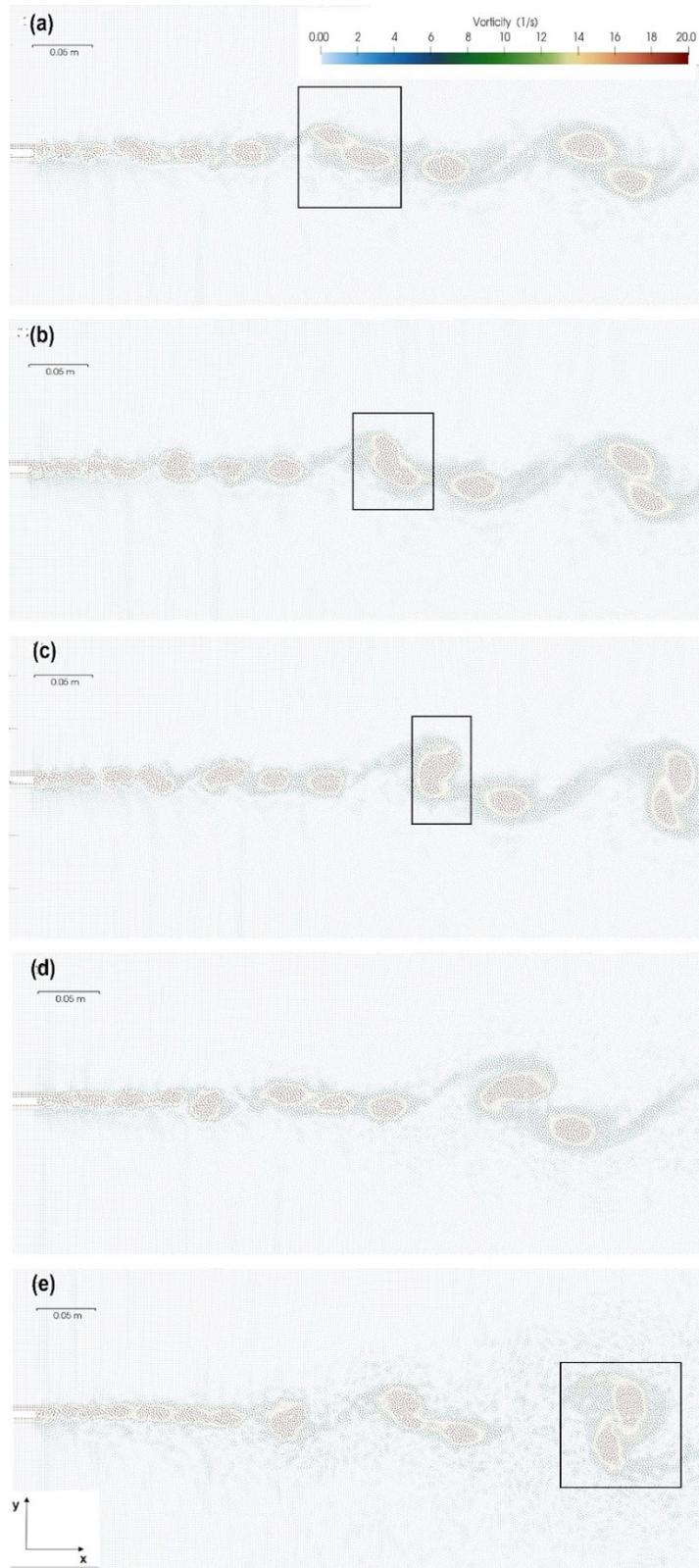


Figure 4.13. The pairing process for Run C at $t = 6$ s (panel a) to $t = 6.5$ s (panel e)

The Okubo-Weiss parameter, W , was introduced by Okubo (1970) and Weiss (1991) to measure the relative importance of deformation and rotation. The Okubo-Weiss parameter is defined as

$$\Omega = S_n^2 + S_s^2 - \omega^2 \quad (4.21)$$

where S_n and S_s are the normal and shear stresses, respectively

$$S_n = \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}, S_s = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \quad (4.22)$$

The Okubo-Weiss parameter has the advantage of being entirely a local parameter. Figure 4.14 shows the contours of W in the computational domain at $t = 7$ s. The blue contours represent the dominance of vorticity over the deformation right downstream of the splitter plate for Run A. This dominance decreased moving downstream, and detached eddies were visible from $x/d = 10$. It can be seen from Figure 4.14 that the general form of an eddy consisted of an inner vorticity-dominated region at the core and an outer region which was strain-dominated (Figure 4.14, yellow color) and surrounding the inner region. The strain-dominated boundary of the eddies performed such a barrier for the particles inside the eddy until the eddies decayed or paired with other eddies. This observation was more visible in Figure 4.14c for Run C than Run A and B.

A decrease in V_r value from 1.5 for Run B to 1.14 for Run A (a 35% decrease) caused a limit on the vorticity-dominated eddies up to $x/d = 4.6$. For Run A, the velocity gradient, which is the nourishing factor for vortical motions, declined at a short distance from the splitter plate, resulting in limited vorticity-dominated eddies. For Run C, an increase in the velocity ratio to 3 impacted the whole length of the computational domain up to $x/d = 15$, as displayed with vorticity-dominated eddies in the central region of the mixing layer. The larger velocity gradient increased the size of these eddies and the extent of the area they occurred.

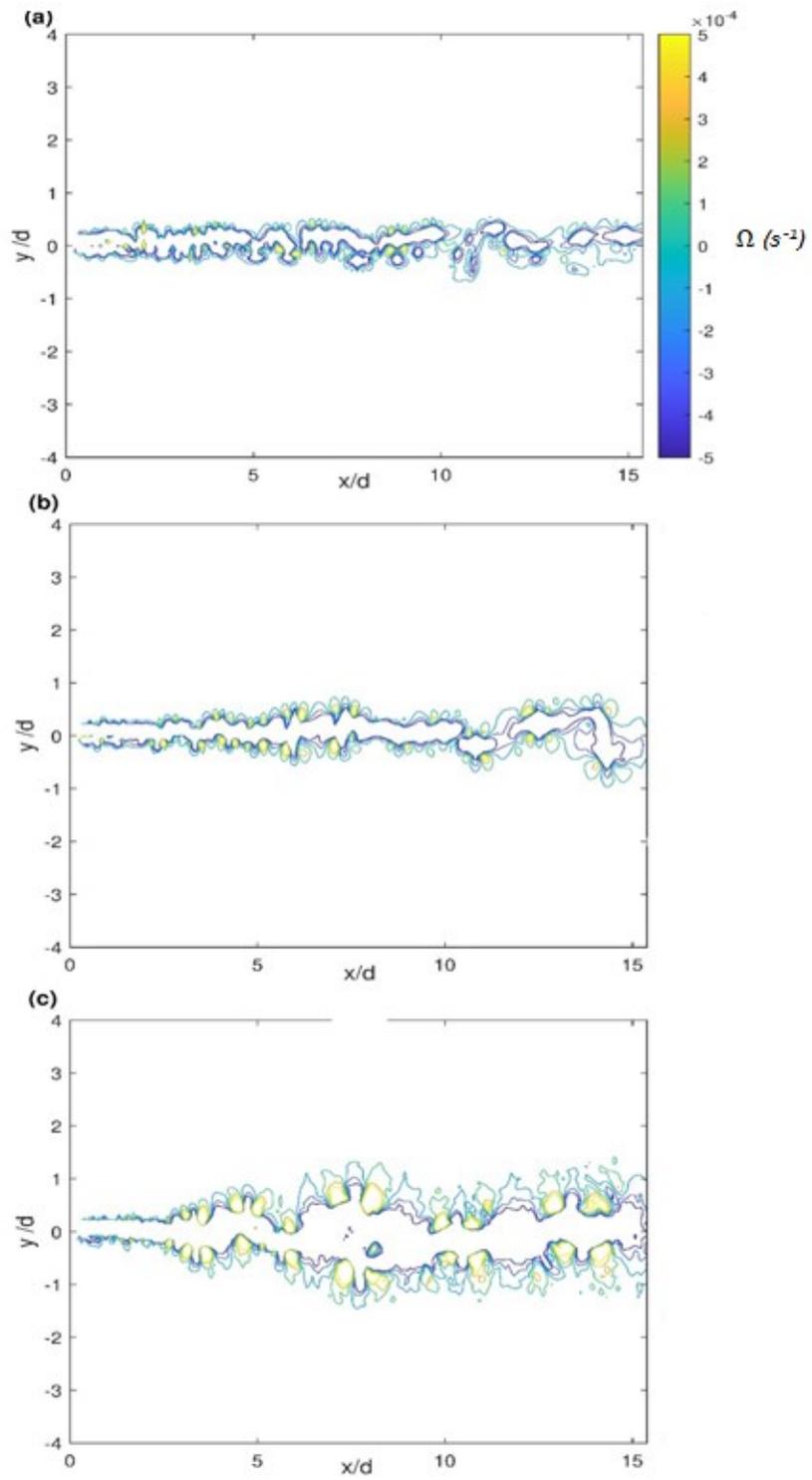


Figure 4.14. Okubo-Weiss parameter contours for three different velocity ratios: a) Run A; b) Run B; c) Run C. The model time is $t = 7$ s

4.4.3 Exchange of fluid mass and momentum

In Figure 4.15, the mass transfer across the channel centerline ($y = 0$) is shown for Run C at model time $t = 6.5$ s, corresponding to the time of Figure 4.13e. As can be seen in Figure 4.15, there was an increase in mass transfer in the channel centerline from upstream to downstream. In the near-field, from the splitter plate up to $x/d \approx 3$, the peak of mass transfer rate was maintained at 0.1 kg/s.m from both sides of the flow, which resulted in the overall mass transfer of 6.89 kg/s.m from the high-velocity stream and 6.40 kg/s.m from the low-velocity stream for $x/d < 3$. The mass transfer peak value reached 0.2 kg/s.m for $x/d > 3$ and again increased to 0.3 kg/s.m after $x/d \approx 6$. These rates can be used to scale up for field conditions (Figure 4.6a). About 80% of mass transfer occurred at $x/d > 3$. The pairing process (Figure 4.13) plays an important role in mass transfer. In the model domain at the specific time step, an overall mass of 98.07 kg/s.m per unit depth from the high-velocity side and 66.05 kg/s.m per unit depth from the low velocity side were transferred toward the opposite side. The unbalanced mass transfer from two streams is because of non-symmetric large-scale eddies (Figure 4.13).

For Run B, the mass transfer was almost equal from high-velocity and low-velocity sides, being 29.75 and 30.8 kg/s.m, respectively. The peak of mass transfer across the channel from the tip of the splitter plate to the end of the computational model domain was constant in contrast to the case for Run C, where the peaks increased from upstream to downstream. This can be explained by the lower velocity gradient between the two streams, which led to smaller eddies and a less energetic pairing process for Run B (Figure 4.12), compared to Run C.

When the velocity gradient was small (Run A), the fluid mass transfer for model time $t = 7$ s (matching the timing of Figure 4.11) was as small as 19.5 kg/s.m and was equal for both sides of the flow. No noticeable change in the fluid mass transfer across the channel occurred from upstream to downstream. This is in agreement with Figure 4.11, where weak eddies and no pairing process were seen.

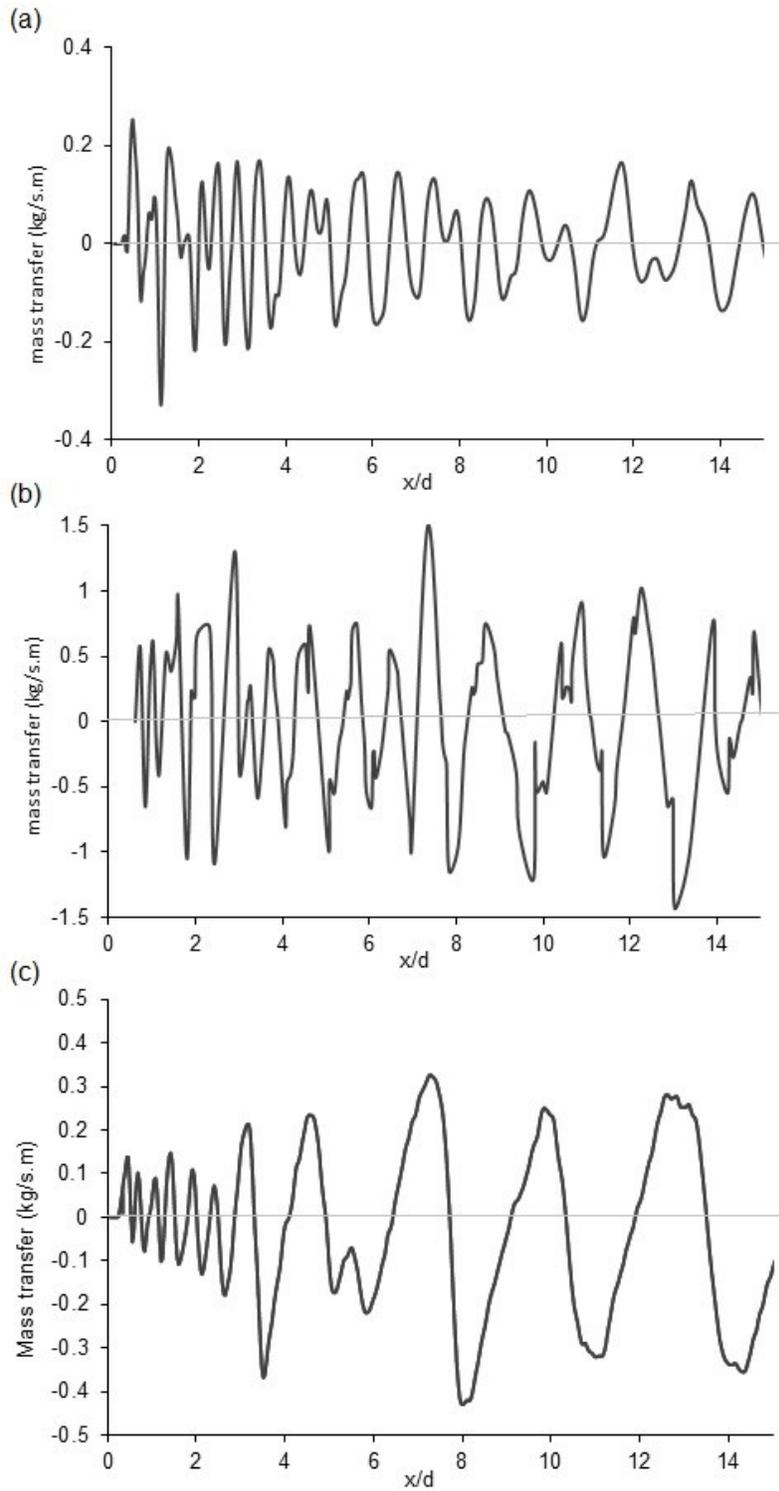


Figure 4.15. Fluid mass transfer across the channel centerline ($y = 0$) for: (a) Run A; (b) Run B; (c) Run C

As shown in Figure 4.16, the momentum transfer for Run C was mainly maintained by the high-velocity stream. The intermittent crests and troughs indicate the evolution of eddies. The larger eddies downstream (Figure 4.13) were responsible for larger momentum transfer. Overall, the momentum transfer at this specific model time was equal to 12.43 and 6.89 kg/s² from the high-velocity and low-velocity sides, respectively.

The momentum transfer for Run B showed almost equal exchange from high-velocity and low-velocity sides, being 1.23 and 1.20 kg/s², respectively. For Run A, the small velocity gradient caused a lower momentum transfer between the two streams, being 0.59 and 0.54 kg/s² from the high-velocity and the low-velocity streams, respectively. The above-mentioned mass and momentum exchanges have great implications to pollutant transfer and the design of drinking-water intakes. The results show that at a high-velocity ratio, pollutants on the fast flow side are more likely to spread to the low-velocity side but not vice versa. This result has implications for the selection of the industrial sites (e.g., chemical plants should be built on the slow flow side) and selective withdrawal of drinking water from a river confluence.

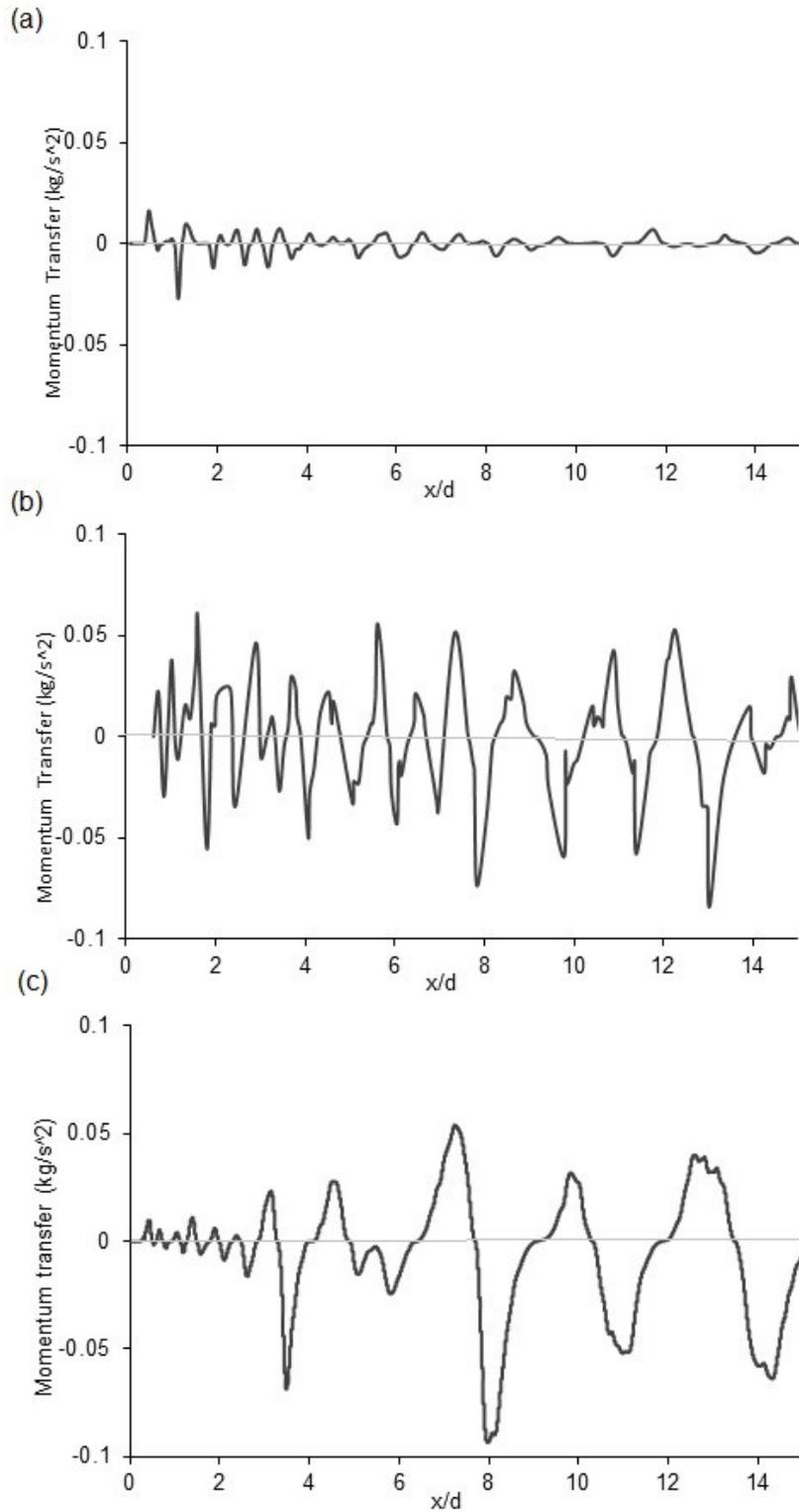


Figure 4.16. Fluid momentum transfer across the channel centerline ($y = 0$) for: (a) Run A; (b) Run B; (c) Run C Data comparison

Chu and Babarutsi (1988) expressed the width of SML as

$$\delta = \frac{\Delta U}{(\partial u / \partial y)_{max}} \quad (4.23)$$

where $\Delta U = U_2 - U_1$. van Prooijen and Uijttewaal (2002) used a similar definition for the width of SML; they used the lateral gradient of the streamwise velocity in the center of the mixing layer $\partial u / \partial y_c$ instead of $(\partial u / \partial y)_{max}$, giving

$$\delta = \frac{\Delta U}{(\partial u / \partial y)_c} \quad (4.24)$$

A comparison of mixing layer width determined by the boundary layer definition and the previous studies is made, and the results are presented in Table 4.2. In this table, the previous studies underestimate the mixing layer width, having a significant difference from the boundary layer method. One possible reason is that the previous two studies do not consider the effect of wake on velocity profiles. The boundary layer growing along each of the two side surfaces of the splitter plate prior to entering the confluence plays an important role in the development of the mixing layer in the confluence. This is especially true when the velocity gradient between the two streams is very small (Run A) due to the relative importance of wake versus SML.

Table 4.2. Comparison of mixing layer width determined using boundary layer definition (this study) and methods based on maximum velocity gradient (previous studies). The superscript a indicates four values for $x = 0.1, 0.25, 0.5,$ and 0.85 m, respectively

Run ID	δ (m) Boundary Layer method	δ (m) Chu and Babarutsi (1988) van Prooijen and Uijttewaal (2002)
A	(0.040, 0.055, 0.071, 0.088) ^a	(0.004, 0.006, 0.007, 0.036)
B	(0.066, 0.055, 0.070, 0.121)	(0.010, 0.019, 0.038, 0.047)
C	(0.107, 0.154, 0.170, 0.175)	(0.019, 0.040, 0.051, 0.069)

Previously, researchers have proposed some empirical relations for the mixing layer width as a function of x (van Prooijen and Uijttewaal 2002; Sukhodolov et al. 2010). A comparison of mixing layer width determined by the boundary layer definition and those relations was made, and the results are presented below. van Prooijen and Uijttewaal (2002) proposed a relation for the velocity difference across the shallow mixing layer (Eq. 2.7) and substituted that into the mixing layer growth rate relation and proposed a relation for shallow mixing layer width as a function of x (Eq. 2.8). Sukhodolov et al. (2010) proposed a parabolic relation for mixing layer width growth based on experimental data from a river confluence (Eq. 2.9).

The mixing layer widths based on the boundary layer definition (this study) and from Eqs 2.8 and 2.9 are compared in Figure 4.17. In these equations, α is the spreading rate defined in Eq. 2.4. With $\alpha = 0.11$ and $\delta_o = d$ according to previous studies (Uijttewaal and Booij 2000), the mixing layer width from the boundary layer definition compares well with both Eqs 2.8 and 2.9. For Run

A, values of δ determined using all the three methods have an increasing trend from upstream to downstream. However, Eqs 2.8 and 2.9 produce a very gradual increase of δ (about 15%) from $x/d = 1.5$ to 13.07. The mixing layer width upstream of $x/d = 7.7$ is significantly affected by the boundary layer development, as seen from Figure 4.17. Therefore, the mixing layer width is limited by the boundary layer effect. This effect is more visible at $x/d = 1.5$, which is close to the tip of the splitter plate.

For Run B, values of δ are generally larger than for Run A due to the larger velocity gradient, although for $x/d = 3.8-7.7$, δ has smaller values. This can be because of the intermittency in the mixing layer as has been shown in Figure 4.6. Similar to Run A, at $x/d = 13$, when the wake effect on velocity profiles is insignificant, δ shows a larger value from the boundary layer definition than from the empirical equations.

For Run C, the largest mixing width is seen, due to the largest velocity gradient among the three runs. For this run, the velocity gradient overcame the wake effect (Figure 4.8c) and did not limit the mixing layer growth. For Run C, the boundary layer definition produced larger δ values at each x/d location, compared to Eqs 2.8 and 2.9. The confinement of shallow flow shows its effect at $x/d = 13$, limiting the increasing trend of the mixing layer width.

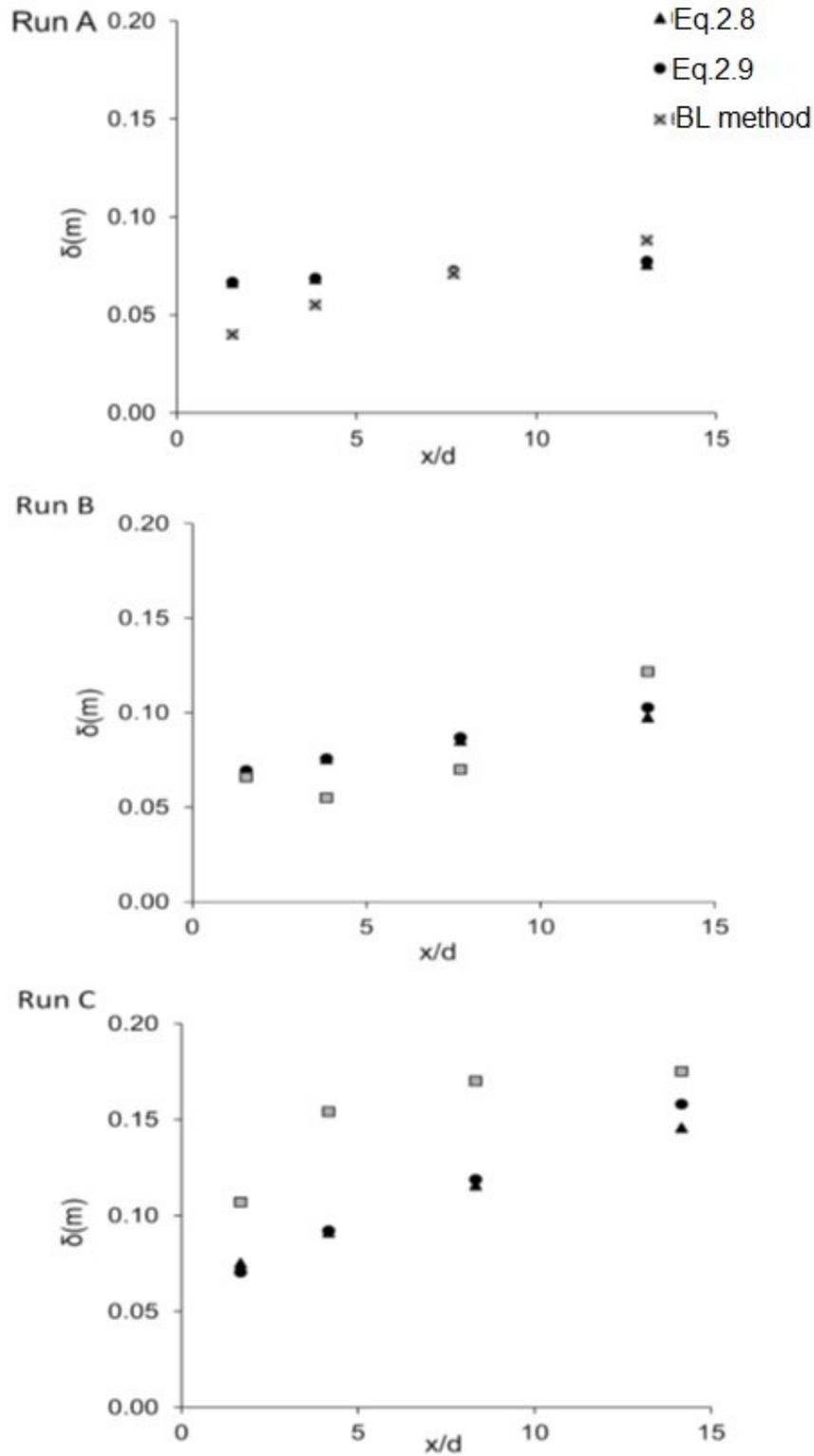


Figure 4.17. Mixing layer width for: a) Run A; b) Run B; c) Run C, from SPH computations at model time $t = 7$ s, compared to values calculated using Equations 2.8 and 2.9

Table 4.3 presents a comparison of mixing layer width between the PTV experiments and SPH computations. Values of δ are determined using the boundary layer definition for both the experiments and computations. A good agreement between the experimental and computational results is achieved. For Run A, at $x = 0.1$ m, and for Run B at $x = 0.25$ m, the SPH computations gave smaller values of the mixing layer width than the experiments. This could be because of the mixing layer intermittency seen in the simulations. The mixing layer intermittency also was seen when δ obtained from the simulation was compared to Eqs. 2.8 and 2.9 (Figure 4.17).

Table 4.3. Comparison of mixing layer width from PTV experiments and SPH computations. The superscript a indicates two values for $x = 0.1$ and 0.25 m, respectively

Run ID	δ (m) PTV	δ (m) SPH
A	(0.05, 0.055) ^a	(0.04,0.055)
B	(0.065, 0.07)	(0.06,0.04)

4.5 Conclusion

This chapter presents the results of SML from SPH computations. From the computed trajectories of particles, we derive flow velocity, vorticity, and the Okubo-Weiss parameter. We compare some of the results with measurements from the SML experiments discussed in Chapter 3, as well as with some empirical relationships reported in the literature. An analysis of the computational results and data comparison have led to the following conclusions:

- When two parallel shallow streams join at a channel confluence, wakes develop behind (or downstream of) their initial joining point. The wakes effectively cause a velocity deficit up to one and a half times the depth of flow in the near-field. This is regardless of whether the velocity ratio between the two incoming streams is small or large. A smaller velocity ratio ($V_r = 1.14$) gives rise to a more visible velocity deficit because of the relative importance of wake versus mixing layer dynamics. At a large velocity ratio ($V_r = 3$), the velocity gradient dominates the velocity deficit.
- On the basis of the computed vorticity field, the velocity ratio has a significant impact on the temporal and spatial evolutions of eddies in SML. At the small velocity ratio ($V_r = 1.14$), the vortices decay in a shorter distance downstream, compared to those at $V_r = 1.5$. A pairing process of these vortices occurs less often when the velocity ratio is small ($V_r = 1.14$), compared to when the ratio is large ($V_r = 1.5$). As expected, eddies are the strongest at $V_r = 3$, among the three values of V_r covered in this chapter; the evolutions of the eddies provide new insights into the pairing process in SML. The pairing activities are mostly affected by the average vorticity magnitude of two neighboring eddies rather than their relative distance. For example, for Run A, although there are eddies close to each other, they do not necessarily pair.
- The Okubo-Weiss parameter illustrates the local dominance of vorticity over the deformation right downstream of the initial joining point of the two incoming streams. The distribution of the parameter values shows that the general form of an eddy in SML consists of an inner vorticity-dominated region at the core and an outer region, which is strain-dominated and surrounds the inner region. The strain-dominated boundary of the eddies performs a barrier for

the particles inside the eddy until the eddies decay or are paired with other eddies. These results have not been reported in the literature and bring new insights into the evolution of flow structures in a SML.

- The exchange of fluid masses across the interface between the two streams is highly affected by the velocity ratio in the SML. The smaller velocity ratios ($V_r = 1.14$ and 1.5) result in lower mass transfer with a constant rate from the initial joining point to the downstream whilst at $V_r = 3$, the mass transfer rate increases due to larger eddies and more profound pairing process. The mass transfer from the two sides is unbalanced when the velocity ratio is large ($V_r = 3$), while at $V_r = 1.14$ the mass transfer is balanced.
- In the SML, the momentum transfer is mainly maintained by the high-velocity stream. The intermittent crests and troughs in momentum transfer indicate the evolution of eddies. The larger eddies downstream of the initial joining point are responsible for the main portion of the momentum transfer. A decrease in the velocity ratio from 3 to 1.5 results in a significant decline in momentum transfer across the interface between the two streams.
- The boundary layer growing (along each of the two side surfaces of the splitter plate or along the inner channel sidewalls) prior to the streams flowing into the confluence, plays a significant role in the development of mixing layer width, especially when the velocity gradient between the two streams is very small (Run A). This is due to the relative importance of wake versus SML. Therefore, the boundary layer definition used in this paper to determine the mixing layer width is rigorous and more appropriate than the previous proposed empirical relations. A comparison of the mixing layer width determined in this study and the previous studies shows that for smaller velocity ratios, the mixing layer width determined by the boundary layer definition is smaller than those by the empirical relations for $x/d < 7.7$, where the mixing layer is significantly affected by the boundary layer development. This is in agreement with the experimental results of SML width obtained by PTV for velocity ratios of 1.14 and 1.5. For the large velocity ratio, the mixing layer width determined by the boundary layer definition was larger compared to those by the empirical relations.

Chapter 5

Conclusions and future directions

5.1 Summary of research contributions

This thesis research has investigated the characteristics of shallow mixing layer (SML) in a channel confluence, by means of particle tracking experiments as well as mathematical computations of particle trajectories. The SML results from the interaction between two parallel streams with different mean streamwise velocities. To the best of our knowledge, so far very limited studies have dealt with SMLs from the Lagrangian perspective. The computations in this study use the Smoothed Particle Hydrodynamics methods; this is perhaps the first Lagrangian computations of SML. The computations in this study have produced detailed turbulence structures not available from the existing studies of SMLs.

Particle tracking velocimetry (PTV) techniques are useful for the study of the behavior of fluid particles in a SML. Recent advances in camera capabilities and computing power for handling a large volume of image data have made PTV a popular technique for turbulent flow measurements. We have made 3D-PTV measurements of particle trajectories from flume experiments of SML. The contributions from this study include appropriate modifications to PTV data acquisition and processing, and the creation of longer trajectories of fluid particles. In addition, we have developed new techniques for removing data outliers from PTV datasets; this fills the knowledge gap in the existing studies that the traditional techniques for outlier detection are not applicable to PTV datasets.

This thesis research includes dye-based flow visualization experiments of SML. These experiments have produced valuable data about SML formation, flow patterns and their transition, and eddy dimensions. The experimental data helps validate the SPH computations of SML.

We have identified several technical limitations (e.g., illumination intensity, seeding, camera memory storage) when using the PTV techniques for measurements of SML, and further proposed appropriate technical modifications to PTV setup. One limitation is the difficulty to handle a wide range of velocity ratios between the two incoming streams. In particular, it is difficult to capture long trajectories of particles at high velocity ratios. Besides, PTV is an expensive technique to use. We have demonstrated combining PTV measurements with computations of particle trajectories as a practical approach to obtaining a more complete picture of SML characteristics. The computations cover large velocity ratios and permit the quantification of wakes and their effects on mixing layer growth and of eddy pairing mechanisms.

5.2 Conclusions

On the basis of SML experiments, the main findings are as follows: The PTV measurements in this study have produced longer trajectories of fluid particles in a SML than previous studies. The success is attributed to the modifications in this study to PTV data acquisition and data processing methods. This study has overcome the limitations of the previous studies in terms of data

acquisition (improved illumination and particle seeding) and data post-processing (developed Python code for particle trajectory analysis).

It is appropriate to apply ensemble average to PTV data for the purpose of analyzing transverse profiles of the streamwise velocity and of determining the SML width. The boundary layer that develops along channel sidewalls upstream of the channel confluence and the wakes in the near field pose a limit on the growth of shallow mixing layer. These effects are weakened by an increase in lateral shear between the two interacting streams at a larger velocity ratio. The effect of the boundary layer on the velocity profiles gradually disappears with downstream distance.

We propose to determine the SML width according to the boundary layer (BL) definition, which is rigorous and more importantly, which better reflects the flow condition at a real-world river confluence than the existing empirical relations reported in the literature. The results show that the mixing layer width determined from the BL definition is smaller than those from the empirical relations. One plausible explanation is that the development of shallow mixing layer is constrained by the boundary layer development, resulting in a smaller mixing layer width.

We determine the finite-time Lyapunov exponent (FTLE) for particle trajectories to measure the divergence and convergence of particles inside or outside of the shallow mixing layer. For particle trajectories located inside the mixing layer, a divergence is evident with a positive value of FTLE. For particle trajectories outside the mixing layer, both positive and negative values of FTLE are observed.

The dye visualization sheds light on the flow behaviour of SML, especially at large downstream distances, in complement to the PTV measurements from a limited region that does not cover the large downstream distances. The dye visualization results show that turbulent instabilities still form in a lack of velocity gradient ($V_r = 1$). However, in the presence of a velocity gradient, the instabilities persist and pairing of eddies is observed. The intermittency of SML is observed with a lack of a temporally fixed pattern in the vortex arrangement at a fixed location.

On the basis of computations, we draw the following conclusions: As seen from the PTV measurements, a velocity deficit in the near field due to wake effects exists in the computed transverse profiles of the streamwise velocity regardless of the velocity ratio. The relative importance of velocity gradient and boundary layer development determines the extent to which the velocity profiles exhibit velocity deficit.

The computed vorticity field shows that the velocity ratio has a significant impact on the temporal and spatial evolutions of eddies. A pairing process of these vortices occurs less often when the velocity ratio is small. The pairing activities in SML are mostly affected by the average vorticity magnitude of two neighboring eddies rather than their relative distance. The distribution of computed values for the Okubo-Weiss parameter reveals that an eddy in SML consists of an inner vorticity-dominated region at the core and an outer region, which is strain-dominated and surrounds the inner region. The strain-dominated boundary of the eddies performs a barrier for the particles inside the eddy until the eddies decay or are paired with other eddies.

The fluid mass exchange across the interface between the fast and the slow flowing streams is highly affected by the velocity ratio in the SML. Smaller velocity ratios result in lower mass transfer with a constant rate, whilst at $V_r = 3$, the mass transfer rate increases in the downstream direction due to larger eddies and a more profound pairing process. In the SML, the momentum

transfer is mainly maintained by the high-velocity stream. The intermittent crests and troughs in momentum transfer indicate the evolution of eddies. The large eddies at the confluence are responsible for the main portion of the momentum transfer. Decreasing the velocity ratio from 3 to 1.5 results in a significant decline in momentum transfer across the interface.

The SPH computations yield mixing layer widths, according to the boundary layer definition, which are in consistency with the PTV measurements. At smaller velocity ratios when the mixing layer is significantly affected by the boundary layer development, the mixing layer width determined by the boundary layer definition is smaller than those by the empirical methods. At a larger velocity ratio, the mixing layer width determined by the boundary layer definition is larger, compared to the empirical relations.

5.3 Limitations

It is important to point out that PTV data involves a long chain of measurements, image processing, coordinate determination, and particle tracking before producing the desired temporal and spatial particle velocities. The complexity of the procedures could cause the improvement of the technique complicated since many contributing factors are involved in the PTV procedure.

The version of DualSPHysics used in this study does not store particle tracking data in output files. This is a limitation. Following the communications with *Ephys* lab at the University of Vigo (one of the DualSPHysics developers), the output feature will be added in the next release.

5.4 Recommendations for future research

The conclusions given in the preceding section are limited to SML parameters that have been analyzed and tested. For future research, we make a number of recommendations:

- Explore the far-field region of SML using PTV techniques. The results from this study qualitatively show that large coherent structures in the far-field play an important role in the fluid mass and momentum exchange. The PTV measurements from this study are limited to the near field. PTV measurements of the far-field would bring new insights into the pairing process.
- Consider using an Ion Laser as the illumination system for PTV measurements because it provides a faster-transmitting light flicker and a longer transmitting distance than both LED and Halogen lamps. Install high-speed cameras in a location where their distances to the target area of imaging can be easily adjusted.
- Study the effect of confluence angle on SML. In this research the two incoming streams are parallel, and the effect is not studied.
- Extend the computations to investigate contributing factors (e.g., flow depth, angle of flowing streams entering a confluence, various levels of channel-bed friction, and background turbulence) other than the velocity ratio in SML evolutions.

References

- Alavian V, Chu VH. (1985). Turbulent exchange flow in shallow compound channel. In Proceedings of the 21st International Congress of IAHR ,1985 Aug ,446-451.
- Akutina, Y. (2015). Experimental investigation of flow structures in a shallow embayment using 3D-PTV. PhD Thesis, Montreal, Canada, McGill University.
- Arnéodo A, Benzi R, Berg J, Biferale L, Bodenschatz E, Busse A, Calzavarini E, Castaing B, Cencini M, Chevillard L, Fisher RT. (2008). Universal intermittent properties of particle trajectories in highly turbulent flows. *Physical Review Letters*. 100(25); 254504.
- Brown, G.L. and Roshko, A. (1974). On density effects and large structure in turbulent mixing layers. *Journal of Fluid Mechanics*, 64(4); 775-816.
- Babarutsi S, Chu VH. (1998). Modeling transverse mixing layer in shallow open-channel flows. *Journal of hydraulic engineering*, 124(7); 718-27.
- Canelas RB, Crespo AJ, Domínguez JM, Ferreira RM, Gómez-Gesteira M. (2016). SPH–DCDEM model for arbitrary geometries in free surface solid–fluid flows. *Computer Physics Communications*. 202; 131-140.
- Chen S, Wang W, van Zuylen H. (2010). A comparison of outlier detection algorithms for ITS data. *Expert Systems with Applications*. 37(2); 1169-78.
- Cheng Z, Constantinescu G. (2020). Near- and far-field structure of shallow mixing layers between parallel streams. *Journal of Fluid Mechanics*. 2020 Dec 1; 904.
- Cheng Z, Constantinescu G. (2021). Shallow mixing layers between non-parallel streams in a flat-bed wide channel. *Journal of Fluid Mechanics*. 2021 Jun 26; 916.
- Chu, V. H. and S. Babarutsi (1988). Confinement and Bed Friction Effects in Shallow Turbulent Mixing Layers. *Journal of Hydraulic Engineering*, 114(10); 1257-1274.
- Chu, V.H., J.H. Wu and R.E. Khayat, (1991). Stability of transverse shear flows in shallow open channels. *Journal of Hydraulic Engineering*, 117(10); 1370-1388.
- Colagrossi A, Landrini M. (2003). Numerical simulation of interfacial flows by smoothed particle hydrodynamics. *Journal of computational physics*. 2003 Nov; 191(2); 448-75.
- Constantinescu, G., Miyawaki, S., Rhoads, B., Sukhodolov, A. and Kirkil, G. (2011). Structure of turbulent flow at a river confluence with momentum and velocity ratios close to 1: Insight provided by an eddy-resolving numerical simulation. *Water Resources Research*, 47(5).
- Crespo AJ, Gómez-Gesteira M, Dalrymple RA. (2007). Boundary conditions generated by dynamic particles in SPH methods. *Computers, Materials and Continua*. 5(3); 173-184.
- Crespo AJ, Domínguez JM, Rogers BD, Gómez-Gesteira M, Longshaw S, Canelas RJ, Vacondio R, Barreiro A, García-Feal O. (2015). DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH). *Computer Physics Communications*. 187; 204-16.
- Dalrymple RA, Knio O. (2001). SPH modelling of water waves. In *Coastal dynamics*, 2001 Jan. 779-787.

- Domínguez, J. M., Fourtakas, G., Altomare, C., Canelas, R. B., Tafuni, A., García-Feal, O., ... & Gómez-Gesteira, M. (2021). DualSPHysics: from fluid dynamics to multiphysics problems. *Computational Particle Mechanics*, 2021 Mar 29, 1-29.
- Duncan J, Dabiri D, Hove J, Gharib M. (2010). Universal outlier detection for particle image velocimetry (PIV) and particle tracking velocimetry (PTV) data. *Measurement Science and Technology*. 21(5); 057002.
- English, A., J. M. Domínguez, R. Vacondio, A. J. C. Crespo, P. K. Stansby, S. J. Lind, L. Chiapponi, and M. Gómez-Gesteira. (2021). Modified dynamic boundary conditions (mDBC) for general-purpose smoothed particle hydrodynamics (SPH): application to tank sloshing, dam break and fish pass problems. *Computational Particle Mechanics*, 1-15.
- Federico, I., Marrone, S., Colagrossi, A., Aristodemo, F., & Antuono, M. (2012). Simulating 2D open-channel flows through an SPH model. *European Journal of Mechanics-B/Fluids*, 34; 35-46.
- Fraga Filho, C. A. D., Fraga Filho, C. A. D., & Castro. (2019). *Smoothed Particle Hydrodynamics*. Springer International Publishing.
- Frazer, R.A., Jones, W.P., Skan, S.W. (1937). Approximations to functions and to the solutions of Differential Equations. Great Britain Aero Council. Rep. and Memo. No. 1799, London.
- Gabreil E., T.S.J.S.S., Nichols, A. (2018). Sphysics simulation of laboratory shallow free surface turbulent flows over a rough bed. *Journal of Hydraulic Research* 56(5); 727-747.
- Gingold RA, Monaghan JJ. (1977). Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*. 181(3):375-89.
- Gómez-Gesteira, M., & Dalrymple, R. A. (2004). Using a three-dimensional smoothed particle hydrodynamics method for wave impact on a tall structure. *Journal of waterway, port, coastal, and ocean engineering*, 130(2), 63-69.
- Gomez-Gesteira M, Rogers BD, Crespo AJ, Dalrymple RA, Narayanaswamy M, Dominguez JM. (2012). SPHysics—development of a free-surface fluid solver—Part 1: Theory and formulations. *Computers & Geosciences*. 48: 289-99.
- Gómez-Gesteira M, Crespo AJ, Rogers BD, Dalrymple RA, Dominguez JM, Barreiro A. (2012). SPHysics—development of a free-surface fluid solver—Part 2: Efficiency and test cases. *Computers & Geosciences*. 48: 300-7.
- Goring DG, Nikora VI. (2002). Despiking acoustic Doppler velocimeter data. *Journal of hydraulic engineering*. 128(1): 117-26.
- Han L, Mignot E, Riviere N. (2017). Shallow mixing layer downstream from a sudden expansion. *Journal of Hydrauliques Engineering*. 143(5):04016105.
- Higham JE, Brevis W, Keylock CJ. (2018). Implications of the selection of a particular modal decomposition technique for the analysis of shallow flows. *Journal of Hydraulic Research*. 56(6): 796-805.
- Hoyer, K., Holzner, M., Lüthi, B., Guala, M., Liberzon, A. and Kinzelbach, W. (2005). 3D scanning particle tracking velocimetry. *Experiments in Fluids*, 39(5), 923.

- Holzner, M., Liberzon, A., Nikitin, N., Lüthi, B., Kinzelbach, W. and Tsinober, A. (2008). A Lagrangian investigation of the small-scale features of turbulent entrainment through particle tracking and direct numerical simulation. *Journal of Fluid Mechanics*, 598, 465-475.
- Hu XY, Adams NA. (2007). An incompressible multi-phase SPH method. *Journal of computational physics*. 227(1): 264-78.
- Jirka, G. H. (2001). Large-scale flow structures and mixing processes in shallow flows. *Journal of Hydraulic Research*, 39(6), 567-573.
- Jimenez, J., Cogollos, M., & Bernal, L. P. (1985). A perspective view of the plane mixing layer. *Journal of Fluid Mechanics*, 152, 125-143.
- Kim, J. T. (2015). Three-dimensional particle tracking velocimetry for turbulence applications, University of Illinois Urbana-Champaign.
- Lüthi B, Ott S, Berg J, Mann J. (2007). Lagrangian multi-particle statistics. *Journal of Turbulence*. 2007 Jan 1(8): N45.
- Luthi, B., A. Tsinober and W. Kinzelbach, (2005). Lagrangian measurement of vorticity dynamics in turbulent flow. *Journal of Fluid Mechanics* 528(1); 87-118.
- Lüthi B. (2002). Some aspects of strain, vorticity and material element dynamics as measured with 3D particle tracking velocimetry in a turbulent flow. 2002 (Doctoral dissertation, ETH Zurich).
- Liu FT, Ting KM, Zhou ZH. (2008). Isolation forest. In 2008 eighth IEEE international conference on data mining 2008 Dec 15 (pp. 413-422). IEEE.
- Liu, M. B., Liu, G. R., & Lam, K. Y. (2003). Constructing smoothing functions in smoothed particle hydrodynamics with applications. *Journal of Computational and applied Mathematics*, 155(2), 263-284.
- Lanczos, C. (1938). Trigonometric interpolation of empirical and analytical functions, *J. Math. Phys* 17, 123-199
- Lesieur, M. (2008). Introduction to turbulence in fluid mechanics. *Turbulence in Fluids: Fourth Revised and Enlarged Edition*. 1-23
- Maas, H. G., A. Gruen and D. Papantoniou, (1993). Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids* 15(2): 133-146.
- Macia, F., Colagrossi, A., Antuono, M., & Souto-Iglesias, A. (2011). Benefits of using a Wendland kernel for free-surface flows. In 6th ERCOFTAC SPHERIC workshop on SPH applications, 2011 June, 30-37.
- Malik, N. A., T. Dracos and D. A. Papantoniou, (1993). Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids* 15(4), 279-294.
- Martínez, C. S. (2017). A near-field study of a shallow mixing layer between coflowing streams using 3D-PTV. Master's thesis, McGill University, Montreal.
- Mehta RD. (1991). Effect of velocity ratio on plane mixing layer development: Influence of the splitter plate wake. *Experiments in Fluids*. 1991 Jan;10(4):194-204.

- Meller, Y and Liberzon, A. (2016). Particle Data Management Software for 3D Particle Tracking Velocimetry and Related Applications – The Flowtracks Package. *Journal of Open Research Software*, 4: e23, DOI: <http://dx.doi.org/10.5334/jors.101>
- Meister, M., Burger, G., & Rauch, W. (2014). On the Reynolds number sensitivity of smoothed particle hydrodynamics. *Journal of Hydraulic Research*, 52(6), 824-835.
- Molteni, Diego, and Andrea Colagrossi. (2009). A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH. *Computer Physics Communications* 180.6; 861-872.
- Monaghan, J. J. (1992). Smoothed particle hydrodynamics. *Annual review of astronomy and astrophysics*, 30(1), 543-574.
- Mulligan S, Creedon L, Casserly J, Sherlock R. (2018) An improved model for the tangential velocity distribution in strong free-surface vortices: An experimental and theoretical study. *Journal of Hydraulic Research*. 2018 Oct 22.
- Nazari Z, Kang D. (2018). Evaluation of Multivariate Outlier Detection Methods with Benchmark Medical Datasets. *International Journal of Computer Science and Network Security*.18(4); 36-43.
- Okubo, A., (1970). Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences. *Deep-Sea Res.*, 17, 445–454
- Oliveira, J.G., van der Geld, C.W.M. and Kuerten, J.G. (2013). Lagrangian and Eulerian statistics of pipe flows measured with 3D-PTV at moderate and high Reynolds numbers. *Flow, turbulence and combustion*, 91(1), 105-137.
- Pope, S. B., & Pope, S. B. (2000). *Turbulent flows*. Cambridge university press.
- Raffel, M. (2007). *Particle image velocimetry: a practical guide*. Heidelberg; New York, Springer.
- Roshko, A. (1981). The plane mixing layer flow visualization results and three-dimensional effects. In *The role of coherent structures in modelling turbulence and mixing* (pp. 208-217). Springer, Berlin, Heidelberg.
- Schölkopf B, Smola A. (2005). Support vector machines and kernel algorithms. In *Encyclopedia of Biostatistics 2005* (pp. 5328-5335). Wiley.
- Schouten, J. C., Vander Stappen, M. L. M., & Van den Bleek, C. M. (1996). Scale-up of chaotic fluidized bed hydrodynamics. *Chemical Engineering Science*, 51(10), 1991-2000.
- Sprott, J. C., & Sprott, J. C. (2003). *Chaos and time-series analysis* (Vol. 69). Oxford: Oxford university press.
- Sukhodolov, A. N., I. Schnauder and W. S. J. Uijttewaal (2010). Dynamics of shallow lateral shear layers: Experimental study in a river with a sandy bed. *WRCR Water Resources Research* 46(11), 1- 18.
- Sukhodolov, A. N., J. Krick, T. A. Sukhodolova, Z. Cheng, B. L. Rhoads, and G. S. Constantinescu (2017). Turbulent flow structure at a discordant river confluence: Asymmetric jet dynamics with implications for channel morphology, *J. Geophys. Res. Earth Surf.* 122, 1278–1293, doi: 10.1002/2016JF004126.

- Tafuni A. (2016). Smoothed Particle Hydrodynamics: Development and application to problems of hydrodynamics, Doctoral dissertation, Polytechnic Institute of New York University.
- Thorne PD, Hanes DM. (2002). A review of acoustic measurement of small-scale sediment processes. *Continental shelf research*. 22(4):603-32.
- Uijttewaal, W. and R. Booij. (2000). Effects of shallowness on the development of free-surface mixing layers. *Physics of Fluids* 12(2), 392-402.
- Uijttewaal, W. S., and J. Tukker. (1998), Development of quasi two-dimensional structures in shallow free-surface mixing layer, *Exp. Fluids*, 24(3), 192–200.
- van Prooijen, B. C. and W. S. J. Uijttewaal. (2002). A linear approach for the evolution of coherent structures in shallow mixing layers. *Physics of Fluids* 14(12): 4105.
- Vacondio R, Rogers BD, Stansby PK, Mignosa P. (2013). Shallow water SPH for flooding with dynamic particle coalescing and splitting. *Advances in Water Resources*. 58:10-23.
- Virant M, Dracos T. (1997). 3D PTV and its application on Lagrangian motion. *Measurement science and technology*. 8(12):1539.
- Winant, C. D., & Browand, F. K. (1974). Vortex pairing: the mechanism of turbulent mixing-layer growth at moderate Reynolds number. *Journal of Fluid Mechanics*, 63(2), 237-255.
- Wu, Ke, Dongmin Yang, and Nigel Wright. (2016). A coupled SPH-DEM model for fluid-structure interaction problems with free-surface flow and structural failure. *Computers & Structures*. 177; 141-161.
- Weiss, J., (1991). The dynamics of enstrophy transfer in two-dimensional hydrodynamics. *PhysicaD*, 48, 273–294
- Wendland H. (1995). Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in computational Mathematics*. 4(1):389-96.
- Westerweel J, Scarano F. (2005). Universal outlier detection for PIV data. *Experiments in fluids*. 39(6):1096-100.
- Wu K, Yang D, Wright N. (2016). A coupled SPH-DEM model for fluid-structure interaction problems with free-surface flow and structural failure. *Computers & Structures*. 177:141-61.
- Xu R, Stansby P, Laurence D. (2009). Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach. *Journal of computational Physics*. 228(18):6703-25.
- Zikanov, O. (2019). *Essential computational fluid dynamics*. John Wiley & Sons.

Appendix A1: Extensible Markup Language (XML) file for SPH simulation, Run A

```
<case>
  <casedef>
    <constantsdef>
      <gravity x="0" y="0" z="0" comment="Gravitational acceleration" units_comment="m/s^2" />
      <rhop0 value="1000" comment="Reference density of the fluid" units_comment="kg/m^3" />
      <hswl value="0" auto="true" comment="Maximum still water level to calculate speedofsound
using coefsound" units_comment="metres (m)" />
      <gamma value="7" comment="Politropic constant for water used in the state equation" />
      <speedsystem value="0" auto="true" comment="Maximum system speed (by default the dam-break
propagation is used)" />
      <coefsound value="0" comment="Coefficient to multiply speedsystem" />
      <speedsound value="15" auto="false" comment="Speed of sound to use in the simulation (by
default speedofsound=coefsound*speedsystem)" />
      <hdp value="1.5" comment="Alternative option to calculate the smoothing length (h=hdp*dp)"
/>
      <cflnumber value="0.2" comment="Coefficient to multiply dt" />
    </constantsdef>
    <mkconfig boundcount="240" fluidcount="9" />
    <geometry>
      <definition dp="0.002">
        <pointmin x="-0.5" y="0" z="-1" />
        <pointmax x="2.5" y="0" z="1" />
      </definition>
      <commands>
        <mainlist>
          <setshapemode>dp | bound</setshapemode>
          <setdrawmode mode="full" />
          <!-- SOLID BOUNDARY -->
          <setmkbound mk="0" />
          <drawbox>
            <boxfill>solid</boxfill>
            <point x="-0.02" y="0" z="-0.395" />
            <size x="1.04" y="0.1" z="-0.05" />
          </drawbox>
          <setmkbound mk="2" />
          <drawbox>
```

```

        <boxfill>solid</boxfill>
        <point x="-0.02" y="0" z="0.395" />
        <size x="1.04" y="0.1" z="0.05" />
    </drawbox>

<!-- splitter -->
    <setmkbound mk="1" />
    <drawbox>
        <boxfill>solid</boxfill>
        <point x="-0.02" y="0" z="0" />
        <size x="0.04" y="0.1" z="0.004" />
    </drawbox>
    <!-- inlet_1 -->
    <setmkfluid mk="0" />
    <drawbox>
        <boxfill>left</boxfill>
        <point x="0.018" y="0" z="-0.393" />
        <size x="0.002" y="0.064" z="0.391" />
    </drawbox>
    <shapeout file="" reset="true" />
    <!-- fluid_1 -->
    <setmkfluid mk="2" />
    <drawbox>
        <boxfill>solid</boxfill>
        <point x="0.020" y="0" z="-0.393" />
        <size x="0.979" y="0.064" z="0.395" />
    </drawbox>
    <shapeout file="" reset="true" />
    <!-- inlet_2 -->
    <setmkfluid mk="3" />
    <drawbox>
        <boxfill>left</boxfill>
        <point x="0.018" y="0" z="0.006" />
        <size x="0.002" y="0.064" z="0.387" />
    </drawbox>
    <!-- fluid_2 -->
    <setmkfluid mk="5" />
    <drawbox>
        <boxfill>solid</boxfill>

```

```

        <point x="0.020" y="0" z="0.004" />
        <size x="0.979" y="0.064" z="0.389" />
    </drawbox>
    <!-- outlet -->
    <setmkfluid mk="6" />
    <drawbox>
        <boxfill>right</boxfill>
        <point x="0" y="0" z="-0.393" />
        <size x="1" y="0.064" z="0.786" />
    </drawbox>
    <shapeout file="" />
</mainlist>
</commands>
</geometry>
<initials>
    <velocity mkfluid="2">
        <direction x="1" y="0" z="0" />
        <velocity v="0.377" comment="Uniform profile velocity" units_comment="m/s" />
    </velocity>
    <velocity mkfluid="5">
        <direction x="1" y="0" z="0" />
        <velocity v="0.43" comment="Uniform profile velocity" units_comment="m/s" />
    </velocity>
</initials>
<motion>
    <objreal ref="0">
        <begin mov="1" start="0" finish="100" />
        <mvrect id="1" duration="100">
            <vel x="0.377" y="0" z="0" units_comment="m/s" />
        </mvrect>
    </objreal>
    <objreal ref="2">
        <begin mov="1" start="0" finish="100" />
        <mvrect id="1" duration="100">
            <vel x="0.43" y="0" z="0" units_comment="m/s" />
        </mvrect>
    </objreal>
</motion>
</casedef>

```

```

<execution>
  <special>
    <inout>
      <memoryresize size0="2" size="4" comment="Initial memory resize (size0) and the
following memory resizes according to initial inlet/outlet particles (default=2 and 4)"/>
      <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order
(default=1e+3)" />
      <!-- INFLOW BUFFER1 -->
      <inoutzone>
        <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
        <inputtreatment value="2" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
        <layers value="10" comment="Number of inlet/outlet particle layers" />
        <zone2d comment="Input zone for 2D simulations">
          <particles mkfluid="0" direction="right" />
        </zone2d>
        <imposevelocity mode="0" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)">
          <velocity v="0.377" comment="Uniform velocity" units_comment="m/s" />
        </imposevelocity>
        <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)" />
      </inoutzone>
      <!-- INFLOW BUFFER2 -->
      <inoutzone>
        <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
        <inputtreatment value="2" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
        <layers value="10" comment="Number of inlet/outlet particle layers" />
        <zone2d comment="Input zone for 2D simulations">
          <particles mkfluid="3" direction="right" />
        </zone2d>
        <imposevelocity mode="0" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)">
          <velocity v="0.43" comment="Uniform velocity" units_comment="m/s" />
        </imposevelocity>

```

```

                <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)" />
            </inoutzone>
            <!-- OUTFLOW BUFFER -->
            <inoutzone>
                <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
                <inputtreatment value="1" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
                <layers value="4" comment="Number of inlet/outlet particle layers" />
                <zone2d comment="Input zone for 2-D simulations">
                    <particles mkfluid="6" direction="left" />
                </zone2d>
                <imposevelocity mode="2" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)"/>
                <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)"/>
            </inoutzone>
            <!-- END BUFFERS -->
        </inout>
        <!-- EXTRAPOLATION FOR SOLID BOUNDARY -->
        <boundcorr>
            <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order"
/>

            <mkzone mkbound="0">
                <limitpoint x="0" y="0" z="-0.385" />
                <direction x="0" y="0" z="1" comment="Direction to fluid" />
            </mkzone>
        </boundcorr>
        <boundcorr>
            <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order"
/>

            <mkzone mkbound="2">
                <limitpoint x="0" y="0" z="0.385" />
                <direction x="0" y="0" z="-1" comment="Direction to fluid" />
            </mkzone>
        </boundcorr>
    </special>
</parameters>

```

```

        <parameter key="SavePosDouble" value="0" comment="Saves particle position using double
precision (default=0)" />
        <parameter key="StepAlgorithm" value="2" comment="Step Algorithm 1:Verlet, 2:Symplectic
(default=1)" />
        <parameter key="VerletSteps" value="0" comment="Verlet only: Number of steps to apply
Euler timestepping (default=40)" />
        <parameter key="Kernel" value="2" comment="Interaction Kernel 1:Cubic Spline, 2:Wendland
(default=2)" />
        <parameter key="ViscoTreatment" value="2" comment="Viscosity formulation 1:Artificial,
2:Laminar+SPS (default=1)" />
        <parameter key="Visco" value="0.000001" comment="Viscosity value" />
        <parameter key="ViscoBoundFactor" value="1" comment="Multiply viscosity value with
boundary (default=1)" />
        <parameter key="DensityDT" value="2" comment="Density Diffusion Term 0:None, 1:Molteni,
2:Fourtakas, 3:Fourtakas(full) (default=0)" />
        <parameter key="DensityDTvalue" value="0.15" comment="DDT value (default=0.1)" />
        <parameter key="Shifting" value="3" comment="Shifting mode 0:None, 1:Ignore bound,
2:Ignore fixed, 3:Full (default=0)" />
        <parameter key="ShiftCoef" value="-20" comment="Coefficient for shifting computation
(default=-2)" />
        <parameter key="ShiftTFS" value="0" comment="Threshold to detect free surface. Typically
1.5 for 2D and 2.75 for 3D (default=0)" />
        <parameter key="RigidAlgorithm" value="1" comment="Rigid Algorithm 0:collision-free,
1:SPH, 2:DEM, 3:Chrono (default=1)" />
        <parameter key="FtPause" value="0.0" comment="Time to freeze the floatings at simulation
start (warmup) (default=0)" units_comment="seconds" />
        <parameter key="CoefDtMin" value="0.05" comment="Coefficient to calculate minimum time
step dtmin=coefdtmin*h/speedsound (default=0.05)" />
        <parameter key="DtIni" value="0" comment="Initial time step. Use 0 to default use
(default=h/speedsound)" units_comment="seconds" />
        <parameter key="DtMin" value="0" comment="Minimum time step. Use 0 to default use
(default=coefdtmin*h/speedsound)" units_comment="seconds" />
        <parameter key="DtFixed" value="0" comment="Fixed Dt value. Use 0 to disable
(default=disabled)" units_comment="seconds" />
        <parameter key="DtFixedFile" value="NONE" comment="Dt values are loaded from file. Use
NONE to disable (default=disabled)" units_comment="milliseconds (ms)" />
        <parameter key="DtAllParticles" value="0" comment="Velocity of particles used to calculate
DT. 1:All, 0:Only fluid/floating (default=0)" />
        <parameter key="TimeMax" value="7" comment="Time of simulation" units_comment="seconds" />

```

```

        <parameter key="TimeOut" value="0.05" comment="Time out data" units_comment="seconds" />
        <parameter key="PartsOutMax" value="1" comment="%/100 of fluid particles allowed to be
excluded from domain (default=1)" units_comment="decimal" />
        <parameter key="RhopOutMin" value="800" comment="Minimum rhop valid (default=700)"
units_comment="kg/m^3" />
        <parameter key="RhopOutMax" value="1200" comment="Maximum rhop valid (default=1300)"
units_comment="kg/m^3" />
        <parameter key="XPeriodicIncZ" value="0" comment="Increase of Z with periodic BC" />
        <simulationdomain comment="Defines domain of simulation (default=Uses minimum and maximum
position of the generated particles)">
            <posmin x="default" y="default" z="default" comment="e.g.: x=0.5, y=default-1,
z=default-10%" />
            <posmax x="default" y="default" z="default" />
        </simulationdomain>
    </parameters>
</execution>
</case>

```

Appendix A2: Extensible Markup Language (XML) file for SPH simulation, Run B

```
<case>
  <casedef>
    <constantsdef>
      <gravity x="0" y="0" z="0" comment="Gravitational acceleration" units_comment="m/s^2" />
      <rho0 value="1000" comment="Reference density of the fluid" units_comment="kg/m^3" />
      <hswl value="0" auto="true" comment="Maximum still water level to calculate speedofsound
using coefsound" units_comment="metres (m)" />
      <gamma value="7" comment="Politropic constant for water used in the state equation" />
      <speedsystem value="0" auto="true" comment="Maximum system speed (by default the dam-break
propagation is used)" />
      <coefsound value="0" comment="Coefficient to multiply speedsystem" />
      <speedsound value="15" auto="false" comment="Speed of sound to use in the simulation (by
default speedofsound=coefsound*speedsystem)" />
      <hdp value="1.5" comment="Alternative option to calculate the smoothing length (h=hdp*dp)"
/>
      <cflnumber value="0.2" comment="Coefficient to multiply dt" />
    </constantsdef>
    <mkconfig boundcount="240" fluidcount="9" />
    <geometry>
      <definition dp="0.002">
        <pointmin x="-0.5" y="0" z="-1" />
        <pointmax x="2.5" y="0" z="1" />
      </definition>
      <commands>
        <mainlist>
          <setshapemode>dp | bound</setshapemode>
          <setdrawmode mode="full" />
          <!-- SOLID BOUNDARY -->
          <setmkbound mk="0" />
          <drawbox>
            <boxfill>solid</boxfill>
            <point x="-0.02" y="0" z="-0.395" />
            <size x="1.04" y="0.1" z="-0.05" />
          </drawbox>
          <setmkbound mk="2" />
        </mainlist>
      </commands>
    </geometry>
  </casedef>
</case>
```

```

<drawbox>
  <boxfill>solid</boxfill>
  <point x="-0.02" y="0" z="0.395" />
  <size x="1.04" y="0.1" z="0.05" />
</drawbox>

<!-- splitter -->
<setmkbound mk="1" />
<drawbox>
  <boxfill>solid</boxfill>
  <point x="-0.02" y="0" z="0" />
  <size x="0.04" y="0.1" z="0.004" />
</drawbox>
<!-- inlet_1 -->
<setmkfluid mk="0" />
<drawbox>
  <boxfill>left</boxfill>
  <point x="0.018" y="0" z="-0.393" />
  <size x="0.002" y="0.064" z="0.391" />
</drawbox>
<shapeout file="" reset="true" />
<!-- fluid_1 -->
<setmkfluid mk="2" />
<drawbox>
  <boxfill>solid</boxfill>
  <point x="0.020" y="0" z="-0.393" />
  <size x="0.979" y="0.064" z="0.395" />
</drawbox>
<shapeout file="" reset="true" />
<!-- inlet_2 -->
<setmkfluid mk="3" />
<drawbox>
  <boxfill>left</boxfill>
  <point x="0.018" y="0" z="0.006" />
  <size x="0.002" y="0.064" z="0.387" />
</drawbox>
<!-- fluid_2 -->
<setmkfluid mk="5" />
<drawbox>

```

```

        <boxfill>solid</boxfill>
        <point x="0.020" y="0" z="0.004" />
        <size x="0.979" y="0.064" z="0.389" />
    </drawbox>
    <!-- outlet -->
    <setmkfluid mk="6" />
    <drawbox>
        <boxfill>right</boxfill>
        <point x="0" y="0" z="-0.393" />
        <size x="1" y="0.064" z="0.786" />
    </drawbox>
    <shapeout file="" />
</mainlist>
</commands>
</geometry>
<initials>
    <velocity mkfluid="2">
        <direction x="1" y="0" z="0" />
        <velocity v="0.3" comment="Uniform profile velocity" units_comment="m/s" />
    </velocity>
    <velocity mkfluid="5">
        <direction x="1" y="0" z="0" />
        <velocity v="0.45" comment="Uniform profile velocity" units_comment="m/s" />
    </velocity>
</initials>
<motion>
    <objreal ref="0">
        <begin mov="1" start="0" finish="100" />
        <mvrect id="1" duration="100">
            <vel x="0.3" y="0" z="0" units_comment="m/s" />
        </mvrect>
    </objreal>
    <objreal ref="2">
        <begin mov="1" start="0" finish="100" />
        <mvrect id="1" duration="100">
            <vel x="0.45" y="0" z="0" units_comment="m/s" />
        </mvrect>
    </objreal>
</motion>

```

```

</casedef>
<execution>
  <special>
    <inout>
      <memoryresize size0="2" size="4" comment="Initial memory resize (size0) and the
following memory resizes according to initial inlet/outlet particles (default=2 and 4)"/>
      <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order
(default=1e+3)" />
      <!-- INFLOW BUFFER1 -->
      <inoutzone>
        <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
        <inputtreatment value="2" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
        <layers value="10" comment="Number of inlet/outlet particle layers" />
        <zone2d comment="Input zone for 2D simulations">
          <particles mkfluid="0" direction="right" />
        </zone2d>
        <imposevelocity mode="0" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)">
          <velocity v="0.3" comment="Uniform velocity" units_comment="m/s" />
        </imposevelocity>
        <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)" />
      </inoutzone>
      <!-- INFLOW BUFFER2 -->
      <inoutzone>
        <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
        <inputtreatment value="2" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
        <layers value="10" comment="Number of inlet/outlet particle layers" />
        <zone2d comment="Input zone for 2D simulations">
          <particles mkfluid="3" direction="right" />
        </zone2d>
        <imposevelocity mode="0" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)">
          <velocity v="0.45" comment="Uniform velocity" units_comment="m/s" />
        </imposevelocity>

```

```

                <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)" />
            </inoutzone>
            <!-- OUTFLOW BUFFER -->
            <inoutzone>
                <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
                <inputtreatment value="1" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
                <layers value="4" comment="Number of inlet/outlet particle layers" />
                <zone2d comment="Input zone for 2-D simulations">
                    <particles mkfluid="6" direction="left" />
                </zone2d>
                <imposevelocity mode="2" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)"/>
                <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)"/>
            </inoutzone>
            <!-- END BUFFERS -->
        </inout>
        <!-- EXTRAPOLATION FOR SOLID BOUNDARY -->
        <boundcorr>
            <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order"
/>

            <mkzone mkbound="0">
                <limitpoint x="0" y="0" z="-0.385" />
                <direction x="0" y="0" z="1" comment="Direction to fluid" />
            </mkzone>
        </boundcorr>
        <boundcorr>
            <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order"
/>

            <mkzone mkbound="2">
                <limitpoint x="0" y="0" z="0.385" />
                <direction x="0" y="0" z="-1" comment="Direction to fluid" />
            </mkzone>
        </boundcorr>
    </special>
</parameters>

```

```

        <parameter key="SavePosDouble" value="0" comment="Saves particle position using double
precision (default=0)" />
        <parameter key="StepAlgorithm" value="2" comment="Step Algorithm 1:Verlet, 2:Symplectic
(default=1)" />
        <parameter key="VerletSteps" value="0" comment="Verlet only: Number of steps to apply
Euler timestepping (default=40)" />
        <parameter key="Kernel" value="2" comment="Interaction Kernel 1:Cubic Spline, 2:Wendland
(default=2)" />
        <parameter key="ViscoTreatment" value="2" comment="Viscosity formulation 1:Artificial,
2:Laminar+SPS (default=1)" />
        <parameter key="Visco" value="0.000001" comment="Viscosity value" />
        <parameter key="ViscoBoundFactor" value="1" comment="Multiply viscosity value with
boundary (default=1)" />
        <parameter key="DensityDT" value="2" comment="Density Diffusion Term 0:None, 1:Molteni,
2:Fourtakas, 3:Fourtakas(full) (default=0)" />
        <parameter key="DensityDTvalue" value="0.15" comment="DDT value (default=0.1)" />
        <parameter key="Shifting" value="3" comment="Shifting mode 0:None, 1:Ignore bound,
2:Ignore fixed, 3:Full (default=0)" />
        <parameter key="ShiftCoef" value="-20" comment="Coefficient for shifting computation
(default=-2)" />
        <parameter key="ShiftTFS" value="0" comment="Threshold to detect free surface. Typically
1.5 for 2D and 2.75 for 3D (default=0)" />
        <parameter key="RigidAlgorithm" value="1" comment="Rigid Algorithm 0:collision-free,
1:SPH, 2:DEM, 3:Chrono (default=1)" />
        <parameter key="FtPause" value="0.0" comment="Time to freeze the floatings at simulation
start (warmup) (default=0)" units_comment="seconds" />
        <parameter key="CoefDtMin" value="0.05" comment="Coefficient to calculate minimum time
step dtmin=coefdtmin*h/speedsound (default=0.05)" />
        <parameter key="DtIni" value="0" comment="Initial time step. Use 0 to default use
(default=h/speedsound)" units_comment="seconds" />
        <parameter key="DtMin" value="0" comment="Minimum time step. Use 0 to default use
(default=coefdtmin*h/speedsound)" units_comment="seconds" />
        <parameter key="DtFixed" value="0" comment="Fixed Dt value. Use 0 to disable
(default=disabled)" units_comment="seconds" />
        <parameter key="DtFixedFile" value="NONE" comment="Dt values are loaded from file. Use
NONE to disable (default=disabled)" units_comment="milliseconds (ms)" />
        <parameter key="DtAllParticles" value="0" comment="Velocity of particles used to calculate
DT. 1:All, 0:Only fluid/floating (default=0)" />
        <parameter key="TimeMax" value="7" comment="Time of simulation" units_comment="seconds" />

```

```

        <parameter key="TimeOut" value="0.05" comment="Time out data" units_comment="seconds" />
        <parameter key="PartsOutMax" value="1" comment="%/100 of fluid particles allowed to be
excluded from domain (default=1)" units_comment="decimal" />
        <parameter key="RhopOutMin" value="800" comment="Minimum rhop valid (default=700)"
units_comment="kg/m^3" />
        <parameter key="RhopOutMax" value="1200" comment="Maximum rhop valid (default=1300)"
units_comment="kg/m^3" />
        <parameter key="XPeriodicIncZ" value="0" comment="Increase of Z with periodic BC" />
        <simulationdomain comment="Defines domain of simulation (default=Uses minimum and maximum
position of the generated particles)">
            <posmin x="default" y="default" z="default" comment="e.g.: x=0.5, y=default-1,
z=default-10%" />
            <posmax x="default" y="default" z="default" />
        </simulationdomain>
    </parameters>
</execution>
</case>

```

Appendix A3: Extensible Markup Language (XML) file for SPH simulation, Run C

```
<case>
  <casedef>
    <constantsdef>
      <gravity x="0" y="0" z="0" comment="Gravitational acceleration" units_comment="m/s^2" />
      <rhop0 value="1000" comment="Reference density of the fluid" units_comment="kg/m^3" />
      <hswl value="0" auto="true" comment="Maximum still water level to calculate speedofsound
using coefsound" units_comment="metres (m)" />
      <gamma value="7" comment="Politropic constant for water used in the state equation" />
      <speedsystem value="0" auto="true" comment="Maximum system speed (by default the dam-break
propagation is used)" />
      <coefsound value="0" comment="Coefficient to multiply speedsystem" />
      <speedsound value="15" auto="false" comment="Speed of sound to use in the simulation (by
default speedofsound=coefsound*speedsystem)" />
      <hdp value="1.5" comment="Alternative option to calculate the smoothing length (h=hdp*dp)"
/>
      <cflnumber value="0.2" comment="Coefficient to multiply dt" />
    </constantsdef>
    <mkconfig boundcount="240" fluidcount="9" />
    <geometry>
      <definition dp="0.002">
        <pointmin x="-0.5" y="0" z="-1" />
        <pointmax x="2.5" y="0" z="1" />
      </definition>
      <commands>
        <mainlist>
          <setshapemode>dp | bound</setshapemode>
          <setdrawmode mode="full" />
          <!-- SOLID BOUNDARY -->
          <setmkbound mk="0" />
          <drawbox>
            <boxfill>solid</boxfill>
            <point x="-0.02" y="0" z="-0.395" />
            <size x="1.04" y="0.1" z="-0.05" />
          </drawbox>
          <setmkbound mk="2" />
        </mainlist>
      </commands>
    </geometry>
  </casedef>
</case>
```

```

<drawbox>
  <boxfill>solid</boxfill>
  <point x="-0.02" y="0" z="0.395" />
  <size x="1.04" y="0.1" z="0.05" />
</drawbox>

<!-- splitter -->
<setmkbound mk="1" />
<drawbox>
  <boxfill>solid</boxfill>
  <point x="-0.02" y="0" z="0" />
  <size x="0.04" y="0.1" z="0.004" />
</drawbox>
<!-- inlet_1 -->
<setmkfluid mk="0" />
<drawbox>
  <boxfill>left</boxfill>
  <point x="0.018" y="0" z="-0.393" />
  <size x="0.002" y="0.064" z="0.391" />
</drawbox>
<shapeout file="" reset="true" />
<!-- fluid_1-->
<setmkfluid mk="2" />
<drawbox>
  <boxfill>solid</boxfill>
  <point x="0.020" y="0" z="-0.393" />
  <size x="0.979" y="0.064" z="0.395" />
</drawbox>
<shapeout file="" reset="true" />
<!-- inlet_2 -->
<setmkfluid mk="3" />
<drawbox>
  <boxfill>left</boxfill>
  <point x="0.018" y="0" z="0.006" />
  <size x="0.002" y="0.064" z="0.387" />
</drawbox>
<!-- fluid_2 -->
<setmkfluid mk="5" />
<drawbox>

```

```

        <boxfill>solid</boxfill>
        <point x="0.020" y="0" z="0.004" />
        <size x="0.979" y="0.064" z="0.389" />
    </drawbox>
    <!-- outlet -->
    <setmkfluid mk="6" />
    <drawbox>
        <boxfill>right</boxfill>
        <point x="0" y="0" z="-0.393" />
        <size x="1" y="0.064" z="0.786" />
    </drawbox>
    <shapeout file="" />
</mainlist>
</commands>
</geometry>
<initials>
    <velocity mkfluid="2">
        <direction x="1" y="0" z="0" />
        <velocity v="0.3" comment="Uniform profile velocity" units_comment="m/s" />
    </velocity>
    <velocity mkfluid="5">
        <direction x="1" y="0" z="0" />
        <velocity v="0.45" comment="Uniform profile velocity" units_comment="m/s" />
    </velocity>
</initials>
<motion>
    <objreal ref="0">
        <begin mov="1" start="0" finish="100" />
        <mvrect id="1" duration="100">
            <vel x="0.3" y="0" z="0" units_comment="m/s" />
        </mvrect>
    </objreal>
    <objreal ref="2">
        <begin mov="1" start="0" finish="100" />
        <mvrect id="1" duration="100">
            <vel x="0.45" y="0" z="0" units_comment="m/s" />
        </mvrect>
    </objreal>
</motion>

```

```

</casedef>
<execution>
  <special>
    <inout>
      <memoryresize size0="2" size="4" comment="Initial memory resize (size0) and the
following memory resizes according to initial inlet/outlet particles (default=2 and 4)"/>
      <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order
(default=1e+3)" />
      <!-- INFLOW BUFFER1 -->
      <inoutzone>
        <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
        <inputtreatment value="2" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
        <layers value="10" comment="Number of inlet/outlet particle layers" />
        <zone2d comment="Input zone for 2D simulations">
          <particles mkfluid="0" direction="right" />
        </zone2d>
        <imposevelocity mode="0" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)">
          <velocity v="0.3" comment="Uniform velocity" units_comment="m/s" />
        </imposevelocity>
        <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)" />
      </inoutzone>
      <!-- INFLOW BUFFER2 -->
      <inoutzone>
        <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
        <inputtreatment value="2" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
        <layers value="10" comment="Number of inlet/outlet particle layers" />
        <zone2d comment="Input zone for 2D simulations">
          <particles mkfluid="3" direction="right" />
        </zone2d>
        <imposevelocity mode="0" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)">
          <velocity v="0.45" comment="Uniform velocity" units_comment="m/s" />
        </imposevelocity>

```

```

                <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)" />
            </inoutzone>
            <!-- OUTFLOW BUFFER -->
            <inoutzone>
                <refilling value="0" comment="Refilling mode. 0:Simple full, 1:Simple below zsurf,
2:Advanced for reverse flows (very slow) (default=1)" />
                <inputtreatment value="1" comment="Treatment of fluid entering the zone. 0:No
changes, 1:Convert fluid (necessary for outlet), 2:Remove fluid (recommended for inlet)" />
                <layers value="4" comment="Number of inlet/outlet particle layers" />
                <zone2d comment="Input zone for 2-D simulations">
                    <particles mkfluid="6" direction="left" />
                </zone2d>
                <imposevelocity mode="2" comment="Imposed velocity 0:fixed value, 1:variable
value, 2:Extrapolated value (default=0)"/>
                <imposerhop mode="2" comment="Outlet rhop 0:Imposed fixed value, 1:Hydrostatic,
2:Extrapolated from ghost nodes (default=0)"/>
            </inoutzone>
            <!-- END BUFFERS -->
        </inout>
        <!-- EXTRAPOLATION FOR SOLID BOUNDARY -->
        <boundcorr>
            <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order"
/>

            <mkzone mkbound="0">
                <limitpoint x="0" y="0" z="-0.385" />
                <direction x="0" y="0" z="1" comment="Direction to fluid" />
            </mkzone>
        </boundcorr>
        <boundcorr>
            <determlimit value="1e+3" comment="Use 1e-3 for first_order or 1e+3 for zeroth_order"
/>

            <mkzone mkbound="2">
                <limitpoint x="0" y="0" z="0.385" />
                <direction x="0" y="0" z="-1" comment="Direction to fluid" />
            </mkzone>
        </boundcorr>
    </special>
</parameters>

```

```

        <parameter key="SavePosDouble" value="0" comment="Saves particle position using double
precision (default=0)" />
        <parameter key="StepAlgorithm" value="2" comment="Step Algorithm 1:Verlet, 2:Symplectic
(default=1)" />
        <parameter key="VerletSteps" value="0" comment="Verlet only: Number of steps to apply
Euler timestepping (default=40)" />
        <parameter key="Kernel" value="2" comment="Interaction Kernel 1:Cubic Spline, 2:Wendland
(default=2)" />
        <parameter key="ViscoTreatment" value="2" comment="Viscosity formulation 1:Artificial,
2:Laminar+SPS (default=1)" />
        <parameter key="Visco" value="0.000001" comment="Viscosity value" />
        <parameter key="ViscoBoundFactor" value="1" comment="Multiply viscosity value with
boundary (default=1)" />
        <parameter key="DensityDT" value="2" comment="Density Diffusion Term 0:None, 1:Molteni,
2:Fourtakas, 3:Fourtakas(full) (default=0)" />
        <parameter key="DensityDTvalue" value="0.15" comment="DDT value (default=0.1)" />
        <parameter key="Shifting" value="3" comment="Shifting mode 0:None, 1:Ignore bound,
2:Ignore fixed, 3:Full (default=0)" />
        <parameter key="ShiftCoef" value="-20" comment="Coefficient for shifting computation
(default=-2)" />
        <parameter key="ShiftTFS" value="0" comment="Threshold to detect free surface. Typically
1.5 for 2D and 2.75 for 3D (default=0)" />
        <parameter key="RigidAlgorithm" value="1" comment="Rigid Algorithm 0:collision-free,
1:SPH, 2:DEM, 3:Chrono (default=1)" />
        <parameter key="FtPause" value="0.0" comment="Time to freeze the floatings at simulation
start (warmup) (default=0)" units_comment="seconds" />
        <parameter key="CoefDtMin" value="0.05" comment="Coefficient to calculate minimum time
step dtmin=coefdtmin*h/speedsound (default=0.05)" />
        <parameter key="DtIni" value="0" comment="Initial time step. Use 0 to default use
(default=h/speedsound)" units_comment="seconds" />
        <parameter key="DtMin" value="0" comment="Minimum time step. Use 0 to default use
(default=coefdtmin*h/speedsound)" units_comment="seconds" />
        <parameter key="DtFixed" value="0" comment="Fixed Dt value. Use 0 to disable
(default=disabled)" units_comment="seconds" />
        <parameter key="DtFixedFile" value="NONE" comment="Dt values are loaded from file. Use
NONE to disable (default=disabled)" units_comment="milliseconds (ms)" />
        <parameter key="DtAllParticles" value="0" comment="Velocity of particles used to calculate
DT. 1:All, 0:Only fluid/floating (default=0)" />
        <parameter key="TimeMax" value="7" comment="Time of simulation" units_comment="seconds" />

```

```

        <parameter key="TimeOut" value="0.05" comment="Time out data" units_comment="seconds" />
        <parameter key="PartsOutMax" value="1" comment="%/100 of fluid particles allowed to be
excluded from domain (default=1)" units_comment="decimal" />
        <parameter key="RhopOutMin" value="800" comment="Minimum rhop valid (default=700)"
units_comment="kg/m^3" />
        <parameter key="RhopOutMax" value="1200" comment="Maximum rhop valid (default=1300)"
units_comment="kg/m^3" />
        <parameter key="XPeriodicIncZ" value="0" comment="Increase of Z with periodic BC" />
        <simulationdomain comment="Defines domain of simulation (default=Uses minimum and maximum
position of the generated particles)">
            <posmin x="default" y="default" z="default" comment="e.g.: x=0.5, y=default-1,
z=default-10%" />
            <posmax x="default" y="default" z="default" />
        </simulationdomain>
    </parameters>
</execution>
</case>

```

Appendix B: Bash file for SPH simulation

```
#!/bin/bash
fail () {
  echo Execution aborted.
  read -n1 -r -p "Press any key to continue..." key
  exit 1
}
# "name" and "dirout" are named according to the testcase
export name=af
export dirout=${name}_out
export diroutdata=${dirout}/data
# "executables" are renamed and called from their directory
export dirbin=/nfs/speed-scratch/a_fazlol/linux
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${dirbin}
export gencase="${dirbin}/GenCase_linux64"
export dualsphysicscpu="${dirbin}/DualSPHysics5.0CPU_linux64"
export dualsphysicsgpu="${dirbin}/DualSPHysics5.0_linux64"
export boundaryvtk="${dirbin}/BoundaryVTK_linux64"
export partvtk="${dirbin}/PartVTK_linux64"
export partvtkout="${dirbin}/PartVTKOut_linux64"
```

```

export measuretool="${dirbin}/MeasureTool_linux64"
export computeforces="${dirbin}/ComputeForces_linux64"
export isosurface="${dirbin}/IsoSurface_linux64"
export flowtool="${dirbin}/FlowTool_linux64"
export floatinginfo="${dirbin}/FloatingInfo_linux64"
# "dirout" to store results is removed if it already exists
if [ -e ${dirout} ]; then rm -r ${dirout}; fi
# CODES are executed according to the selected parameters of execution in this testcase
${gencase} ${name} ${dirout}/${name} -save:all
if [ $? -ne 0 ]; then fail; fi
${dualsphysicsgpu} ${dirout}/${name} ${dirout} -dirdataout data -svres -gpu
if [ $? -ne 0 ]; then fail; fi
export dirout2=${dirout}/particles
${partvtk} -dirin ${diroutdata} -savevtk ${dirout2}/PartFluid -onlytype:-all,fluid -vars:+idp,+vel,+rhop,+press,+vor
if [ $? -ne 0 ]; then fail; fi
# Executes MeasureTool to create VTK files with velocity and a CSV file with velocity at each simulation time.
export dirout2=${dirout}/measuretool
${measuretool} -dirin ${diroutdata} -points Points.txt -onlytype:-all,+fluid -vars:-all,+vel -savecsv ${dirout2}/_PointsVelocity
if [ $? -ne 0 ]; then fail; fi
# Executes MeasureTool to create VTK files with pressure and a CSV file with pressure at each simulation time.
export dirout2=${dirout}/measuretool

```

```

${measuretool} -dirin ${diroutdata} -points Points.txt -onlytype:-all,+fluid -vars:-all,+press -savecsv ${dirout2}/_PointsPressureV
if [ $? -ne 0 ] ; then fail; fi
# Executes MeasureTool to create VTK files with pressure and a CSV file with pressure at each simulation time.
export dirout2=${dirout}/measuretool
${measuretool} -dirin ${diroutdata} -points Points.txt -onlytype:-all,+fluid -vars:-all,+press -savecsv ${dirout2}/_PointsPressureH
if [ $? -ne 0 ] ; then fail; fi
export dirout2=${dirout}/measuretool
${measuretool} -dirin ${diroutdata} -points Points.txt -onlytype:-all,+fluid -height:0.5 -savecsv ${dirout2}/_heights
if [ $? -ne 0 ] ; then fail; fi
# Executes IsoSurface to create VTK files with surface fluid and slices of surface.
export dirout2=${dirout}/surface
export planesy="-slice3pt:0:-0.4:0.01:0:0.4:0.01:1:0.4:0.01 -slice3pt:0:-0.4:0.03:0:0.4:0.03:1:0.4:0.03 -slice3pt:0:-0.4:0.06:0:0.4:0.06:1:0.4:0.06"
${isosurface} -dirin ${diroutdata} -vars:-all,+vel,+rhop,+idp,+vor,+press,+ace -saveiso ${dirout2}/Surface -saveslice
${dirout2}/Slices ${planesy}
if [ $? -ne 0 ] ; then fail; fi
# Executes FlowTool to create VTK files with particles assigned to different zones and a CSV file with information of each zone.
export dirout2=${dirout}/flow
${flowtool} -dirin ${diroutdata} -fileboxes FileBoxes.txt -savecsv ${dirout2}/_ResultFlow.csv
if [ $? -ne 0 ] ; then fail; fi
echo All done

```

```
read -n1 -r -p "Press any key to continue..." key
```