TECHNISCHE UNIVERSITÄT MÜNCHEN

Professur für Hydromechanik

An Open Source 2D-3D Coupled Model for flood simulations

Hao Zeng

Vollständiger Abdruck der an der Fakultät für Bauingenieur- und Vermessungswesen der Technischen Universität zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender: Prof. Dr.rer.nat. Ernst Rank
Prüfer der Dissertation:

1. Prof. Dr.-Ing. Michael Manhart
2. Prof. Dr. Sc. Lado Kranjčević

Die Dissertation wurde am 24.06.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Bauingenieur- und Vermessungswesen am 04.11.2020 angenommen.

Abstract

The main objective of this work is to develop a multi-scale flow simulation tool for flood assessment in complex environmental settings, like urban environments. We intend to predict floods and impact of floods on infrastructure within a multi-scale framework reaching from the river down to the scale of the built infrastructure, such as railways, subways, tunnels, waste water channels, buildings, and other infrastructures. A numerical model is presented which combines the 2D Shallow Water equations and the 3D Reynolds-Averaged Navier-Stokes (RANS) equations with fairly good efficiency and accuracy. This coupled model takes advantage of the 2D and 3D models to speed up the calculations while maintaining adequate accuracy: The simulation domain is mainly described with the 2D model, while the 3D model is only applied in small regions where the hydrodynamic effects need to be described by a three-dimensional solution.

The coupled model, `shallowInterFoamOL`, is implemented in the open source framework OpenFOAM by merging two existing OpenFOAM numerical solvers: `shallowFoam`, a 2D Shallow Water solver and `interFoam`, a 3D Navier-Stokes solver. The computational domain is decomposed into a set of smaller regions, each region is governed by the corresponding sub-solvers (*i.e.* `interFoam` or `shallowFoam`). Solutions at the coupling part are exchanged and updated via the coupling algorithm. The overlapping coupling algorithm uses a simplified cell-to-cell overlapping method for data exchange between the sub-domains.

The quality of the model is assessed for the transport of waves including solitary waves in both upstream and downstream directions and upstream-travelling surges. The results of the coupled model are compared with the pure 2D and the 3D model. An extreme hydrodynamic scenario, a tsunami, is also set up to further investigate and quantify the hydraulic bore-structure interactions. The numerical results are compared to experimental measurements.

Finally, workflow for putting the model into practice is proposed to allow users get into the model easily. The workflow enables users to generate the numerical mesh, set up the cases, run the coupled model and perform some basic post-processing operations. A risk analysis framework could then be developed on this basis for civil engineers, architects, planners, insurance analysts, and politicians. The user of that framework will become a part of the system or the iteration loop, allowing him or her to interactively explore and evaluate various scenarios and, thus, enabling decision makers to decide based on informations with a new level of detail and accuracy.

Zusammenfassung

Das Hauptziel dieser Arbeit ist die Entwicklung eines mehrskaligens Simulationswer-kzeugs zur Simulation von Hochwasserszenarien in komplexen Umgebungen wie dem städti-schen Raum. Überschwemmungen und Auswirkungen von Überschwemmungen auf die Infrastruktur sollen in einem mehrskaligen Rahmen vorhergesagt werden, der vom Fluss bis hinab zur Größe der gebauten Infrastruktur reicht, z. B. Eisenbahn, U-Bahn, Tunnel, Abwasserkanäle, Gebäude und Gebäudeinfrastrukturen. Ein numerisches Modell wird vorgestellter, das die 2D Flachwassergleichungen und die 3D-Reynolds-gemittelten Navier-Stokes (RANS) Gleichungen mit guter Effizienz und Genauigkeit kombiniert. Dieses gekoppelte Modell nutzt die Vorteile der 2D- und 3D-Modelle, um die Berechnungen zu beschleunigen, ohne an Genauigkeit zu verlieren: Das Simulationgebiet wird hauptsächlich durch das 2D-Modell abgebildet, während das 3D-Modell nur in kleinen Bereichen eingesetzt wird, in denen eine dreidimensionale Beschreibung der hydrodynamischen Phänomene erforderlich ist.

Das gekoppelte Modell `shallowInterFoamOL` wird im OpenFOAM implementiert, indem zwei vorhandene numerische OpenFOAM-Löser zusammengeführt werden: Der 2D-Flachwasserlöser `shallowFoam` und der 3D-Navier-Stokes Löser `interFoam`. Das Berechnungsgebiet wird in eine Reihe kleinerer Teilgebiete zerlegt, wobei jedes Teilgebiet mit Hilfe der entsprechenden Unterlöser (*i.e.* `interFoam` oder `shallowFoam`) gesteuert wird. Lösungen am Kopplungsteil werden über den Kopplungsalgorithmus ausgetauscht und aktualisiert. Der überlappende Kopplungsalgorithmus verwendet ein vereinfachtes Überlappungsverfahren von Zelle zu Zelle für den Datenaustausch zwischen den Unterdomänen.

Die Gültigkeit des Modells wird für den Wellentransport einschließlich solitonen und stromaufwärts laufender Schwallwellen bestimmt. Die gekoppelten Ergebnisse werden mit dem reinen 2D-Flachwassermodell und dem 3D-RANS-Modell verglichen. Außerdem wird ein hydrodynamisches Extremszenario, ein Tsunami, simuliert, um die Wechselwirkungen zwischen einer hydraulischer Bore und einer Struktur weiter zu untersuchen und zu qualifizieren. Numerische Ergebnisse werden mit den experimentellen Messungen verglichen.

Schließlich wird ein Arbeitsablauf zur praktischen Anwendung des Modells vorgeschlagen, um Benutzern einen leichten Einstieg zu ermöglichen. Der Arbeitsablauf ermöglicht es Benutzern, das Berechnungsnetz zuerzeugen, die Fälle einzurichten, das gekoppelte Modell auszuführen und einige grundlegende Postprocessingschritte durchzuführen. Auf dieser Grundlage könnte dann ein Werkzeug für die Risikoanalyse für Bauingenieure, Architekten, Planer, Versicherungsanalysten und Politiker entwickelt werden. Der Benutzer dieses Frameworks wird Teil des Systems oder der Iterationsschleife, sodass sich er verschiedene Szenarien interaktiv untersuchen und bewerten kann, sodass Entscheidungsträger auf Informationen größerem Detailreichtums und höherer Genauigkeit stützen können.

Acknowledgement

I would like to thank and express my gratitude to the following people for supporting fuelling my interest live during my four years of Ph.D. research:

My supervisor Prof. Dr.-Ing. habil. M. Manhart from the Technical University of Munich of who in spite of being extraordinarily busy with his duties, took time out to hear, guide and keep me on the correct path.

My mentor Prof. Dr.-Ing. O. Delestre from the University of Nice Sophia Antipolis for the continuous support, encouragement, insightful comments of my Ph.D. study and research.

The previous team leader Dr.-Ing. F. Mintgen who has finished his Ph.D. from the Technical University of Munich that helps me a lot during the first two years for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me especially for starting this Ph.D. study. The successive team leaders Dr.-Ing. C. Strobl and Dr.-Ing. D. Quosdorf from the same institute managed the administration in well organization.

Nevena Perović, from the Chair for Computation in Engineering (TUM) mainly works in high-performance computing and numerical simulations of fluid flows.We collaborate with each other during the first two years in mesh generation.

My dear husband J. Zhang supports me the most when I was depressed and inspirit me to move on. He also helps me quite a lot in English grammar's checking of the dissertation.

I express my deepest thanks to the entire colleagues of the chair of Hydromechanics from the Technical University of Munich for their encouragement, insightful comments, and hard questions.

Thanks to the TUM International Graduate School of Science and Engineering (IGSSE) (project 9.11 - High Performance Interactive Flood Simulation - iFlood) and the German Academic Exchange Service (DAAD) (funding program Graduate School Scholarship Programm 2016, ID number 57437801) for their financial support.

Contents

List of Tables

List of Figures

Notations

Roman Letters

k_{st}	Strickler coefficient, $\mathrm{m}^{1/3} \cdot \mathrm{s}^{-1}$
A	Wave amplitude, m
B	Width, m
C_{D}	Drag coefficient, N
f	Body force, $\mathrm{m} \cdot \mathrm{s}^{-2}$
F_{d}	Hydrodynamic force, N
Fr	Froud number, $-$
Fr_{rel}	Relative Froude numbers, $-$
g	Gravity, 9.81 $\mathrm{m} \cdot \mathrm{s}^{-2}$
h	Flow depth, m
hu	Specific discharge, $\mathrm{m}^2 \cdot \mathrm{s}^{-1}$
k	Turbulent kinetic energy, $\mathrm{m}^2 \cdot \mathrm{s}^{-2}$
n	Manning coefficient, $\mathrm{s} \cdot \mathrm{m}^{-1/3}$
p	Pressure, $\mathrm{kg} \cdot \mathrm{m}^{-1} \cdot \mathrm{s}^{-2}$
p_{d}	Dynamic pressure, $\mathrm{kg} \cdot \mathrm{m}^{-1} \cdot \mathrm{s}^{-2}$
R_{h}	Hydraulic radius, m
t	Time, s
u	Magnitude of flow velocity, $\mathrm{m} \cdot \mathrm{s}^{-1}$
x	Cartesian coordinates, m
z_{b}	Bottom elevation, m

Greek Letters

α_1	VOF function indicator, $-$
$\bar{\phi}$	Ensemble averaging quantity, $-$
δ_{d}	Distance from the cell center to the face center
δ_{ij}	Kronecker delta
Γ	Coupling interface
Γ_{2D}	2D side if the coupling interface

Γ_{3D}	3D side if the coupling interface
κ	von-Karman's constant, $-$
μ	Position of the wave center, m
ν	Kinematic viscosity, $\mathrm{m}^2 \cdot \mathrm{s}^{-1}$
ν_{t}	Eddy viscosity, $\mathrm{m}^2 \cdot \mathrm{s}^{-1}$
Ω	Global domain
ω	Phase fraction value
Ω_{2D}	2D region
Ω_{3D}	3D region
ϕ	Physical quantity, $-$
$\phi^{'}$	Fluctuating component of the quantity from ensemble averaging value, $-$
Φ_{c}	Cell center value
Φ_{f}	Face center value
Φ_{if}	Interface value
Φ_{ref}	Reference cell center value
ρ	Flow density, $\mathrm{kg} \cdot \mathrm{m}^{-3}$
σ	Wave length parameter, $-$
τ	Shear stress, $\mathrm{kg} \cdot \mathrm{m}^{-1} \cdot \mathrm{s}^{-2}$
τ_{B}	Bottom stress, $\mathrm{kg} \cdot \mathrm{m}^{-1} \cdot \mathrm{s}^{-2}$

Subscripts

i,j	Index number with 1,2,3 respectively in 3D flow and with 1,2 respectively in 2D flow

Abbreviations

1D/2D/3D	One-/Two-/Three-Dimensional
CFD	Computational Fluid Dynamic
CPU	Central Processing Unit
CV	Control Volume
FSI	Fluid-Structure Interaction
FVM	Finite Volume Method
GPU	Graphic Processing Unit
HPC	High Performance Computing
LRZ	Leibniz-Rechenzentrum (Leibniz Supercomputing Centre)
MPI	Message Passing Interface

MPP	Massively Parallel Processing
OpenFOAM	Open source Field Operation And Manipulation
PDE	Partial Differential Equations
PISO	Pressure Implicit with Splitting of Operators
RANS	Reynolds-Averaged Navier-Stokes Equations
SWE	Shallow Water Equations
VOF	Volume of Fluid

1. Introduction

Free surface flows, such as water waves, surface run-off, channel flow, and effluent discharge, are important physical phenomena in environmental fluid mechanics which is comprised of two-phase immiscible fluids (*e.g.* liquid water and air) separated by a well-defined interface (Katopodes (2018)). Sometimes these flows can bring a dramatic impact on our daily lives. Among a wide range of free surface flow scenarios, flooding is one of the most damaging natural hazards to human societies (Schanze (2006)). Understanding and predicting flood behaviours are of great interests in science and technology. Computational fluid dynamics (CFD) is an efficient numerical analysis that provides the ability to simulate fluid performance.

The present work is embedded with project No. 9.11 - High Performance Interactive Flood Simulation - iFlood, from the Technical University of Munich (TUM) International Graduate School of Science and Engineering (IGSSE) with the Chair of Hydromechanics (TUM), the Chair for Computation in Engineering (TUM) and the School of Engineering and Computing Science (Durham University). This chapter describes the background and the motivation for flood prediction by using numerical methods. The research objectives and the scope of the study are provided here as well.

1.1. Background

A flood occurs when a water resource (*e.g.* rivers, lakes and oceans) gains excessive water in a short time such that an overflow of water is generated in an area which is usually dry. Floods happen all over the world for various reasons. An effective way to establish flood control by hydraulic structures *e.g.* dams or reservoirs, is therefore one of the most basic hydraulic engineering tasks. These structures, which can be built in any water body, are used to control the fluid flow velocities and depths in dams, spillways, overflow weirs, dikes and so on. As reported in Munich Re Group (2005), flood events have increased in recent decades, both in terms of magnitude and frequency due to a combination of natural and man-made factors. Human causes of floods, for instance, include poorly designed hydraulic infrastructures. On the other hand, varying degrees of natural hazards, *e.g.* extreme precipitation, high sea water levels and snow melt, may also increase flooding risk (Pedersen et al. (2012)).

Extreme hydrological events are likely to produce an increasing number of flooding occurrences resulting in significant losses. Heavy rain is the most common factor leading to a flood event in urban areas. Large parts of many cities are made of concrete or other impermeable materials, so run-off water is mainly collected by rainwater gullies and drained into city sewage systems. During a longer period of heavy rain, such systems are overwhelmed, which means the rainwater cannot be transferred to a wastewater treatment plant fast enough. Consequently, the drainage pipes fill quickly and the rainwater spreads across the city, resulting in floods. Heavy rain also can cause dam breaks, which are another direct reason for urban floods. When dams fail, a huge amount of water rushes into rivers and may flow into nearby cities.

The potential damage of floods in urban areas is so high that urban flooding is an important consideration for city planners. The deadliest flood ever took place in China (1931) and killed between 2,000,000 and 4,000,000 people (O'Connor and Costa (2004)). As a consequence, a sustainable urban flood risk management tool is of great value for city managers. Flood risk management is a well-established process for dealing with flood risks and for rebuilding areas at risk in order to reduce losses (Plate (2002)). The most effective approach for flood risk management is incorporating several elements covering before, during and after flood periods: (1) preventing floods caused damage by optimisation of houses and industries location; and by facilitating land-use planning; (2) protecting potential flooding area by building hydraulic constructions; (3) educating the masses about relevant knowledge of floods; (4) setting up an emergency plan during the flood and returning back to ordinary life as soon as possible (COM (2004)).

Studying free surface flow plays a significant role in the first two elements of flood risk management mentioned above which are damage prevention and flooding protection. Flooding flow can have a direct impact on the design of hydraulic structures (Katopodes (2018)). The design of hydraulic structures varies depending on the goals of the hydraulic projects (Chen (2015b)). By means of establishing flood models, the reaction between structures can be analysed and the flooding area can be estimated. In general, there exist two approaches to model a flood event: physical experiment and numerical simulation. Experiments are usually carried out on a scaled model. However, establishing a physical complex model is too time consuming. Numerical simulations offer an alternative to experiments and are widely employed for the efficient solution of problems in flooding investigations under a broad range of configurations, known as computational fluid dynamics (CFD). Applying CFD to flood simulations enables researchers to reproduce or predict the movement of flood waves and helps them to analyse drag and lift forces on structures which are necessary to storm, and flood assess risk. The choice of a specific numerical model depends on the problem under examination.

1.2. State of the art

CFD solves problems numerically based on governing equations of fluid dynamics. These mathematical models describe fluid physical properties in accordance with the conservation laws for the mass, momentum, and energy. Typically, mathematical models are categorised into three types: one-dimensional (1D), two-dimensional (2D) and three-dimensional (3D). The fundamental mathematical models are 3D since nature flows are 3D. However, a 3D model is difficult to solve, and computationally expensive. For some simplified cases which can be estimated to 2D or even 1D flows, low-dimensional models are used to reduce the computational effort. Coupled models among 1D, 2D or 3D models have been investigated for some specific applications to speed up the calculations without losing adequate accuracy.

1.2.1. Basic mathematical models

1D Flood waves can be described by the Saint-Venant equations named after a French engineer de Saint-Venant (1855). The 1D Saint-Venant equations system is derived from the 3D Navier-Stokes equations for very shallow flows under certain assumptions: (1) the flow is 1D, incompressible and homogeneous; (2) the pressure distributions are hydrostatic; (3) vertical velocity components are neglected; (4) the bed slope is constant and relatively small; (5) the streamline curvature is small; (6) fixed boundaries are considered with no erosion and deposition. Dressler (1978) derived an extended approach to the Saint-Venant equations by introducing curvilinear coordinates system based on the river beds. This method is employed by Sivakumaran et al. (1983) for steady shallow flow and by Sivakumaran et al. (1983) for unsteady shallow flow over a curved boundary. Fenton and Nalder (1995) applies the same conception in the Saint-Venant model to capture the effects of channel curvature on the propagation of floods and long waves. Hager and Hutter (1983) has implemented a method based on potential flow in a streamline coordinate system and a higher-order method incorporating friction effects was developed by Matthew (1991) in Cartesian coordinate system. Another modified model with a force-corrected term has been proposed by Zhang and Bao (2012) enhancing the ability to simulate tidal rivers. Correction coefficients combining flow velocity and the change rate of the tidal level have been applied to the force term. Parameters in 1D models are averaged over the water depth and the cross section (*e.g.* hydraulic diameters and mean flow velocities) where the geometry is typically represented by a set of cross sections. 1D models (*e.g.* `HEC-RAS`, `MIKE 11`) predict well for modelling constant and well-defined flow path. For example, the main flow path is carried by a large river with minimal tributaries. When a flow path is poorly defined, a multi-dimensional model is needed.

The Shallow Water equations (SWE), 2D approaches, are widely used in current urban flood simulation assessment tools, *e.g.* Bradford and Sanders (2002) and Simões (2011). The 2D model, based on integration of the 3D Navier-Stokes equations in the vertical direction, is derived by taking the first three assumptions in the Saint-Venant equations derivation mentioned above. Detailed representation of the geometry is considered in 2D models contributing to better estimations in frictional losses and inundation extents. 2D models are also able to capture flood dynamics in topographically complex floodplain. For non-hydrostatics flow, the Serre-type equations firstly derived by Serre (1953) and then Dias and Milewski (2010) presented a fully-nonlinear weakly dispersive system for the shallow water wave regime. Since the SWE have no ability to represent short waves, the Boussinesq equations are applicable to slightly shorter length waves (Boussinesq (1872)). 2D models (*e.g.* `ISIS 2D`, `MIKE FLOOD` and `HYDRO AS-2D`) reproduce accurately the surface run-off phenomenon in large river basins, with complex bed shapes and meandering open channels. However, it cannot capture depth varied velocities around structures. When the flow interacts with structures, higher numeric accuracy is demanded, and more properties shall be calculated.

The basic equation system of 3D flows model is the Navier-Stokes equations which is applicable to all flow problems. It can handle complex topography and free surface deformations. With a focus on small scales analysis 3D models (*e.g.* `FLOW 3D`, `MIKE 3` and `FLUENT`) are required to represent the features of flow around structures. Nevertheless, due to the high computational effort of solving these complexities, a full 3D simulation is rarely applied in flood simulations.

1D/2D/3D models have been developed for different usages. Approximate predictions can be obtained for large scales by using 1D/2D models. For practical purposes of flood simulation, flooding flows in most rural areas are considered as 1D flows, therefore 1D models is adequate to delineating floodplains and assess risk regions in such areas. Most urban flood models employ 2D surface run-off models. 3D models are more likely to assess local dominant fluid-structure regions. Each model has its purpose of applications, but in reality, some target applications are overlapped with multi-scales problems, which require 1D, 2D or 3D model at the same time. Coupled models are therefore developed for those specific applications. Coupled hydrodynamic numerical methods are often used in urban floods by taking an expensive method regarding time consumption (Sailor (2013)).

1.2.2. Coupled models

With the objective to improve the simulation results within a reasonable cost, lower dimensional models are commonly coupled with higher dimensional models in many fields as well as in fluid mechanics simulations. The coupling between 1D and 2D models are typical approaches in urban flood models. Various 1D-2D coupling approaches have been developed in the last decades.

On the basis of 1D channel flow simulations, 2D Shallow Water models can be coupled where the domain is located laterally to the 1D channels. Such a coupling approach is suitable for flood inundation modelling. Kuiry et al. (2010) have implemented a quasi 1D-2D flood inundation model that 1D and 2D regions have been coupled via Manning's equation. Ahmadian et al. (2015) have achieved the coupling via a weir formula. The coupling can also be accomplished by superposing a 2D domain on top of 1D results. D'Alpaos and Dena (2007) have employed the continuity equation from both domains. Gejadze and Monnier (2007) have added a source term to transfer the information from 2D to 1D and boundary conditions are imposed from 1D to 2D. Coupling approaches mentioned above are mainly applied in river-floodplain modelling.

When 1D and 2D regions are solved independently from each other, two regions are coupled via boundary conditions, a longitudinal coupling is therefore defined. The fundamental principle is the conservation law and these variety of models are eligible for solving the river-lake or river-estuary systems. Chen et al. (2012) conserves the continuity equations at the coupling interface and in addition to that Blade et al. (2012) conserves the momentum equations. Urban drainage systems can be coupled into the 2D simulations to represent urban floods, such as the coupling of 1D-sewer and 2D-surface models (Leandro et al. (2009), Schlauß and Grottker (2016) and Adeogun et al. (2015)).

In addition to 1D-2D couplings, there exist 2D-3D coupling algorithms allowing the coupling between the interaction of the surface and groundwater. Furthermore, the coupling between 2D-3D surface flow models are becoming popular recently. A 2D Boussinesq model with a 3D Reynolds-Averaged Navier-Stokes Equations (RANS) model have been coupled by Qi and Hou (2004) using an overlapping decomposition method. A direct matching between water depths and flow velocities are imposed via the overlapping region. This approach saves most of the computational expense with respect to a full 3D RANS model in a good agreement with the velocity profiles of the 3D model. However, the solver remains incomplete in the description, and no further validation data has been published.

Another common 2D-3D coupling approach is coupling the 2D Shallow Water solver with the 3D Naiver-Stokes solver. Kilanehei et al. (2011) have implemented a coupling method by taking the 2D SWE results as initial and boundary conditions of a 3D RANS solver. These boundary conditions defined by 2D results help the 3D solver converge fast even in realistic applications. Nevertheless, the feedback from 2D to 3D is not available and only steady-state can be modelled under this approach. To improve the aforementioned coupled method, the backwater effects from 3D to 2D has been taken into account by coupling the 2D SWE solver `Hydro AS-2D` and the 3D free surface RANS solver `interFoam` by Gerstner et al. (2014). Apart from using 2D results as boundary conditions, 3D results will be updated on the 2D model by introducing an overflow coefficient. Unfortunately, this method is also only capable of steady-state conditions.

A full bi-directional coupling between 2D SWE and 3D RANS enables simulating unsteady phenomena, which combines the 2D Shallow Water solver `shallowFoam` with the 3D RANS solver `interFoam` by Mintgen and Manhart (2018). These solvers are available as an Open Source software in the OpenFOAM framework. The coupling algorithm uses combinations of Dirichlet-Neumann boundary conditions at the 2D-3D interface depending on the flow situations and it performs well in most application scenarios. However, an upstream-travelling shock wave is unable to cross the 2D-3D interface due to the direction of the characteristics in supercritical flows. This approach (details can be found in chapter 3 and 4) is the one that is used in this work as a standard coupling method. Within the collaboration of the iFlood project, N. Perovic who works at the Chair for Computation in Engineering (TUM) have also studied the flow coupling and contributed to high performance computing simulations of large flooding events (Perovic et al. (2017)).

1.3. Problems formulation

A good flood assessment model should offer a flood risk evaluation in an efficient and accurate manner, which covers flood waves propagating simulation, maximum water levels calculation, flooded areas and waves arriving time estimation. This is mainly determined by setting up a numerical flood simulation.

A complete numerical flood simulation usually contains three procedures: mesh generation, problem-solving and results analysis. Problem-solving is the core for each simulation which means solving the target model numerically. 3D models are too slow and 2D models are not accurate enough for flood assessment in complex environments, such as urban areas. The 2D-3D coupled model would be of great value for establishing an urban flood risk assessment tool. However, the coupled models on the markets have severe limitations. An integrated prediction of local flows together with large scale flow dynamics has rarely been established, mainly because of intrinsic differences between the character of the depth integrated 2D model and the full 3D RANS. A coupling of the Shallow Water model with a full solution of the RANS model is needed in cases when the flow interacts with buildings, bridge piers, or bridge abutments for instance. Such 2D-3D coupling code should capture different scale flow dynamics.

Apart from identifying the problem for coupled model, grid generation and adaptation should also be considered in mesh generation. Generally, meshes have different representation for 2D and 3D models, a 2D-3D coupling mesh should be determined by seeking for the most suitable and efficient grid structure to be used in a coupled 2D-3D solve. In this study, grids at the coupling region are constrained to be perfectly matched in horizontal directions (details can be found in chapter 4). Current prototype of mesh generation are exclusive for easy and regular geometry. Hence special treatments would be expected to build up a coupling case. Thus, a new workflow for the coupled model is required.

In addition to mesh generation and problem-solving parts, post processing mainly including data analysis remains vapoury. In order to better analyse the outcomes and make the results in a more readable way, data filtration and extraction for desired locations from simulating solutions are essential at the workflow.

1.4. Objective of this work

The overall goal of this research is to develop, implement, and validate a prototype for urban flood scenarios based on multi-scale 2D and 3D fluid flow simulations in order to evaluate, for instance, possible impacts on built infrastructure, such as railway, subway, tunnels, waste water channels, buildings, and building infrastructures. Therefore, different aspects such as efficient interaction / exploration methods, grid generation and adaptation, and the 2D-3D coupling must be studied in detail in order to handle the computational complexity of the underlying problem and to make it feasible for an interactive treatment.

1.4.1. Development of an improved method for 2D-3D coupling

In the latest research by Mintgen and Manhart (2018), a coupling of a 2D SWE and a 3D RANS solver with free surface has been implemented in the Open Source CFD environment OpenFOAM. This coupled solver can handle most of the open channel flow simulations. But it has no ability to track the hydraulic bore when the upstream-travelling wave is approaching to the 2D-3D simulation interface, which leads to a wrong simulation result. Such upstream-travelling wave is an important phenomenon in flooding events which should not be negligible. The work focuses on developing an improved method for a better treatment for upstream-travelling shock waves.

1.4.2. Workflow development

All relevant flow models used in this work are implemented in OpenFOAM framework and the outcome of this research will also be available as an open source package for publics. In conformity with the OpenFOAM framework, mesh generation and results analysis have also to be done by using OpenFOAM compatible utilities.

The coupling algorithm developed by Mintgen and Manhart (2018) has only been tested and valid in simple geometry cases (*e.g.* rectangular open channel with constant slope and a cubic structure located on a flat plane). However, the mesh from complicated geometry leads to an unmatched grid structure at the coupling region thus the coupled solver is no longer applicable. As a result, a well-suited method for mesh generation should be determined as a part of the coupling workflow.

2D solver and 3D solver have different configurations for numerical variables. How to set up a realistic application with more complex data for each solver is therefore another issue to be tackled.

1.5. Outline

This study provides a systematic analysis tools for environmental flows representation. The present work is organised into seven chapters.

In chapters 2 and 3, literature reviews of mathematical equation systems and numerical models are briefly presented. Chapter 2 discusses mathematical modelling of 3D and 2D flows. It starts with the explanation of the full 3D Navier-Stokes problems with focus on two-phase flows, and then continues with the simplified 2D Shallow Water models which are built upon certain of assumptions and simplifications from the 3D flow models.

Chapter 3 further explains the numerical discretization procedure of the derived equation systems which are depicted in the previous chapter. First, the chosen CFD framework, OpenFOAM, will be introduced as the foundation of this study. On top of it, highlighting techniques are put forward: a spatial discretization approach - the Finite Volume Method, interface tracking techniques with emphasis on the Volume of Fluid and the special technique pressure-velocity coupling to solve inter-equation coupling. Accordingly, the implementation of different flow models and the set up for these models in OpenFOAM are detailed.

The following chapter (chapter 4) elaborates the progress in 2D-3D coupling algorithms about balance accuracy and computational efficiency. A published coupling method developed by Mintgen and Manhart (2018) exchanges information at the 2D-3D boundaries by determining flow characteristics. An open issue of the coupling method concerning an upstream-travelling wave remains to be solved. For this reason, the improvement of the coupled 2D-3D models is investigated in this study together with test cases, where an upstream-travelling wave can be modelled correctly by applying the modified coupled model.

Test cases for quantitative validation and comparison of various numerical models are presented in chapter 5. In chapter 6, a tsunami-like hydraulic bore interacts with a square structure is conducted and outcomes from the proposed coupled model are compared with experimental data as well as 2D and 3D numerical results.

Chapter 7 devotes to develop a practical workflow with respect to the coupled model. With the help of this workflow, realistic scenario applications can be put into practice by using the coupling code.

The thesis is concluded in chapter 8 with the benefits and drawbacks of the proposed coupling code. Due to the modification of the coupling method and the time limit, the high-performance computing of the coupling code is still required to be implemented. The outlook for further extension of discussion on methodologies which may have a future potential to improve the capacity for the coupling numerical simulation are discussed at the end of this chapter.

Appendices are provided afterwards as well as the bibliography list.

2. Mathematical Models

This chapter will introduce mathematical models used in this work. A typical mathematical model contains governing equations, supplementary sub-models and assumptions. Fluid properties can be defined as a set of variables in mathematical models to describe the flow behaviour. Section 2.1 covers the equation system for general flows and turbulence. Section 2.2 describes the simplification of 3D flow systems, named as the 2D shallow flows, which are widely used in large simulation areas. None of these equations have analytical solutions, the following chapter (chapter 3) will present numerical methods which are required to solve these problems.

2.1. 3D flow models

The Navier-Stokes equations system is a set of partial differential equations (PDE) describing the dynamics of fluids mathematically, which are derived by Navier (1821) and Stokes (1880). These equations are expressed based on laws of conservation: mass, momentum and energy. In the present study, we focus on incompressible fluids and there is no relevant heat transfer in the flow. Consequently, energy conservation equation can be neglected and only conservation of mass and momentum are taken into account.

2.1.1. Navier-Stokes equations and turbulence

The free surface flow is considered as an incompressible Newtonian flow that can be governed by the full Navier-Stokes equations. The Navier-Stokes equations consist of the conservation of mass equation (2.1.1) and the conservation of momentum equation (2.1.2). These equations describe the relationship between velocity and pressure of a moving fluid.

$$\frac{\partial u_{\mathrm{i}}}{\partial x_{\mathrm{i}}} = 0 \quad (\mathrm{i} = 1, 2, 3) \tag{2.1.1}$$

$$\frac{\partial u_{\mathrm{i}}}{\partial t} + u_{\mathrm{j}}\frac{\partial u_{\mathrm{i}}}{\partial x_{\mathrm{j}}} = -\frac{1}{\rho}\frac{\partial p}{\partial x_{\mathrm{i}}} + \frac{\partial}{\partial x_{\mathrm{j}}}[\nu(\frac{\partial u_{\mathrm{i}}}{\partial x_{\mathrm{j}}} + \frac{\partial u_{\mathrm{j}}}{\partial x_{\mathrm{i}}})] + f_{\mathrm{i}} \quad (\mathrm{i}, \mathrm{j} = 1, 2, 3) \tag{2.1.2}$$

where t [s] is the time , u $[\mathrm{m} \cdot \mathrm{s}^{-1}]$ is the flow velocity, ρ $[\mathrm{kg} \cdot \mathrm{m}^{-3}]$ is the fluid density, p $[\mathrm{kg} \cdot \mathrm{m}^{-1} \cdot \mathrm{s}^{-2}]$ is the pressure, ν $[\mathrm{m}^2 \cdot \mathrm{s}^{-1}]$ is the kinematic viscosity and f $[\mathrm{m} \cdot \mathrm{s}^{-2}]$ is the body force, *e.g.* gravity. Subscripts i, j are Einstein summation convention indices vary from 1 to 3 representing three directions respectively.

Generally, the flows can be classified as laminar, transitional and turbulent subject to the Reynolds number (Reynolds (1883)). The dimensionless quantity Reynolds number formulated as $Re = \frac{uh}{\nu}$,with h [m] a characteristic length with respect to the object, is the ratio of inertial forces to viscous forces in the fluid flow. When inertial effects dominate resulting in a large Re, turbulent flow appears; and in contrast, when viscous effects dominates with a small Re, the flow will be laminar flow. The critical Re is defined where transition flow occurs. Most of flows in our natural environment are turbulent which is characterized by the irregular movement of particles of the fluid. Those flows are inherently unsteady, diffusive and fluctuating that fluid velocities vary highly and irregularly both in space and in time.

Analytical solution of the Navier-Stokes equations for turbulent flows is infeasible due to their inherently nonlinearity and complexity, which requires a very fine mesh and extremely large computational time. In order to predict the effects of turbulence, special treatment of turbulence is of great importance to solve the Navier-Stokes equations numerically. Several approaches are commonly applied to compute turbulent flows in nowadays.

Direct Numerical Simulation (DNS)

A numerical simulation without any turbulence model to solve the time-dependent Navier-Stokes equations resolving all spatial and temporal scales is called direct numerical simulation. In order to cover full scales of the turbulence, the computational box should be sufficient broad, however, the numerical grid and the time step must be fine enough. This method is well-known due to its high accuracy with all details of turbulent flows with low efficiency by requiring a great demand of computational costs. Details provided by direct numerical simulation are not required for most hydraulic engineering purposes and large scale resolution is enough for applications.

Large-Eddy Simulation (LES)

To speed up the computational efficiency, the large-eddy simulation is proposed by ignoring the smallest length scales. In the large-eddy simulation approach, the large energy-containing scales are resolved and the small scale effects should be modelled. In other words, it computes time-varying flow, but models sub-grid-scale motions for turbulence of scales which are smaller than the computational grid spacing. Large-eddy simulation has become one of the most promising and successful techniques for turbulent flows simulations to calculate complex engineering flows, which is currently applied in a wide variety of engineering applications.

Reynolds-Averaged Navier-Stokes (RANS)

A further less expensive model based on Reynolds decomposition, the Reynolds-Averaged Navier-Stokes (RANS) equations, are used to describe turbulent flows. They modelled the mean flow and give time-averaged mean values for velocity fields. RANS equations are chosen as the 3D flow model in this work since they offer the most economic approach for computing complex turbulent river flows. Elaborated description of the RANS equations will be given in the following.

2.1.2. Reynolds-Averaged Navier-Stokes equations

A time-averaged equations of motion for turbulent flow, the RANS equations, are widely used instead of solving the Navier-Stokes equations directly. RANS equations apply Reynolds decomposition with approximations of the turbulent properties and provide averaged solutions to the Navier-Stokes equations. As a consequence of Reynolds decomposition, flow quantities can be resolved into mean and fluctuating components, $\phi = \langle\phi\rangle + \tilde{\phi}$. More specifically, $\langle\phi\rangle$ denotes the expectation value of ϕ as a steady component and $\tilde{\phi}$ is the deviation from the expectation value (Adrian (2000)). By further time averaging the instantaneous flow variable, the Reynolds time-averaged quantity is derived.

Mass conservation equation

The incompressible continuity equation (2.1.1) in Navier-Stokes equations contains only linear terms. The Reynolds decomposition of flow velocity is expressed in equation (2.1.3), where $\langle u_\mathrm{i}(x_\mathrm{i})\rangle$ is the mean velocity in space, and it must be larger compared to the typical time scale of the fluctuations. $\tilde{u}_\mathrm{i}(x_\mathrm{i}, t)$ is the fluctuation about the mean value.

$$u_\mathrm{i}(x_\mathrm{i}, t) = \langle u_\mathrm{i}(x_\mathrm{i})\rangle + \tilde{u}_\mathrm{i}(x_\mathrm{i}, t) \tag{2.1.3}$$

Time averaging the fluctuation parts to right-hand side (RHS) of equation (2.1.3) is zero.

$$\langle \tilde{u}_\mathrm{i}(x_\mathrm{i}, t)\rangle = 0 \tag{2.1.4}$$

Therefore, equations (2.1.3) and (2.1.4) lead to the ensemble averaging interval in time and space of velocity.

$$\langle u_\mathrm{i}(x_\mathrm{i}, t)\rangle = \langle u_\mathrm{i}(x_\mathrm{i})\rangle \tag{2.1.5}$$

Applying the resulting Reynolds time averaging simplification form of the flow velocity to the equation (2.1.1) and easily gives the time-averaged continuity equation (2.1.6).

$$\frac{\partial\langle u_\mathrm{i}\rangle}{\partial x_\mathrm{i}} = 0 \quad (\mathrm{i} = 1, 2, 3) \tag{2.1.6}$$

Momentum conservation equation

Non-linear terms appear in the momentum equation (2.1.2) of the Navier-Stokes equations, thus additional terms - product of components of fluctuating velocities - are produced during the averaging technique. Time-averaging convective term yields equation

(2.1.7).

$$\langle u_{\mathrm{i}}(x_{\mathrm{i}},t)u_{\mathrm{j}}(x_{\mathrm{j}},t)\rangle = \langle u_{\mathrm{i}}(x_{\mathrm{i}})\rangle\langle u_{\mathrm{j}}(x_{\mathrm{j}})\rangle + \langle \tilde{u}_{\mathrm{i}}(x_{\mathrm{i}},t)\tilde{u}_{\mathrm{j}}(x_{\mathrm{j}},t)\rangle \tag{2.1.7}$$

The left hand side (LHS) of the averaged momentum equation (2.1.2) results in equation (2.1.8).

$$\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial t} + \langle u_{\mathrm{j}}\rangle\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} = \frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial t} + \frac{\partial\langle u_{\mathrm{i}}\rangle\langle u_{\mathrm{j}}\rangle}{\partial x_{\mathrm{j}}} + \frac{\partial\langle \tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle}{\partial x_{\mathrm{j}}} \tag{2.1.8}$$

The RHS in the momentum equation (2.1.2) is simply taking an average, this leads to the RANS momentum equation (2.1.9).

$$\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial t} + \langle u_{\mathrm{j}}\rangle\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} = -\frac{1}{\rho}\frac{\partial\langle p\rangle}{\partial x_{\mathrm{i}}} + \frac{\partial}{\partial x_{\mathrm{j}}}[\nu(\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} + \frac{\partial\langle u_{\mathrm{j}}\rangle}{\partial x_{\mathrm{i}}}) - \langle \tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle] + \langle f_{\mathrm{i}}\rangle \quad (\mathrm{i},\mathrm{j}=1,2,3) \tag{2.1.9}$$

Hence the well-known non-conservative form the RANS equations are formulated into equations (2.1.6) and (2.1.9).

The Reynolds averaging process creates additional terms and no further equations. Additional terms, $\langle \tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle$, named the Reynolds-Stress. There are six independent elements of the Reynolds stress tensor and therefore six more unknowns in the above equations which must be modelled to close the RANS system.

2.1.3. Turbulence models in 3D flows

The Reynolds stresses are components of a symmetric second-order tensor. The diagonal components of the Reynolds stresses are known as normal stresses and the off-diagonal components are shear stresses. In order to predict turbulence effects, the Reynolds stresses can be modelled in terms of known quantities by turbulent modelling. More parameters are defined in turbulence model. Boussinesq (1877) introduced the turbulent-viscosity hypothesis that the momentum transfer caused by turbulent eddies can be modelled via an eddy viscosity which is the basis of many turbulence models (Rodi (1993)). The Reynolds stresses can be expressed by the turbulent eddy viscosity. $\nu_{\mathrm{t}}(x_{\mathrm{i}},t)$ $[\mathrm{m}^2\cdot\mathrm{s}^{-1}]$

$$-\langle \tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle = \nu_t(\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} + \frac{\partial\langle u_{\mathrm{j}}\rangle}{\partial x_{\mathrm{i}}}) \tag{2.1.10}$$

The momentum equation (2.1.9) then becomes:

$$\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial t} + \langle u_{\mathrm{j}}\rangle\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} = -\frac{1}{\rho}\frac{\partial\langle p\rangle}{\partial x_{\mathrm{i}}} + \frac{\partial}{\partial x_{\mathrm{j}}}[(\nu+\nu_{\mathrm{t}})(\frac{\partial\langle u_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} + \frac{\partial\langle u_{\mathrm{j}}\rangle}{\partial x_{\mathrm{i}}})] + \langle f_{\mathrm{i}}\rangle \quad (\mathrm{i},\mathrm{j}=1,2,3) \tag{2.1.11}$$

In turbulent flows eddy viscosity is not a property of the fluid, but depends upon the fluid density and distance from the river bed or ground surface. The principle of the eddy viscosity is only the basic framework that approximations have to be considered as turbulence models and to calculate $\nu_\mathrm{t}(x_\mathrm{i}, t)$.

Various turbulence models are available in CFD simulations. Based on the time-averaged RANS equations, four classical model hierarchies are used:

- Zero equation model: mixing length model (Prandtl (1925)).
- One equation model: Spalart-Almaras (Spalart and Allmaras (1994)).
- Two equation models: $k-\omega$ model (Saffman and Wilcox (1974)), $k-\epsilon$ model (Jones and Launder (1972a)) and Algebraic Stress Models (Rodi (1974))
- Seven equation model: Reynolds stress model (Launder et al. (1975)).

RANS based turbulence models are suitable for a large variety in engineering applications with good accuracy. However, none of these models is universal, different model shall be taken in different situations (Leschziner (2010)).

2.2. 2D flow models

A 3D model has a full momentum balance on small scale that might be too expensive for most practical problems. Due to the high requirement of computational cost, a few assumptions and simplifications can be made to reduce the simulation expense. For this purpose, 2D models are proposed and 3D flow problems become 2D flow problems in most applications such as atmospheric flows, river flows, coastal flows and so on. When the vertical velocity distributions are insignificant (*i.e.* vertical velocity is much smaller than the horizontal velocities) and the flow is dominated by a hydrostatic pressure distribution, the flow is considered as a shallow flow.

2.2.1. Shallow Water equations

The behaviour of a fluid in shallow water areas is governed by the Shallow Water equations (SWE), which has been obtained from the Navier-Stokes equations by depth-integrating under several simplifying assumptions. The SWE describe the properties and dynamics of the depth-averaged velocities $\bar{u}$ and the flow depth h. Derivation of the SWE has been achieved by two principal procedures:

- Transport equation for flow depth h is derived by integrating the mass conservation equation from the bottom z_b [m] to the free surface z_w [m] and specifying boundary conditions for a water column at the water surface and at the bottom.
- The momentum equations for flow velocities are depth-averaged in x-, y-direction and in z-direction the momentum transport is negligible compared to the horizontal ones.

A physical quantity ϕ can be split into a depth-averaged component $\bar{\phi}$ and a fluctuation part $\phi^{'}$, $\phi = \bar{\phi} + \phi^{'}$, where the depth-averaged quantity is defined as in equation (2.2.1).

$$\bar{\phi} = \frac{1}{h}\int_{z_\mathrm{b}}^{z_\mathrm{w}} \phi dz \tag{2.2.1}$$

A typical water column definition is shown in figure 2.1. The subscript w and b mean the bottom and the water surface respectively.

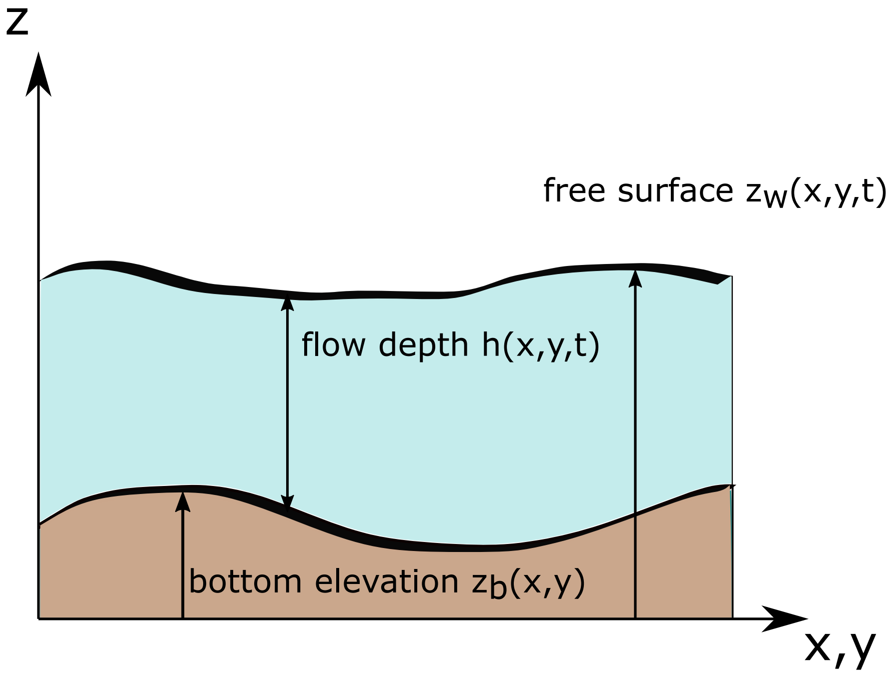


Figure 2.1: Typical water column

Kinematic boundary conditions at the bottom are defined in equations (2.2.2) and (2.2.3),

$$u(z_\mathrm{b}) = v(z_\mathrm{b}) = 0 \tag{2.2.2}$$

$$w(z_\mathrm{b}) + u(z_\mathrm{b})\frac{\partial z_\mathrm{b}}{\partial x} + v(z_\mathrm{b})\frac{\partial z_\mathrm{b}}{\partial y} = 0 \tag{2.2.3}$$

resulting equation (2.2.4).

$$w(z_\mathrm{b}) = 0 \tag{2.2.4}$$

where u, v, w $[\mathrm{m} \cdot \mathrm{s}^{-1}]$ are flow velocity components in $x-$, $y-$ and $z-$directions.

Kinematic boundary condition at the surface gives:

$$w(z_{\mathrm{w}}) = \frac{\partial h}{\partial t} + u(z_w)\frac{\partial z_w}{\partial x} + v(z_{\mathrm{w}})\frac{\partial z_{\mathrm{w}}}{\partial y} \tag{2.2.5}$$

Depth integration of mass conservation

As indicated in figure 2.1, the flow depth h is the distance from the bottom to the free surface ($h = z_{\mathrm{w}} - z_{\mathrm{b}}$). Integrate the continuity equation (2.1.6) from z_{b} to z_{w} (both z_{b} and z_{w} are time-dependent) resulting in:

$$\int_{z_{\mathrm{b}}}^{z} (\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z})dz = 0 \tag{2.2.6}$$

Use the Leibniz integral rule:

$$\frac{\partial}{\partial x}\int_{z_{\mathrm{b}}}^{z_{\mathrm{w}}} udz + \frac{\partial}{\partial y}\int_{z_{\mathrm{b}}}^{z_{\mathrm{w}}} vdz - (u(z_{\mathrm{w}})\frac{\partial z_{\mathrm{w}}}{\partial x} + u(z_{\mathrm{b}})\frac{\partial z_{\mathrm{b}}}{\partial x}) - (v(z_{\mathrm{w}})\frac{\partial z_w}{\partial y} + v(z_{\mathrm{b}})\frac{\partial z_{\mathrm{b}}}{\partial y}) + (w(z_{\mathrm{w}}) - w(z_{\mathrm{b}})) = 0 \tag{2.2.7}$$

Then apply the depth averaged definition into:

$$0 = \frac{\partial h\bar{u}}{\partial x} + \frac{\partial h\bar{v}}{\partial y} + u(z_{\mathrm{b}})\frac{\partial z_{\mathrm{b}}}{\partial x} - (v(z_{\mathrm{w}})\frac{\partial z_{\mathrm{w}}}{\partial y} + v(z_{\mathrm{b}})\frac{\partial z_{\mathrm{b}}}{\partial y}) + (w(z_{\mathrm{w}}) - w(z_{\mathrm{b}})) \tag{2.2.8}$$

The transport equation for the flow depth is derived by substituting equations (2.2.2), (2.2.3) and (2.2.5) into equation (2.2.8):

$$\frac{\partial h}{\partial t} + \frac{\partial h\bar{u}}{\partial x} + \frac{\partial h\bar{v}}{\partial y} = 0 \tag{2.2.9}$$

Depth integration of momentum conservation

Based on the primary assumption in the SWE, horizontal scales are much larger than vertical scales, the relationship between the horizontal length l and the flow depth h are as follows: $l \gg h$. We use U represent the order of magnitude of horizontal velocities u and v and W stand for the order of magnitude of the vertical velocity w. We have equations (2.2.10) and (2.2.11).

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \approx \frac{U}{l} \tag{2.2.10}$$

$$\frac{\partial w}{\partial z} \approx \frac{W}{h} \tag{2.2.11}$$

Equation (2.2.12) is then obtained.

$$W \approx U\frac{h}{l} \tag{2.2.12}$$

By a scaling analysis and neglecting the vertical acceleration, all terms in Navier-Stokes $z-$momentum equation except the pressure derivative and the gravity term are rather small. As a consequence, pressure gradient balances the gravitational acceleration denoted by g $[\mathrm{m}\cdot\mathrm{s}^{-2}]$.

$$\frac{\partial p}{\partial z} = -\rho g \tag{2.2.13}$$

Horizontal velocities u and v are independent of the vertical velocity w. Simplified $x-$ and $y-$momentum equations in (2.2.14) - (2.2.15) are derived.

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}) \tag{2.2.14}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}) \tag{2.2.15}$$

For the following derivation, we take x-momentum equation as an example.

Since the shallow water is considered having hydrostatic pressure distribution, the integral form of equation (2.2.13) gives us (with atmospheric pressure p_{atm} $[\mathrm{kg}\cdot\mathrm{m}^{-1}\cdot\mathrm{s}^{-2}]$):

$$p = g\int_{z_{\mathrm{b}}}^{z_{\mathrm{w}}} \rho dz + p_{\mathrm{atm}} \tag{2.2.16}$$

With constant density assumption, finally we have a new expression for pressure:

$$p = \rho g(z_{\mathrm{w}} - \mathrm{z}) + p_{\mathrm{atm}} \tag{2.2.17}$$

Pressure term of the SWE in momentum equation can be obtained by implying equation

(2.2.17) and integrating over depth:

$$\begin{aligned}-\frac{1}{\rho}\int_{z_\mathrm{b}}^{z_\mathrm{w}}\frac{\partial p}{\partial x} &= \int_{z_\mathrm{b}}^{z_\mathrm{w}}\frac{\partial(g(z_\mathrm{w}-z)+p_\mathrm{atm})}{\partial x}\\ &= -\frac{\partial}{\partial x}\int_{z_\mathrm{b}}^{z_\mathrm{w}} g(z_\mathrm{w}-z)dz + g(z_\mathrm{w}-z_\mathrm{w})\frac{\partial z_\mathrm{w}}{\partial x} - g(z_\mathrm{w}-z_\mathrm{b})\frac{\partial z_\mathrm{b}}{\partial x}\\ &= -\frac{g}{2}\frac{\partial h^2}{\partial x} - gh\frac{\partial z_\mathrm{b}}{\partial x}\end{aligned} \tag{2.2.18}$$

The stress tensor defined in equation (2.2.19) is introduced to express stress terms in 2D models with μ $[\mathrm{kg}\cdot\mathrm{m}^{-1}\cdot\mathrm{s}^{-1}]$ the dynamic viscosity.

$$\tau_\mathrm{ij} = \mu(\frac{\partial u_\mathrm{i}}{\partial u_\mathrm{j}} + \frac{\partial u_\mathrm{j}}{\partial u_\mathrm{i}}) \tag{2.2.19}$$

Depth integration of the stress term yields

$$\begin{aligned}\int_{z_\mathrm{b}}^{z_\mathrm{w}}[\nu(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2})]dz =& \frac{1}{\rho}\int_{z_\mathrm{b}}^{z_\mathrm{w}}(\frac{\partial \tau_\mathrm{xx}}{\partial x}+\frac{\partial \tau_\mathrm{yx}}{\partial y}+\frac{\partial \tau_\mathrm{zx}}{\partial z})dz\\ =& \frac{1}{\rho}\frac{\partial h\overline{\tau_\mathrm{xx}}}{\partial x}+\frac{1}{\rho}\frac{\partial h\overline{\tau_\mathrm{yx}}}{\partial y}+\frac{1}{\rho}\tau_\mathrm{zx}(z_\mathrm{w})-\frac{1}{\rho}\tau_\mathrm{zx}(z_\mathrm{b})\end{aligned} \tag{2.2.20}$$

where τ_xx and τ_yx $[\mathrm{kg}\cdot\mathrm{m}^{-1}\cdot\mathrm{s}^{-2}]$ are depth-averaged stresses that can be neglected. All stress terms at the free surface are also neglected. In addition, shear stress at the bottom except the wall shear stress $\tau_\mathrm{zx}(z_\mathrm{b})$ are either neglected due to their small contribution.

When integrating the LHS of the equation (2.2.14), we get,

$$\int_{z_\mathrm{b}}^{z_\mathrm{w}}(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y})dz = \frac{\partial h\bar{u}}{\partial t}+\frac{\partial h\bar{u}^2}{\partial x}+\frac{\partial h\bar{u}\bar{v}}{\partial y}+\frac{\partial h\overline{u'u'}}{\partial x}+\frac{\partial h\overline{u'v'}}{\partial y} \tag{2.2.21}$$

where $\overline{uu} = \bar{u}\bar{u} + \overline{u'u'}$ and $\overline{uv} = \bar{u}\bar{v} + \overline{u'v'}$.

Similar to Time-averaging technique for non-linear terms in equation (2.1.7), the last two terms in equation (2.2.21) represent a lateral momentum transfer due to rotating secondary flow are diffusive terms, which are not covered in the SWE.

Putting everything together, the final SWE in index form of the continuity equation (2.2.22)

$$\frac{\partial h}{\partial t}+\frac{\partial hu_\mathrm{i}}{\partial x_\mathrm{i}} = 0 \quad (\mathrm{i}=1,2) \tag{2.2.22}$$

and of momentum equation (2.2.23):

$$\frac{\partial hu_{\mathrm{i}}}{\partial t}+u_{\mathrm{j}}\frac{\partial hu_{\mathrm{i}}}{\partial x_{\mathrm{j}}}=-\frac{g}{2}\frac{\partial h^2}{\partial x_{\mathrm{i}}}-gh\frac{\partial z_{\mathrm{b}}}{\partial x_{\mathrm{i}}}+\frac{\partial h\overline{\tau_{\mathrm{ji}}}/\rho}{\partial x_{\mathrm{j}}}-\frac{\tau_{\mathrm{Bi}}}{\rho}\quad(\mathrm{i},\mathrm{j}=1,2) \tag{2.2.23}$$

with τ_{B} $[\mathrm{kg}\cdot\mathrm{m}^{-1}\cdot\mathrm{s}^{-2}]$ the bottom shear stress.

The first two terms on the RHS of the momentum equation (2.2.23) are derived from the pressure term in equation (2.1.8), the first one of which characterizes the change in water level and the second is lead by the bottom slope. The other terms on the RHS of equation (2.2.23) embody the stress that need to be modelled.

2.2.2. Turbulence models in 2D flows

Compared to 3D flows, the modelling of turbulence in shallow water flows has been considered less profusely according to Cea et al. (2007). 2D depth-averaged equations can be solved by applying a depth-averaged turbulence model. Several depth-averaged turbulence models derived from the RANS turbulence models have been offered for the SWE of different complexity. The effects of bottom friction is generally included. Two zero-equation turbulence models are the depth-averaged parabolic eddy viscosity model and the depth-averaged mixing length model. The eddy viscosity is computed from the mean horizontal shear and the bed effect in the depth-averaged mixing length model. The depth-averaged parabolic eddy viscosity model is thus formulated if no horizontal mean velocity is considered or if the turbulence is dominated by bottom friction (Dorfmann (2016)). A two-equation model, the depth-averaged $k-\epsilon$ model, first proposed by Rastogi and Rodi (1978) is the most commonly used SWE turbulence model. Such model is achieved by depth integrating the 3D standard $k-\epsilon$ model of Jones and Launder (1972b). Different versions of the two-equation depth-averaged turbulence models have been developed later.

In the present study, the depth-averaged parabolic eddy viscosity model is used and more details can be found in sections 3.5.1 and 3.5.2.

2.3. Type of PDEs

Most of mathematical physics are described by PDEs. A given non-linear PDE is typically only approachable to numerical solution. The type of solutions for PDE changes with different PDE because certain types of equations need appropriate boundary conditions. It is no doubt that determining a PDE system type is important in order to solve the problem properly.

2.3.1. Characteristic of PDE

The type of PDE implies the propagation of information. A PDE can be sorted by the order, the linearity and the homogeneity. The single first order PDE is always hyperbolic. A second order PDE is classified into three types as elliptic, parabolic and hyperbolic. Such classification helps in knowing the allowable initial and boundary conditions to a given problem which helps in the effective choice of numerical methods.

The classification of second order PDE is most easily described when the equations are formulated in matrix notation Garabedian (1965). The characteristic system, in which each equation involves partial differentiations in one direction only, can be gained if we discuss for a moment the question of classification of a more general quasi-linear system of PDE of first order in two independent variables.

Consider the quasilinear system:

$$\frac{\partial \mathbf{U}}{\partial x} + \mathbf{C}\frac{\partial \mathbf{U}}{\partial y} = \mathbf{F} \tag{2.3.1}$$

where $\mathbf{U}$ denotes a $m \times 1$ column vector, $\mathbf{C}$ is a $m \times m$ matrix and $\mathbf{F}$ is a $m \times 1$ column vector. Their arguments c_{ij} and f_{i} are functions of x and y.

The curves determined by $\frac{dy}{dx} = \lambda_{\mathrm{k}}$ $(\mathrm{k} = 1, ..., m)$ are characteristic curves, or simply called characteristics. The system (2.3.1) is said to be `hyperbolic` if c_{ij} has m real eigenvalues λ_{k}, defined as the roots of the characteristic equation $|\mathbf{C} - \lambda\mathbf{I}| = 0$, and if, in addition, $\mathbf{C}$ has a full set of m normalized independent orthogonal eigenvectors. If all eigenvalues are real and distinct, the system is of `strictly hyperbolic` type. If we have less than m λ_{k}, the system is `parabolic`, if we have complex eigenvalues (the characteristics are imaginary), the system is `elliptic`.

Hyperbolic PDE have two real characteristics, thus two initial conditions are required. Information propagates at a finite speed along the characteristics and the solution can be effected in a limited influence zone. Equations of this type are often presented in time-dependent marching problem with neglecting energy dissipation.

Parabolic PDE have only one real characteristics and one initial conditions are required. In addition, boundary conditions must be specified for all time $t > 0$. This problem is called an initial-boundary value problem. The preferred physical information propagation paths are lines or surfaces of constant time. Parabolic equations normally describe time-dependent marching problems with significant diffusion involving.

Elliptic PDE have no real characteristics and no preferred physical information propagation path, which means the information travels equally in all directions. Solution will be influenced by any disturbance at any point in the problem domain. As a consequence, one boundary condition is required at all points on the boundary. This is called a boundary-value problem. Elliptic equations describe steady state or equilibrium problems.

2.3.2. Type of Navier-Stokes equations

Many theory researches have been conducted through the type of Navier-Stokes equations among the versions of these equations, *e.g.* steady and unsteady, viscid and inviscid, compressible and incompressible.

Anderson et al. (1984) stated that the unsteady compressible Navier-Stokes equations are a mixed set of hyperbolic-parabolic equations and Navier-Stokes equations become a mixed set of hyperbolic-elliptic equations under steady conditions. For viscid flows, Navier-Stokes equations are elliptic under steady state and are parabolic under unsteady conditions. The classification of inviscid flows have been conducted by Anderson (1995) and Versteeg and Malalasekera (2007). The system of equations is always hyperbolic if unsteady flow is presented. For compressible flow, density of a given fluid particle changes with position. Type of equations for steady flows are therefore further classified regarding with the flow conditions. The classification of the system is elliptic the flow is subsonic and hyperbolic if the flow is supersonic. Steady incompressible inviscid flows are elliptic because the incompressible flow is a subcase of subsonic flow with Mach number is zero (Garg (1998)).

As concluded by Murman and Krupp (1971), the unsteady viscous incompressible Navier-Stokes equations is a parabolic equation and the steady viscous incompressible Navier-Stokes equations is an elliptic equation. However, elliptic equations are difficult to solve. In most science and engineering research we consider the Navier-Stokes equations is hyperbolic if convection terms dominated and is parabolic if diffusion terms are dominated.

2.3.3. Type of Shallow Water equations

The 2D SWE system has the same characteristics as compressible flows. Compressible Navier-Stokes equations may be considered as mixed hyperbolic, parabolic and elliptic equations depends on the viscosity and flow conditions. Since the SWE system applies for incompressible flows, flow conditions depend on the Froud number instead of the Mach number.

The inviscid fully non-linear 2D SWE system is of hyperbolic type. The time independent part of the SWE, *i.e.* steady state, is of hyperbolic type with supercritical flow; of parabolic with critical flow; and of elliptic with subcritical flow. A full analysis of PDE type on the SWE can be found in appendix A.

2.3.4. Shock wave travelling upstream from 3D to 2D problems

In this study, the 3D model is a viscid unsteady incompressible Navier-Stokes equations system, which has a parabolic PDE type. The 2D model has the same PDE type as the inviscid unsteady compressible flow, which is hyperbolic. Problems with the 2D-3D coupling are non-linear mixed hyperbolic-parabolic type. The problem may be either an initial or initial-boundary value problem or a combination thereof, depending on the particular situation. The flow condition changes will not change the boundary condition requirements for parabolic equations, which applies in the 3D model.

According to the theory of characteristics, we need to prescribe boundary conditions depending on the number of positive eigenvalues. As demonstrated in appendix A, the unsteady 2D SWE system has three eigenvalues with $\lambda_1 = u$, $\lambda_{2,3} = u \pm c$. When the flow is supercritical, all eigenvalues are positive, three boundary conditions are prescribed at the inlet of the 2D domain. When the flow is subcritical, one eigenvalue turns to be negative, therefore one boundary condition should be set at the outlet and two boundary conditions are given at the inlet.

This can be explained in modelling shock wave travelling upstream from 3D to 2D where boundary conditions change in the 2D model. To simulate this phenomenon properly, boundary condition at the outlet of the 2D domain is necessary.

2.4. Summary

Numerical methods based on the RANS equations and the SWE are both popular in fluid simulations. 3D models provide users with more accurate results as well as with more flow features which are highly time consuming in full scales, while 2D models are developed to compensate the high computational expense by reducing the complexity of solving models. The SWE come up with several limitations:

- On account of depth averaging, the SWE provide no information about the velocity distribution along the flow depth.
- The lack of velocity distribution and the ignored vertical velocity component therefore lead to the disability of the secondary flow motions representation.
- In addition, modelling of the river bed friction is necessary in the SWE model which means the chosen friction models influence the solutions.
- In cases where the prerequisites for the SWE are unsatisfied (no hydrostatic pressure distribution or a relevant small velocity component), the SWE give wrong results.

This chapter epitomizes the most widely used models for the computation of fluid flows, including governing equations, special treatment of turbulence models, the type of PDE systems and their limitations. There exist no universal model unless the problem is defined. How to weigh the advantages and disadvantages of a model is very case depending. Although 3D models can capture more flow details and are becoming more popular thanks to increasingly development in computational power, 3D models are more inclined to model the flow in hydraulic structures, where turbulent features should be calculated with a high accuracy. As we can see in this chapter, 2D models put less effort on turbulence model but give an overall good representation of shallow flow. When simulating a problem in a large scale, more precisely, a realistic river channel or a coastal region, 2D models have the priority over 3D models, because computational cost should be taken into account.

As an increasing requirement of urban flood simulations, where the large field and the local structures shall all be considered in one simulation, coupling 2D-3D models are emerged in order to benefit the accuracy from 3D models and the efficiency from 2D models.

3. Numerical Models in OpenFOAM

Numerical methods for computational fluid dynamics (CFD) are employed to solve mathematical models in a numerical manner. Today, various CFD codes have been developed and are on the market. An open source CFD framework, OpenFOAM, is selected for this study and is briefly introduced in section 3.1. Section 3.2 presents the discretization method that OpenFOAM uses. Temporal discretization will be short explained in section 3.3. Sections 3.4 to 3.6 describe solver implementations in OpenFOAM as well as some special techniques which are relevant to this work. A general coupling framework will also be included in section 3.6.

3.1. OpenFOAM framework

3.1.1. CFD program selection

CFD software packages fall into one of the following three categories:

- Commercial: Many business institutions sale the all-in-one offering CFD software as a product, where grid generators, flow solvers, and post-processors are integrated into a single CFD package. They are user friendly yet expensive. Two well-known commercial CFD packages in surface flow simulations are `ANSYS` Mechanical (2017) and `FLOW-3D` Santa Fe (2019).
- Open source: an open-source software is provided in accordance with a software license (*e.g.* the General Public License, the Berkeley Software Distribution or similar licenses) that allows users to use, study, change, and improve the software. Under the license structure, the software remains free of charge and retains a wide audience and developer base. The most widely used open-source CFD software package for general purpose is OpenFOAM ESI Group (2012).
- Customer: Besides existing software, users may write their own codes. For example, the in-house code `MGLET` (Manhart (2004) and Peller et al. (2006)) for large-scale space and time resolved simulations of turbulent flows.

We intend to set up multi-scale flow simulations for urban flooding scenarios where the 2D-3D coupling in both meshing and solving stages are necessary. To complete this objective, a great deal of research is indispensable. OpenFOAM is a solid CFD software that is highly competitive in terms of technology with commercial solutions currently available. With full access to source codes, developers can create tailored solutions that are suited to their needs. With its open source philosophy, this platform can fit easily into any development cycle. Although the largest drawback to working with it is the lack of a user interface, processes can be fully automated by writing a shell script in the Unix environment. Once a case has been set up for the first time, changes in geometry or flow conditions can be rapidly reset. Lastly, OpenFOAM gives robust results and it can be effectively scaled when applying to high computing performance. Therefore, OpenFOAM is selected as the CFD package platform in this research.

OpenFOAM was initially released by OpenCFD Ltd and it has been developed into three branches: (1) the official version of OpenFOAM development and maintenance that supported by OpenCFD Ltd (ESI Group (2012)), (2) the community edition of OpenFOAM called foam-extend which is provided by Wikki Ltd (Jasak (2004)), and (3) a fork of the OpenFOAM on behalf of OpenFOAM Foundation maintained by CFD Direct Ltd (Weller et al. (2015)). The standard 2D-3D coupling solver has been developed within foam-extend-3.1 environment, thus foam-extend-3.1 is the chosen OpenFOAM branch and denoted as OpenFOAM in this dissertation.

3.1.2. General introduction

Open source Field Operation and Manipulation (OpenFOAM) is an open-source object-oriented library for numerical simulations in continuum mechanics written in the C++ programming language which is especially popular in the fields of science and engineering. It uses the finite volume method (FVM) to solve systems of PDE. The fluid flow solvers are developed within a robust, implicit, pressure-velocity, iterative solution framework, although alternative techniques are applied to other continuum mechanics solvers. An entire numerical simulation generally consists of three steps: pre-processing, problem solving and post-processing. OpenFOAM framework listed below is able to realize all these procedures (Greenshilds and Ltd (2015)). Converters to / from other pre- and post-processors are available.

- Pre-processing: transfer the provided input into a suitable form that can be used by the solver, for instance mesh generation *e.g.* `blockMesh`, `mapField`.
- Standard solvers: solve a specific problem in continuum mechanics, like a two-phase flow solver *e.g.* `interFoam` (a standard multi-phase 3D solver detailed in section 3.4).
- User applications: new solvers and utilities can be created by its users, *e.g.* `shallowFoam` (a custom user-defined solver that will be presented in section 3.5).
- Post-processing: extract the desired information from the computed flow field (*e.g.* `sample`) and visualize the simulation results (*e.g.* `paraFoam`).

Every OpenFOAM simulation has been built with different inputs under a specific structure depending on the solver. Details of OpenFOAM simulation cases can be founded in appendix B.

3.2. Finite volume method implementation in OpenFOAM

Numerical algorithms, as employed in CFD, cannot represent continuous fields but only discrete field values at discrete points or control volumes. The solution of PDE can be very challenging, depending on many factors, *e.g.* the type of the equation, the number of independent variables as well as boundary and initial conditions. A common feature of all discretization methods is that a system of algebraic equations needs to be solved at the grid points or cells due to the discretization of PDEs (Sayma, 2009). This section focuses on the implementation of FVM, one of the most widely used discretization methods for PDEs, which is used in OpenFOAM.

The FVM, which was developed in the early 1970's, is a discretization technique for representing and evaluating PDEs in the form of algebraic equations (LeVeque (2002), Toro (2009) and Ferziger and Peric (1996)). The fundamental property of FVM is the integral conservation of the quantity, which makes it a preferred method in CFD. The core idea of this method is the approximation of the integral conservation law on each control volume. Furthermore, FVM can be easily formulated on unstructured meshes which fit the physical space better. Since the unknown variables are estimated at the cell center not at the cell faces, it is simple to achieve different boundary conditions in a non-invasive manner.

In general, FVM involves the following steps: (1) Decomposition of the problem domain into a sequence of control volumes. (2) Formulation of integral balance equations for each control volume. (3) Approximation of integrals by numerical integration. (4) Approximation of function values and derivatives by interpolation with nodal values. (5) Assembling and solution of discrete algebraic system.

3.2.1. Domain discretization

The finite volume cell-centered discretization of the computational domain results in a group of control volumes on which the governing equations are eventually solved. These non-overlapping control volumes completely fill the solution domain. Values of variables are assessed at the centroids of the control volume. This allows the optimal flexibility in describing unstructured grids in a general polyhedron shape. Discrete grids in OpenFOAM can be arranged in space as structured grids, unstructured grids and block structured grids.

For simplicity, we take the 2D Cartesian mesh illustrated in in figure 3.1 as an example to demonstrate the principal procedures of FVM. Data of variables are stored at the cell center (marked as a point) and values at faces (marked as a cross) are obtained from the interpolation scheme. Cell faces are classified into internal faces (dashed lines) and boundary faces (solid lines). Internal faces are located between two cells, while boundary faces coincide with the boundaries of the domain.

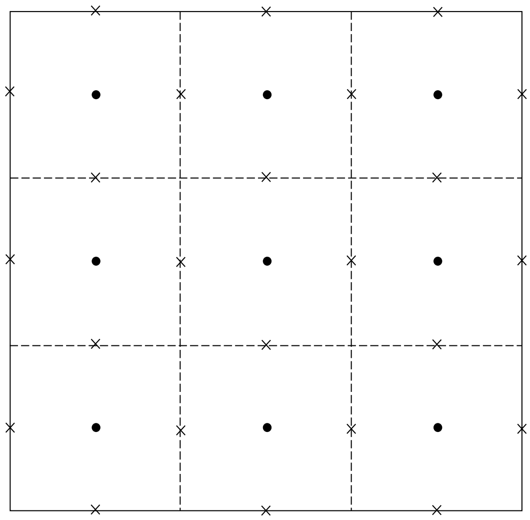


Figure 3.1: Numerical grids with cell center and face center in a 2D Cartesian mesh

3.2.2. Spatial discretization of equations

The conservation equation for a general variable ϕ can be expressed in vector notation form as:

$$\frac{\partial \rho\phi}{\partial t} + \nabla \cdot (\rho \mathbf{U} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S(\phi) \tag{3.2.1}$$

with the vector form velocity $\mathbf{U}$, the diffusion coefficient Γ and the source term S.

Such standard transport equation (3.2.1) is first integrated over the cell domain and spatial derivatives (*i.e.* convection and diffusion terms) are then converted by surface integral over the bounding surface using the Gauss divergence theorem. The resulting equation (3.2.2) is given below:

$$\int_{V_\mathrm{c}} \frac{\partial \rho\phi}{\partial t} dV + \int_{\partial V_\mathrm{c}} (\rho \mathbf{U} \phi) \cdot \mathbf{n} d\mathbf{S} = \int_{\partial V_\mathrm{c}} (\Gamma \nabla \phi) \cdot \mathbf{n} d\mathbf{S} + \int_{V_\mathrm{c}} S(\phi) dV \tag{3.2.2}$$

with V_c the volume of the cell, ∂V_c the closed surface encircling the control volume V_c, $\mathbf{n}$ the normal vector of this surface pointing outwards and $d\mathbf{S}$ the outward pointing differential surface area vector.

Equation (3.2.2) consists volume integrals (time derivative and source terms) and surface integrals (convection and diffusion terms) which can be further approximated. We employ exemplary the midpoint rule of approximation on the temporal derivative and spatial derivatives. Figure 3.2 exhibits the surface integration of fluxes from the target cell center P to its neighbour cell center N. $\mathbf{S}_\mathrm{f}$ is the face area normal vector pointing outwards with the magnitude is equal to the face area. d_PN is the distance between points P and

N.

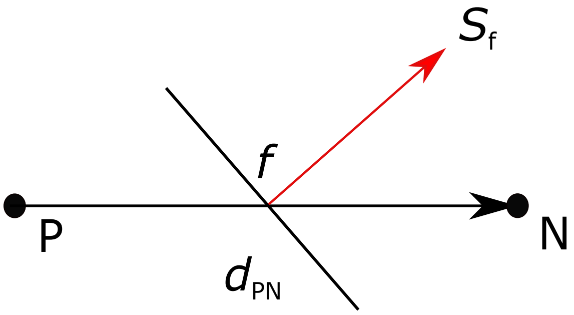


Figure 3.2: Surface integration of fluxes using midpoint rule

Time derivative term

The volume integral of the time-dependent term is therefore become a linear variation of the integrand (equation (3.2.3)),

$$\int_{V_\mathrm{c}} \frac{\partial \rho \phi}{\partial t} dV = \frac{\partial \rho \phi_\mathrm{P} V_\mathrm{c}}{\partial t} \tag{3.2.3}$$

where ϕ_P is the cell center value at P.

Convection term

Surface integral terms are evaluated as fluxes at the surfaces, which ensures the conservation of fluxes of the control volume. Since each cell is bounded by a series of flat faces, these terms can be transformed into a sum of integrals over all faces using the centroid value of face surfaces. The convection term approximation is accessed in equation (3.2.4),

$$\int_{\partial V_\mathrm{c}} (\rho \mathbf{U} \phi) \cdot \mathbf{n} d\mathbf{S} = \sum_\mathrm{f} (\rho \mathbf{U})_\mathrm{f} \phi_\mathrm{f} \mathbf{S}_\mathrm{f} \tag{3.2.4}$$

where ϕ_f is the face center value at f and $(\rho \mathbf{U})_\mathrm{f} \mathbf{S}_\mathrm{f}$ is the face flux through the face f.

Diffusion term

In a similar way, we obtain the surface integral form of the diffusion term in equation (3.2.5)

$$\int_{\partial V_{\mathrm{c}}} (\Gamma \nabla \phi) \cdot \mathbf{n} d\mathbf{S} = \sum_{\mathrm{f}} \Gamma_{\mathrm{f}} (\nabla \phi)_{\mathrm{f}} \mathbf{S}_{\mathrm{f}} \tag{3.2.5}$$

where $(\nabla \phi)_{\mathrm{f}}$ is the face gradient.

Source term

Several source term discretization methods can be found in Patankar (1981). One point Gauss integration will be presented here. The source term should first be linearised as:

$$S(\phi) = S_{\mathrm{C}} + \phi S_{\mathrm{L}} \tag{3.2.6}$$

where S_{C} is the constant part and S_{L} is the linear part of the source term. Finally the volume integral of the source term is computed in equation (3.2.7).

$$\int_{V_{\mathrm{c}}} S(\phi) dV = S_{\mathrm{C}} V_{\mathrm{c}} + S_{\mathrm{L}} V_{\mathrm{c}} \phi_{\mathrm{P}} \tag{3.2.7}$$

This approximation is second order accurate and is applicable in two and three dimensions if ϕ_{P} is in the center of the cell.

3.2.3. Interpolation schemes

The solution is available only at computational nodes, *i.e.* center of the control volume. Face values and face gradients are embedded in the equation discretization. Interpolation schemes are therefore demanded to transform cell-center quantities to face centers. OpenFOAM offers a wide range of options for interpolation schemes as well as gradient schemes (Greenshilds and Ltd (2015)). The accuracy of FVM disretizations are mesh and discretization schemes dependent. Due to the coupling mesh feature in this work, orthogonal meshes are required. For bounded transport quantities, bounded differencing schemes are also indispensable.

For convection terms, a promising approach that guarantees boundedness of the solution is the upwind differencing scheme. The idea is taking information from upstream. The transportation property of the term specifies which point should be taken into account during the interpolation (equation (3.2.8)). Therefore, the convective transport of ϕ only happens downstream. It gives a bounded solution by reducing the accuracy. The leading error resembles a diffusive flux using Taylor series expansion. Consequently, no oscillations appear in the result, but the solution is smeared on coarse meshes.

$$\phi_{\mathrm{f}} = \max(F, 0)\phi_{\mathrm{P}} + \min(F, 0)\phi_{\mathrm{N}} \tag{3.2.8}$$

where $F = (\rho\mathbf{U})_{\mathrm{f}}\mathbf{S}_{\mathrm{f}}$ and ϕ_{N} is the cell center value at N. When $F > 0$, it is a backward difference scheme. Otherwise, it is a forward difference scheme when $F < 0$.

Discrete form of the diffusion term preserves boundedness on orthogonal meshes, which implies vectors d_{PN} and $\mathbf{S}_{\mathrm{f}}$ are aligned to each other. The face normal gradient of ϕ can be calculated by equation (3.2.9).

$$(\nabla\phi)_{\mathrm{f}}\mathbf{S}_{\mathrm{f}} = |\mathbf{S}_{\mathrm{f}}|\frac{\phi_{\mathrm{N}} - \phi_{\mathrm{P}}}{|d_{\mathrm{PN}}|} \tag{3.2.9}$$

3.2.4. Treatment of boundary conditions

Numerical boundary conditions provides numerical treatment of conditions in the model. These include three general boundary types in OpenFOAM: Dirichlet boundary condition, Neumann boundary condition and Robin boundary condition. Implementation of these boundary types are explained as follows.

Dirichlet boundary condition

A Dirichlet boundary condition specifies a value explicitly on the boundary patch by using `fixedValue` in OpenFOAM as a dictionary entry.

$$\phi_{\mathrm{f}} = \phi^{\mathrm{ref}} \tag{3.2.10}$$

where ϕ_{f} denotes the face value and ϕ^{ref} indicates the reference value.

Neumann boundary condition

A value on the boundary can be extrapolated by specifying the respective normal gradient via a Neumann boundary condition.

$$\phi_{\mathrm{f}} = \phi_c + \Delta\nabla\phi^{\mathrm{ref}} \tag{3.2.11}$$

where ϕ_c gives the cell value, Δ is the face-to-cell distance and $\nabla\phi^{\mathrm{ref}}$ is the reference gradient prescribed by either `zeroGradient` or `fixedGradient` in OpenFOAM.

Robin boundary condition

Robin type boundary condition is a linear blend of Dirichlet boundary condition and Neumann boundary condition by defining a weighted value fraction, thus a `mixed` entry type is provided for OpenFOAM users. Face values are evaluated according to a function of reference value and reference gradient.

$$\phi_{\mathrm{f}} = \omega\phi^{\mathrm{ref}} + \phi^{c} + (1-\omega)\Delta\nabla\phi^{\mathrm{ref}} \tag{3.2.12}$$

where ω is the value fraction between 0 and 1.

Besides numerical boundary conditions, physical boundary describe physical meanings in an algebraic equation. For free surface cases, *e.g.* flooding events, the flow discharge is always prescribed at the inlet boundary patch and the depiction of the flow depth at the inlet or outlet boundary depends on the flow conditions.

3.3. Temporal discretization

Equation discretization contains two procedures: spatial and temporal discretization. When the problem is time dependent, we must integrate the PDE in time as well. Assuming that the control volumes do not change in time, temporal discretization of equation (3.2.2) over a time step Δt can be represented in equation (3.3.1),

$$\frac{(\rho\phi)^{\mathrm{n+1}} - (\rho\phi)^{\mathrm{n}}}{\Delta t}V_{\mathrm{c}} = \int_{\mathrm{t}}^{\mathrm{t+\Delta t}} [L(\phi)]dt \tag{3.3.1}$$

where $L(\phi)$ denotes discrete terms containing full discretize convection, diffusion and source term from (equation (3.2.4), (3.2.5) and (3.2.7)).

$$L(\phi) = -\sum_{\mathrm{f}}(\rho\mathbf{U})_{\mathrm{f}}\phi_{\mathrm{f}}S_{\mathrm{f}} + \sum_{\mathrm{f}}\Gamma_{\mathrm{f}}(\nabla\phi)_{\mathrm{f}}S_{\mathrm{f}} + S_{\mathrm{C}}V_{\mathrm{c}} + S_{\mathrm{L}}V_{\mathrm{c}}\phi_{\mathrm{P}} \tag{3.3.2}$$

The RHS of equation (3.3.1) has to be computed using the time scheme, which defines how a property is integrated as a function of time. The choice of time scheme is highly dependent on the type of equations which you are solving. Various time schemes have been implemented in OpenFOAM which can be categorized into explicit and implicit methods. Table 3.1 indicates the advantages and drawbacks of explicit and implicit methods.

The most well-known method is the Euler method, a first-order numerical procedure for solving ordinary differential equations with a given initial value. In our work, different time schemes are employed in the 2D and the 3 solvers, details will be discussed in the section of each solver implementation.

	Explicit methods	Implicit methods
time step	small	a wide range
cost per time step	low	high
stability	conditionally	unconditionally

Table 3.1: Comparison of explicit and implicit time schemes

3.4. Reynolds-Averaged Navier Stokes equations solver - `interFoam`

The 3D solver `interFoam` is a standard OpenFOAM multiphase solver using the Volume of Fluid method for capturing the interface of incompressible fluids.

3.4.1. Governing equations

Let us apply the turbulence model equation (2.1.10) into the momentum equation (2.1.9), we get:

$$\frac{\partial \bar{u}_{\mathrm{i}}}{\partial t} + \bar{u_j}\frac{\partial \bar{u}_{\mathrm{i}}}{\partial x_{\mathrm{j}}} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_{\mathrm{i}}} + \frac{\partial}{\partial x_j}[(\nu + \nu_{\mathrm{t}})\frac{\partial \bar{u}_i}{\partial x_{\mathrm{j}}} + \frac{\partial \bar{u}_{\mathrm{j}}}{\partial x_{\mathrm{i}}}] + \bar{f}_{\mathrm{i}} \tag{3.4.1}$$

where ν_{t} needs to be modelled in the second term on the RHS.

For each phase the normal component of the pressure gradient at a stationary inclined wall is different under hydrostatic conditions. In OpenFOAM, a modified pressure variable p_{d} replaces the total pressure p to simplify the boundary condition description (Rusche (2002)):

$$p_{\mathrm{d}} = p - \rho\mathbf{g}\cdot\mathbf{x} \tag{3.4.2}$$

where $\mathbf{x}$ is the position vector.

Substituting the equation (3.4.2) and getting rid of the constant fluid density, a vector notation of the RANS equations can be expressed:

$$\nabla\cdot\mathbf{U} = 0 \tag{3.4.3}$$

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla\cdot(\mathbf{UU}) = -\frac{1}{\rho}\nabla p_{\mathrm{d}} - \mathbf{g}\cdot\mathbf{x} + (\nu + \nu_{\mathrm{t}})\Delta\mathbf{U} + \mathbf{F} \tag{3.4.4}$$

with $\mathbf{U}$ is the notation form of fluid velocity and $\mathbf{F}$ is the notation form of the body force, more specifically the gravity. The velocity vector here acts as a shared velocity of the two fluids which are water and air in most cases, and the interface is determined by solving the transport equation of volume fraction indicator α_1 introduced in the following.

3.4.2. Volume of fluid method

In fluid dynamics, Lagrangian and Eulerian coordinates have been employed as the basis for numerical solution algorithms. The selection of which coordinate system to use relies on the characteristics of the problem to be solved. OpenFOAM uses Eulerian formulations for 3D multiphase flow problems, in particular, problems where the discontinuities exist in one or more variables that Lagrangian methods cannot be applied (Hirt and Nichols (1981)). As mentioned in Rusche (2002), an indicator function, *e.g.* volume fraction, level set or phase field, is used to represent the interface. The Volume of Fluid (VOF) method is a simple, but powerful free-surface modelling technique. Based on the concept of a fractional volume of fluid, it describes the shape and position of the interface. In OpenFOAM, the VOF model was implemented by Rusche (2002) within the `interFoam` solver.

When defining the fluid state, it is customary to use only one value for each variable in each cell of the mesh. The VOF method requires only one storage word for each mesh cell, which is consistent with the storage requirements for all other dependent variables. An indicator function of α_1 (equation (3.4.5)) indicates the fractional volume of the cell occupied by fluid whether a cell is full of fluid, the cell contained no fluid or some state in between. The phase fraction α_1 values within the range $0 < \alpha_1 < 1$ must then contain a free surface. Accordingly, gradients of the phase fraction are encountered only in the region of the interface.

$$\alpha_1(x, y, z, t) = \begin{cases} 0 & \text{air} \\ 1 & \text{water} \\ 0 < \alpha_1 < 1 & \text{both} \end{cases} \tag{3.4.5}$$

The transport equation for the indicator function is solved simultaneously with the continuity and momentum equations.

$$\frac{\partial \alpha_1}{\partial t} + \nabla \cdot \alpha_1 u = 0 \tag{3.4.6}$$

The physical properties (*i.e.* density and viscosity) of which fluids phase are also calculated as weighted mixture based on the distribution of the liquid volume fraction:

$$\rho = \rho_{\text{water}} \alpha_1 + \rho_{\text{air}} (1 - \alpha_1) \tag{3.4.7}$$

$$\mu = \mu_{\text{water}}\alpha_1 + \mu_{\text{air}}(1 - \alpha_1) \tag{3.4.8}$$

where ρ and μ are density and dynamic viscosity respectively. Subscripts air and water indicates the properties of the fluid as air (gas) and water (fluid) respectively.

According to Berberović et al. (2009), with respect to the conservation of the phase fraction, problems may occur when the ratio of two fluids density is high. Small errors in volume fraction α_1 may lead to significant errors in calculations of physical properties such as surface curvature and the corresponding pressure gradient. Moreover, the VOF method is highly grid resolution sensitive. A numerical simulation with large grid sizes, the formation of small bubbles or droplets are smaller than the minimum grid size will be ignored. Much research has been conducted to overcome these difficulties, the most commonly way was formulated by Weller (2002), introducing the artificial compression term as an extra term in the equation (3.4.6).

$$\frac{\partial \alpha_1}{\partial t} + \nabla \cdot \alpha_1 u + \nabla \cdot u_{\text{r}}\alpha_1(1 - \alpha_1) = 0 \tag{3.4.9}$$

The third term on the LHS of equation (3.4.9) is then the artificial compression term with u_{r} the relative velocity between the two fluids (air and water). u is the average velocity of the two fluids calculated by:

$$u = \alpha_1 u_{\text{water}} + (1 - \alpha_1)u_{\text{air}} \tag{3.4.10}$$

The multi-dimensional limiter for explicit solution (MULES) scheme has been used for solving equation (3.4.9) as a very effective method of guaranteeing boundedness of α_1.

3.4.3. Pressure-velocity coupling

Because the pressure field is unknown as well as the three velocity components, there are four unknowns in only three equations. Furthermore, due to the quasi-linearity and the interdependence between the velocity $\mathbf{U}$ and the pressure p, the momentum equations can hardly be solved. A special numerical technique is required for solving this inter-equation coupling which is called the pressure-velocity coupling.

Fortunately for incompressible flow, the pressure field does not affect the flow continuity, the mass conservation is thus used to derive an additional condition for the pressure field. The idea is to make use of the continuity equation to get rid of terms and end up with the pressure Poisson equation.

Calculation of the pressure gradient is necessary. One way to get the pressure gradient of a control volume is interpolation from the face values, which will lead to checker-boarding problem on a uniform grid. In order to avoid this unphysical pressure field appearing in any situation, OpenFOAM uses Rhie-Chow correction (Ferziger and Peric (1996)).

Algorithms of dealing with the pressure-velocity coupling can be either simultaneously or in sequence. The former one also called the direct-coupled solver, is resource-demanding. It solves the equation system simultaneously over the whole domain which is more applicable under the small number of computational points simulations and is unaffordable for large cases. For speeding up the simulations, the later one is commonly put into practice called the segregated approach.

In the segregated approach, we keep all other variables constant for solving each unknown quantity. The solving equations will be done sequentially which means the velocity field will be solved first and then is the pressure field. In addition, the solution of the procedures above should satisfy the mass conversation and the given boundary conditions. The standard solution algorithms for pressure-velocity coupling in OpenFOAM are listed:

- SIMPLE: Semi-Implicit Method of Pressure Linked Equations that are used for steady state cases (Patankar (1981)).
- PISO: Pressure Implicit of Split Operations that more preferred in transient situations with a limited time step (Issa (1986)).
- PIMPLE: a PISO and SIMPLE combination that allows user to apply bigger time-step. It also supports partial convergence of intermediate iterations.

Realization of PISO algorithm in OpenFOAM

`interFoam` solver is capable to solve the general free surface problems including transient calculation thus the PISO procedure is used for the pressure-velocity coupling. PISO was developed originally for non-iterative computation of unsteady flow.
It involve one predictor step for $\mathbf{U}, p$ and two corrector steps, which can be also defined as a SIMPLE plus a further corrector step.

Figure 3.3 indicates the PISO algorithm splitting the operators into an implicit predictor and multiple explicit corrector steps. In every time step, it involves one predictor for $\mathbf{U}$ and two correctors for $\mathbf{U}$, p. The velocity field is corrected and updated explicitly. The point of the corrector step is to make the corrected velocity field divergence free to satisfy the continuity equation. PISO algorithm is robust and effective that within a very few corrector steps a desired accuracy is obtained.

3.4.4. Time integration

Systems of PDE in the 3D solver are treated in the segregated way due to the inter-equation coupling. For instance, the VOF solver MULES solving the transport equation of α_1 with its very strict limit on time-step. And both implicit and explicit manners have been presented in the PISO loop. Therefore, an explicit time integration scheme is actually employed in the 3D solver .

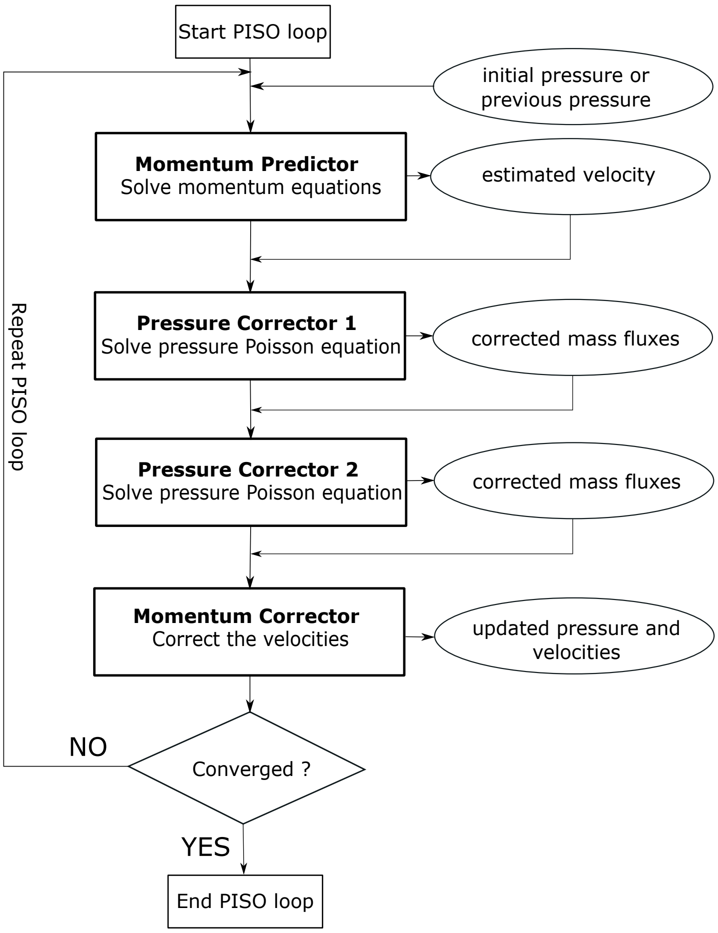


Figure 3.3: PISO loop in `interFoam`

As an explicit method, the estimation of a new time value is based on the rate of change at the current time value. The explicit Euler time scheme of equation (3.3.1) can be reformulated as:

$$(\rho\phi)^{\mathrm{n}+1} = (\rho\phi)^{\mathrm{n}} + \frac{\Delta t}{V_{\mathrm{c}}} L(\phi)^{\mathrm{n}} \tag{3.4.11}$$

The consequence of this method is that all terms on the RHS of equation (3.4.11) depend only on current values. The new time value of ϕ can be calculated directly. With respect to the convergence and stability of numerical schemes, the Courant number was addressed. Equation (3.4.12) defines the Courant number, which is the ratio between the propagation speed u and the gird travelling speed $\frac{\Delta x}{\Delta t}$. It measures how much information traverses a computational grid cell in a given time-step.

$$C_{\mathrm{o}} = \frac{\Delta t}{V_{\mathrm{c}}} \mathbf{U} \tag{3.4.12}$$

The Courant-Friedrichs-Lewy (CFL) condition name after Courant et al. (1928) is the maximum allowable Courant number that a time-integrator can use. In dealing with the explicit Euler method, time step lengths have to be fitted such that the CFL number should be smaller than one regarding to equation (3.4.12), otherwise the simulation diverges and the model will blow up.

3.4.5. `interFoam` solution procedure

After the computational domain is created, `interFoam` is applied to solve the problem numerically. Simulation set up in OpenFOAM for `interFoam` solver is detailed in appendix B. A flow chart of the solution procedure is displayed in figure 3.4. It starts with initial conditions, combing with the required CFL number a new time step is calculated. Then the transport equation for the indicator function $\alpha 1$ is solved. The PISO loop comes after getting the pressure p_{d} and the velocity U. Turbulent model is computed in the next step. The simulation will stop only when the final time is reached. The sub-parts enclosed by a dashed-line frame is briefed as 3D solver part, named `Solve 3D regions`, in the coupled solver.

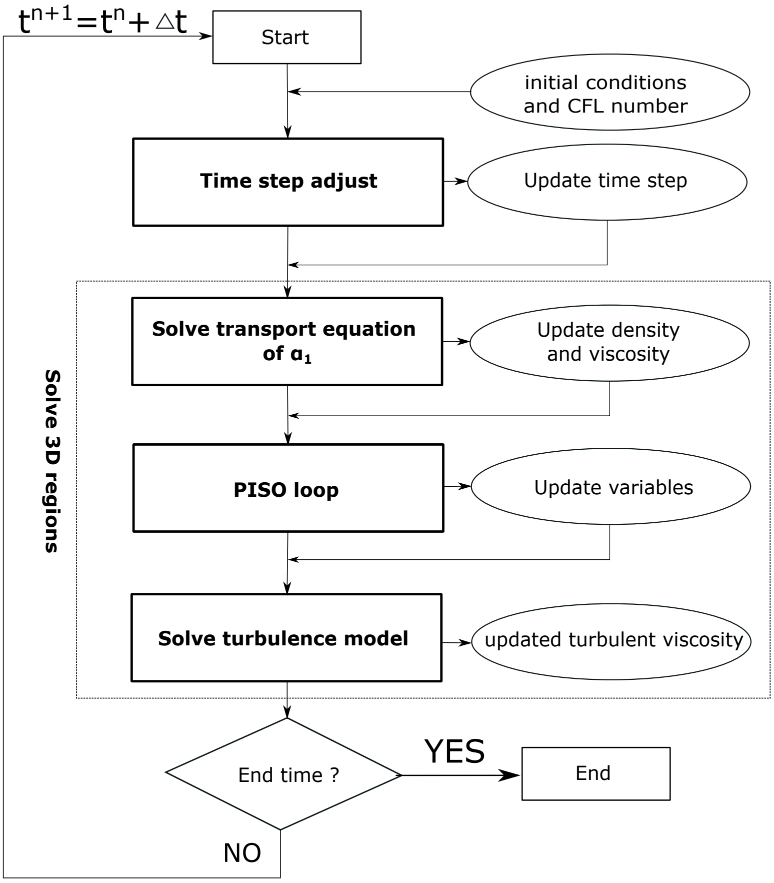

$t^{n+1}=t^n+\triangle t$
Start
initial conditions and CFL number
Time step adjust
Update time step
Solve transport equation of α_1
Update density and viscosity
PISO loop
Update variables
Solve turbulence model
updated turbulent viscosity
Solve 3D regions
End time ?
YES
End
NO

Figure 3.4: Solution procedure in `interFoam`

3.5. Shallow Water equations solver - `shallowFoam`

SWE are widely used in large-scale open channel simulations and the results are proved to be close to reality. Chair of Hydromechanics at Technical University of Munich developed the 2D solver `shallowFoam` under OpenFOAM environment Mintgen (2017). It solves the depth-averaged 2D shallow water equations which consist of the continuity equation (2.2.22) and the momentum equations (2.2.23). Bottom friction and eddy viscosity have been implemented to represent bottom shear stress.

3.5.1. Bottom friction modelling

The bottom friction is modelled under the assumption of horizontally uniform and steady flow. Therefore, equation (2.2.23) becomes:

$$\frac{\tau_{Bx_\mathrm{i}}}{\rho} = gh\frac{\partial Z_\mathrm{bi}}{\partial x_\mathrm{i}} \quad (i = 1, 2) \tag{3.5.1}$$

Manning's empirical formula relates the cross section average velocity u ($u = \sqrt{u_\mathrm{i}^2 + u_\mathrm{j}^2}$) [$\mathrm{m} \cdot \mathrm{s}^{-1}$] of open channel flow with the hydraulic radius R_h, the empirical roughness coefficient n and the slope of the energy head line I. The Manning formula gives as (Manning (1981)):

$$u = \frac{1}{n}R_\mathrm{h}^{2/3}I^{1/2} \tag{3.5.2}$$

When the flow depth keeps constant in a rectangular open channel shallow flow, the slope of the energy head line equals to the slope of the channel $I = -\frac{\partial Z_\mathrm{bi}}{\partial x}$ and the hydraulic radius can be simplified to the flow depth $R_\mathrm{h} \approx h$.

$$u_\mathrm{i} = -\frac{1}{n}h^{2/3}(\frac{\partial Z_\mathrm{b}}{\partial x_\mathrm{i}})^{1/2} \quad (i = 1, 2) \tag{3.5.3}$$

The expression for calculating the bottom stresses is obtained by substituting the slope term in equation (3.5.3) into the equation (3.5.1):

$$\frac{\tau_{Bx_\mathrm{i}}}{\rho} = u_\mathrm{i}^2 g\frac{n^2}{h^{1/3}} \quad (i = 1, 2) \tag{3.5.4}$$

The Strickler coefficient k_st [$\mathrm{m}^{1/3} \cdot \mathrm{s}^{-1}$] is introduced by definition $k_\mathrm{st} = \frac{1}{n}$ and is used in `shallowFoam`. The coefficient k_st varies from 20 (rough stone and rough surface) to 80 (smooth concrete and cast iron). Since the direction of the bottom stresses are against the flow direction, the velocity vector not only contains the magnitude value but also with the direction information. The bottom stresses modelling by means of Strickler-equation is therefore:

$$\frac{\tau_{Bx_\mathrm{i}}}{\rho} = u_\mathrm{i} \left| u \right| g \frac{n^2}{h^{1/3}} \quad (i = 1, 2) \tag{3.5.5}$$

3.5.2. Eddy viscosity model

Turbulence feature is captured by an eddy viscosity model. The mean shear stress has both viscous (first term on the RHS) and turbulent (second term on the RHS) parts defining in equation (3.5.6).

$$\frac{\tau_\mathrm{ij}}{\rho} = \nu \frac{\partial \langle u_\mathrm{i} \rangle}{\partial x_\mathrm{j}} - \langle u_\mathrm{i} u_\mathrm{j} \rangle \tag{3.5.6}$$

Under the eddy viscosity hypothesis the Reynolds stresses are assumed to be proportional to the mean velocity gradients in a manner analogous to viscous stress.

$$-\langle u_\mathrm{i} u_\mathrm{j} \rangle = \nu_\mathrm{t} \frac{\partial \langle u_\mathrm{i} \rangle}{\partial x_\mathrm{j}} \tag{3.5.7}$$

The kinematic eddy viscosity ν_t is a hypothetical property of the flow that must be modelled.

The mean shear stress now becomes:

$$\frac{\tau_\mathrm{ij}}{\rho} = (\nu + \nu_\mathrm{t}) \frac{\partial \langle u_\mathrm{i} \rangle}{\partial x_\mathrm{j}} \tag{3.5.8}$$

The friction velocity u^* is introduced to describe shear-related motion in moving fluids. It characterizes the shear at the boundary and defined on the basis of the wall shear stress. In turbulent flows, a scaling parameter for velocity fluctuating component is often represented by u^*.

$$u^* = \sqrt{\frac{\tau_\mathrm{Bx_i}}{\rho}} \tag{3.5.9}$$

In most of shallow water flows Re is very high thus the viscous term in equation (3.5.6) is negligible compared to turbulent stress because of $\nu_\mathrm{t} \gg \nu$. ν_t varies in position. The turbulent boundary layer over roughness elements within a high Re is classified into three sub-layers that the eddy sizes are scaled with different parameters:

- Roughness sublayer: Eddy size $\sim$ roughness size r
- Logarithmic region: Eddy size $\sim$ distance from the boundary z

- Outer region or a full-turbulent layer: Eddy size ∼ flow depth h

Above the sub-layer, the mean velocity can be correctly approximated by the logarithmic profile H.Tennekes and Lumley (1972). The profile shape depends both on the bed stress (through u^*) and also on the bed texture. The resulting parabolic eddy viscosity formula is implemented in `shallowFoam`:

$$\nu_\mathrm{t} = C_\nu u^* h \tag{3.5.10}$$

where the viscosity coefficient C_ν equals 0.7.

3.5.3. Time integration

`shallowFoam` uses the implicit Euler time scheme, which is the default time scheme in OpenFOAM. The new time values of variable are taken instead of the current time values compared to the explicit method. Equation (3.3.1) then becomes:

$$(\rho\phi)^{\mathrm{n+1}} = (\rho\phi)^{\mathrm{n}} + \frac{\Delta t}{V_\mathrm{c}} L(\phi)^{\mathrm{n+1}} \tag{3.5.11}$$

This method leads to the necessity of solving the system of an algebraic equation for the unknown, which can be very expensive because not only a few more iterations are required to converge the nonlinear problem but also at each iteration the linear matrix problem needs to be solved. In this way, the implicit Euler method is unconditionally stable even with large time-step. However, due to the order of accuracy in time, large time-step gives rise to low accuracy.

3.5.4. `shallowFoam` solution procedure

Similar to the 3D solver, the solution procedure is outlined in figure 3.5 where the dashed line frame surrounds the 2D solver part in the coupled solver. Although an implicit time scheme is applied, a CFL number can be specified as well to calculate the corresponding time step. Comparing to the explicit scheme, a larger CFL can be prescribed in order to set a larger time-step. Simulation set up for `shallowFoam` solver can be also found in appendix B.

3.6. Hybrid solver - `shallowInterFoamOL`

The coupled solver `shallowInterFoamOL` has been implemented in OpenFOAM by merging the 2D Shallow Water solver `shallowFoam` with the 3D Navier-Stokes solver `interFoam`. Coupling between different solvers can be achieved by means of coupling environment.

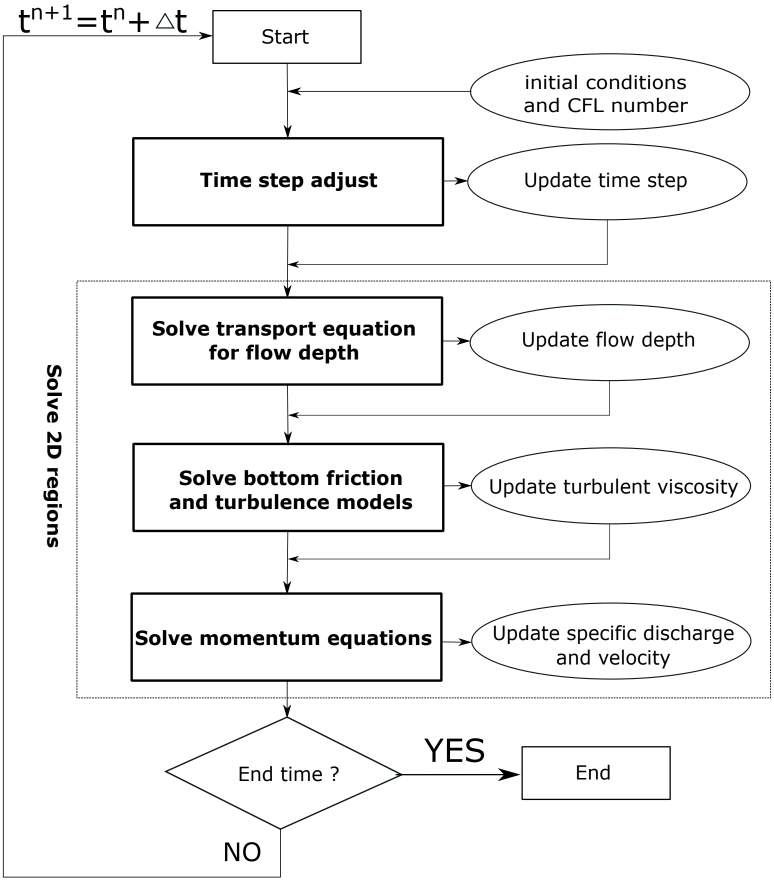

$t^{n+1}=t^n+\triangle t$
Start
initial conditions and CFL number
Time step adjust
Update time step
Solve 2D regions
Solve transport equation for flow depth
Update flow depth
Solve bottom friction and turbulence models
Update turbulent viscosity
Solve momentum equations
Update specific discharge and velocity
End time ?
YES
End
NO

Figure 3.5: Solution procedure in `shallowFoam`

3.6.1. General coupling environments

Many coupling structures are available nowadays which offer a wide usage of applications to users. Users take advantage of these software to couple the desired solvers for some specific problems. Some of them will be exemplified in the following.

`preCICE` coupling framework

Under the development corporation of the Technical University of Munich and the University of Stuttgart, an open-source plug-and-play coupling library `preCICE` framework is exhibited in figure 3.6. Such coupling framework `preCICE` which stands for Precise Code Interaction Coupling Environment is available to couple the multi-physics simulations such as fluid-structure interaction problems and conjugate heat transfer simulations that can be applied by different existing solvers (*e.g.* OpenFOAM, SU2, or CalculiX) (Bungartz et al. (2016)). With the help of the adapters, the solvers communicate, map data and couple via the `preCICE` process. It is an implicit coupling which means each solver will work independently and communicate not only at the coupling boundary or the interface but also at a given time stamp. In this manner, it permits high flexibility with different scenarios.

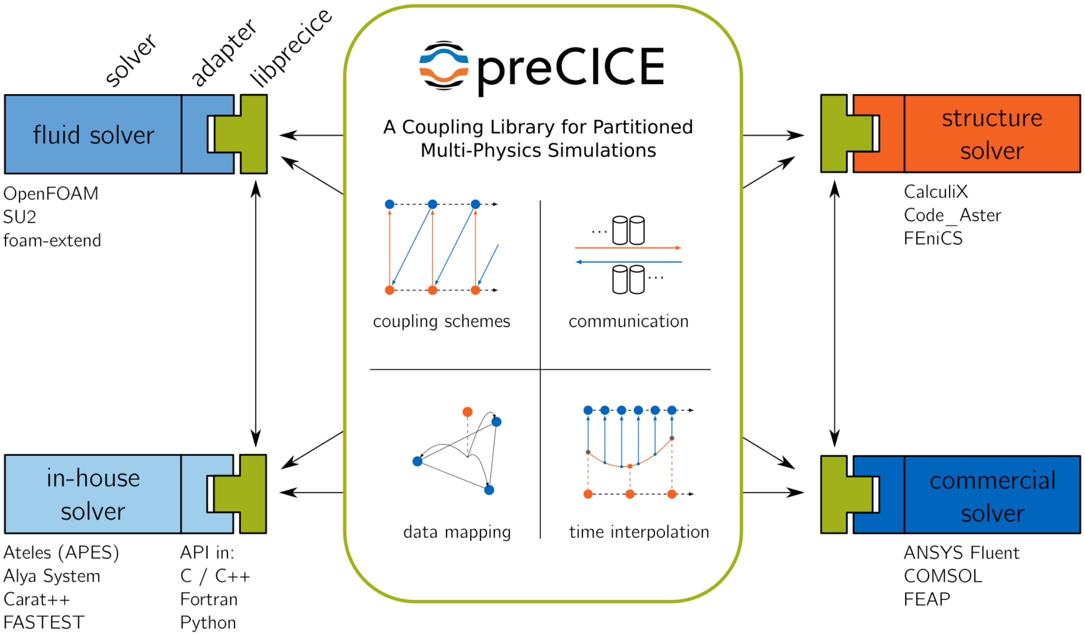


Figure 3.6: `preCICE` overview (Bungartz et al. (2016))

DLR's coupling module

The DLR's Institute of Aerodynamics and Flow Technology in Brunswick has developed a coupling module which enables two flow solvers coupling with arbitrary spatial and temporal discretization (Schwarz et al. (2010)). Independent meshes communicates at the common boundaries and flow variables of each flow solver exchange through these interfaces based on Chimera interpolation.

A two-way hybrid RANS-LES approach of the LES code from MGLET (Manhart (2004)) and the RANS code from TAU (Keye (2011) has been introduced using such coupling environment for overlapping meshes to estimate the gust effect on aircraft wake vortices by Stephan et al. (2017). The RANS domain is moving through the LES domain and both domains exchange flow data at the interfaces.

Fluid-Structure coupling

Most commonly, the Computational Structural Dynamics (CSD) problems and the CFD problems are solved separately in reference to different governing equations. For the fluid-structure interaction (FSI) problems, coupling at interfaces of fluid and structural domains is the typical approach.

In Apostolatos (2019) doctoral dissertation, a novel isogeometric mortar-based mapping method has been applied to couple the fluid and structural problems. The partitioned FSI simulation has been conducted by coupling an Isogeometric Analysis (IGA) structural model and a FVM fluid model (OpenFOAM). The coupling has been achieved by solving the fluid model as a Dirithlet problem with respect to the solution of the structural model; and solving the structural model as a Neumann problem depending on solution of the fluid model.

Such partitioned FSI approach offers best-suited specialized solution schemes for fluid and structural fields independently. The aforementioned mapping method and the coupling methodology have been both implemented in the open source software EMPIRE (Enhanced MultiPhysics Interface Research Engine).

3.6.2. OpenFOAM coupling environment

OpenFOAM itself provides several approaches for problems of interest (Bungartz et al. (2018)).

- `Framework approach` allows two different regions governed by two different solvers. Everything can be implemented inside the OpenFOAM and only OpenFOAM solver can be coupled. Each sub-solver works sequentially depending on boundary conditions computed from the other sub-solver.
- `Master-Slave approach`, however, enables the OpenFOAM solver to call external solver library that a non-OpenFOAM solver can be coupled within this approach.
- In `Files-based approach` each sub-solver solves independently and communicates only at the coupling time-step during the whole simulation. Additional script for setting up are necessary.
- If every solver calls an external library, the `Server library approach` can be applied. These solvers do not communicate directly which makes them flexible to be coupled.

Both `interFoam` and `shallowFoam` are OpenFOAM solvers, the framework approach of OpenFOAM has been chosen in the present coupled solver. Each solver has its target solving regions (meshes) as demonstrated in figure 3.7. Implementation of 2D and 3D solvers has been rearranged in the corresponding sub-regions and custom boundary condition types have been investigated to transfer the information between each sub-solver. In addition, region pointers has to be defined which combing the sub-regions and the sub-solvers.

3.6.3. `shallowInterFoamOL` solution procedure

Figure 3.8 presents the solution procedure in the coupled solver using the staggered solution procedure (Felippa (2001)), which the previous 3D solution is used to update the 2D boundary and the new 2D solution is then used as a boundary condition factor to the 3D domain.

The coupling algorithm proposed by Mintgen (2017) and Mintgen and Manhart (2018) is used in this work as a reference, the resulting solver is named `shallowInterFoam`. It uses combinations of Dirichlet-Neumann boundary conditions at the 2D-3D interface depending on the flow situations. A different approach is developed in this study which overlaps subdomains allowing Dirichlet conditions regardless of flow conditions (*i.e.* supercritical flows or subcritical flows), yields `shallowInterFoamOL`. Fully explanation of both coupling methods will be detailed in chapter 4.

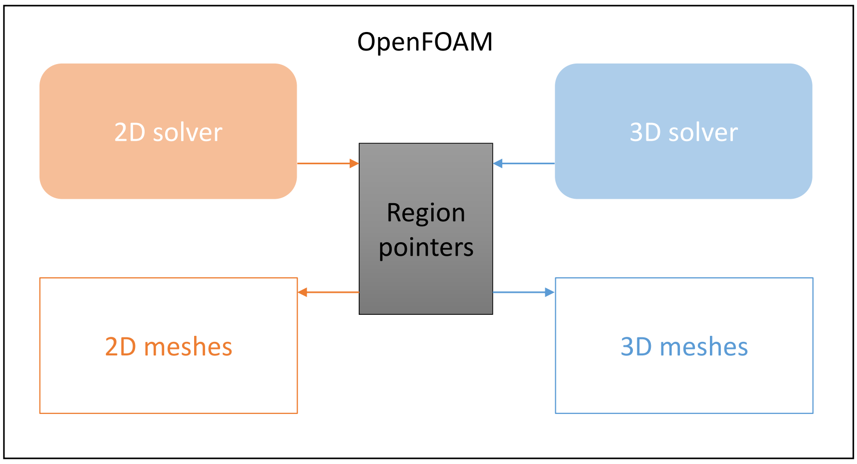


Figure 3.7: Coupling framework of `shallowInterFoamOL`

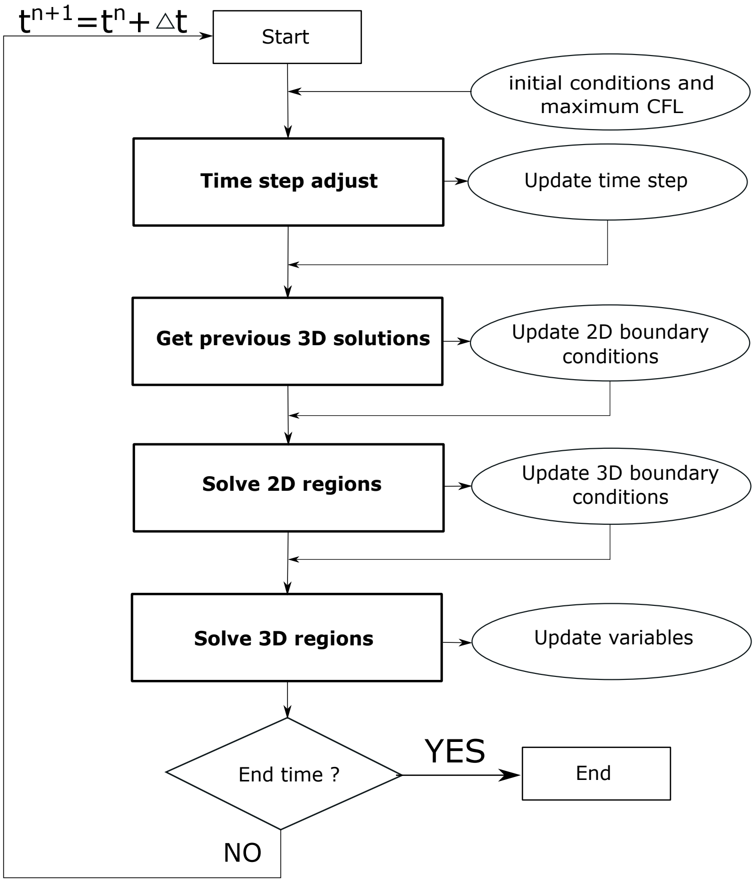

$t^{n+1}=t^n+\triangle t$
Start
initial conditions and maximum CFL
Time step adjust
Update time step
Get previous 3D solutions
Update 2D boundary conditions
Solve 2D regions
Update 3D boundary conditions
Solve 3D regions
Update variables
End time ?
YES
End
NO

Figure 3.8: Solution procedure in `shallowInterFoamOL`

4. Development of a new 2D-3D coupling algorithm

A coupling algorithm generally requires domain decomposition regarding the mesh structure and the coupling method. The whole computing domain is decomposed into a set of sub-domains in which regions for 2D and 3D sub-domains are governed by corresponding sub-solvers. Solutions at the common region or interface from sub-solvers are then exchanged and updated via the coupling method. This chapter develops a new 2D-3D coupling algorithm for the coupled solver `shallowInterFoamOL`. Several coupling algorithms will be introduced to better explain the development procedure.

These coupling algorithms use different coupling methods and mesh structures, while they share the same mapping functions between 2D and 3D flow quantities implemented by Mintgen (2017). Section 4.1 will focus on these mapping functions.

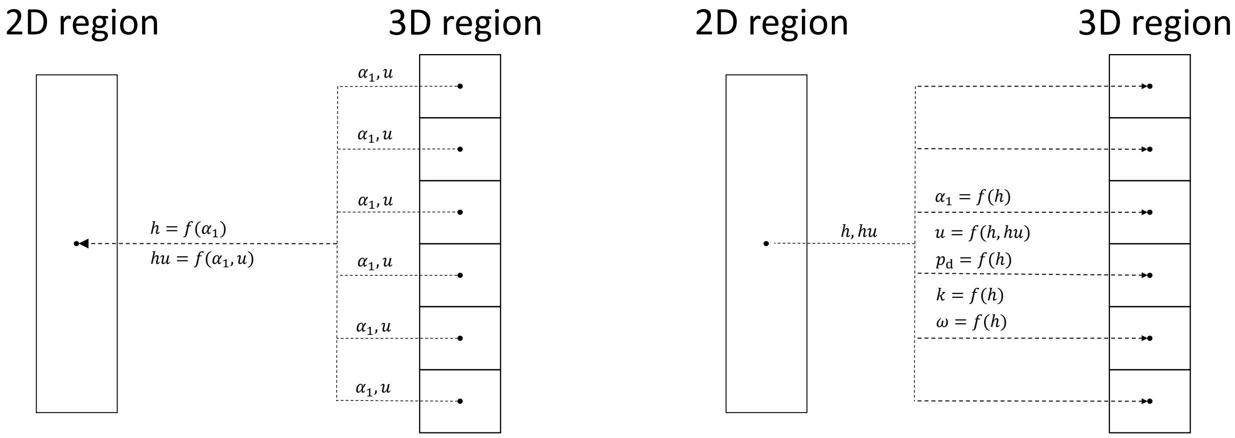
The characteristics-based coupling algorithm proposed by Mintgen (2017) will be explained first in section 4.2 as the starting point of this work. Due to its main drawback in upstream-travelling wave capture, three ad-hoc modifications were tested and will be described in section 4.3. Finally, an improved coupling algorithm using the overlapping method which overcomes the aforementioned fault was developed and will be detailed in 4.4.

Each coupling algorithm mentioned in this chapter will be further tested in a simple case to show their performance in an upstream positive surge travelling over an approaching supercritical flow. Results of the upstream-travelling surge from 5 - 20 [s] simulated by each coupling algorithm will be compared with the fully 2D and 3D simulations. 2D simulation results are from now on denoted as `2D` and 3D simulation results are from now on denoted as `3D`. Full details of the test case and more results will be given in section 5.2 of chapter 5.

4.1. Flow quantity mappings between 2D and 3D

In accordance with the governing equations of each sub-solver, three variables in the 3D sub-solver (*i.e.* u, α_1 and p_d) and two variables in the 2D sub-solver (*i.e.* h and hu) have to communicate. The flow discharge information u and α_1 interrelate with hu, while α_1 and p_d are related with h to represent the flow depth. Moreover, turbulent parameters in the 3D sub-solver shall be coupled as well. The $k-\omega$ SST turbulence model was implemented in the coupled solver with two turbulent parameters: the turbulent frequency ω and the turbulent kinetic energy k.

Solutions from one sub-solver cannot be read and recognised directly by the other one, and vice versa. Therefore, mapping functions for data transfer must be determined. As described in appendix C, meshes in OpenFAOM are constructed differently for the 2D and 3D solvers. In the vertical direction of the mesh, 2D meshes have only one cell while 3D meshes consist of multiple cells. As a consequence, the 3D data are integrated over the cells column and passed to the 2D region, while the 2D data are redistributed into the 3D cells. The mapping diagrams between each sub-solver are demonstrated in figure 4.1, and will be elaborated in the following.



a: From 3D to 2D

b: From 2D to 3D

Figure 4.1: Mapping diagrams between 2D and 3D flow quantities

4.1.1. Mapping functions

Since the cell height of a 2D cell is irrelevant, therefore the mesh information is considered only in 3D regions. A_{xy} is the area of a horizontal face. Δz is the single cell height and z_{c} is the center elevation of the cell / face (depending on the coupling algorithm used).

In order to better depict the mapping approach, variables for 3D and 2D regions are denoted with corresponding superscriptions (*i.e.* $^{\mathrm{3D}}$ and $^{\mathrm{2D}}$). As displayed in table 4.1, these variables are sorted to two categories: inner variables, which are solved by the sub-solver inside the sub-region; and mapping variables, which are calculated after inner variables are solved and then mapped to the neighbour sub-region. z_{b} is the bottom elevation and z_{w} is the absolute water level.

Category	2D variable	3D variable
Inner	h^{2D}	α_1^{3D}
	hu^{2D}	u^{3D}
	u^{2D}	$p_{\mathrm{d}}^{\mathrm{3D}}$
	$z_{\mathrm{b}}^{\mathrm{2D}}$	k^{3D}
	$z_{\mathrm{w}}^{\mathrm{2D}}$	ω^{3D}
	-	$z_{\mathrm{b}}^{\mathrm{3D}}$
Mapping	$\alpha_1^{\mathrm{2D}}(z_{\mathrm{c}})$	h^{3D}
	$u^{\mathrm{2D}}(z_{\mathrm{c}})$	hu^{3D}
	$p_{\mathrm{d}}^{\mathrm{2D}}$	$z_{\mathrm{w}}^{\mathrm{3D}}$
	$k^{\mathrm{2D}}(z_{\mathrm{c}})$	-
	$\omega^{\mathrm{2D}}(z_{\mathrm{c}})$	-

Table 4.1: Computing variables in 2D and 3D regions

Calculated flow depth from 3D regions (h^{3D})

Cell height may change along the 3D cell column, thus the flow depth in 3D region h^{3D} can be obtained by integrating the VOF indicator α_1^{3D} over the cell column.

$$h^{\mathrm{3D}} = \int \alpha_1^{\mathrm{3D}} \ \mathrm{d}\Delta z \tag{4.1.1}$$

Calculated discharge from 3D regions (hu^{3D})

Multiply the flow depth by the velocity gives the discharge. Since the 2D solver requires the specific discharge, the calculated discharge from 3D regions should be further divided by the cell surface in horizontal directions.

$$hu^{\mathrm{3D}} = \frac{\int u^{\mathrm{3D}} \alpha_1^{\mathrm{3D}} \ \mathrm{d}\Delta z}{A_{\mathrm{xy}}} \tag{4.1.2}$$

Calculated interface indicator from 2D regions ($\alpha_1^{\mathrm{2D}}(z_{\mathrm{c}})$)

The location of the water surface is required to assess the interface indicator. The water surface in 2D regions is the sum of the flow depth and the bottom elevation:

$$z_{\mathrm{w}}^{\mathrm{2D}} = z_{\mathrm{b}}^{\mathrm{2D}} + h^{\mathrm{2D}} \tag{4.1.3}$$

If a water surface from 2D side is greater than the top face of a 3D cell, this 3D cell is then fully submerged by the water; if the water surface is lower then the bottom face of a 3D cell, this 3D cell contains no water; otherwise, there is an interface in this 3D cell. The volume fraction of a cell contains an interface nor not are calculated via a linear interpolation prescribed in equation (4.1.4).

$$\alpha_1^{\mathrm{2D}}(z_\mathrm{c}) = \begin{cases} 0 & z_\mathrm{w}^{\mathrm{2D}} < z_\mathrm{c} - 0.5\Delta z \\ 1 & z_\mathrm{w}^{\mathrm{2D}} > z_\mathrm{c} + 0.5\Delta z \\ \frac{z_\mathrm{w}^{\mathrm{2D}} - z_\mathrm{c}}{\Delta z} + 0.5 & other \end{cases} \tag{4.1.4}$$

Calculated 3D velocity profile from 2D regions ($u^{\mathrm{2D}}(z_\mathrm{c})$)

Velocities are combined with interface indicators to represent the flow discharge in 3D domain. Flow velocity transferring from 2D to 3D should take special care due to different velocity profiles are employed in 2D and 3D sub-solvers. As demonstrated in figure 4.2, 2D model uses a mean velocity profile calculated by equation (4.1.5) where the flow velocity remains constant over the flow depth.

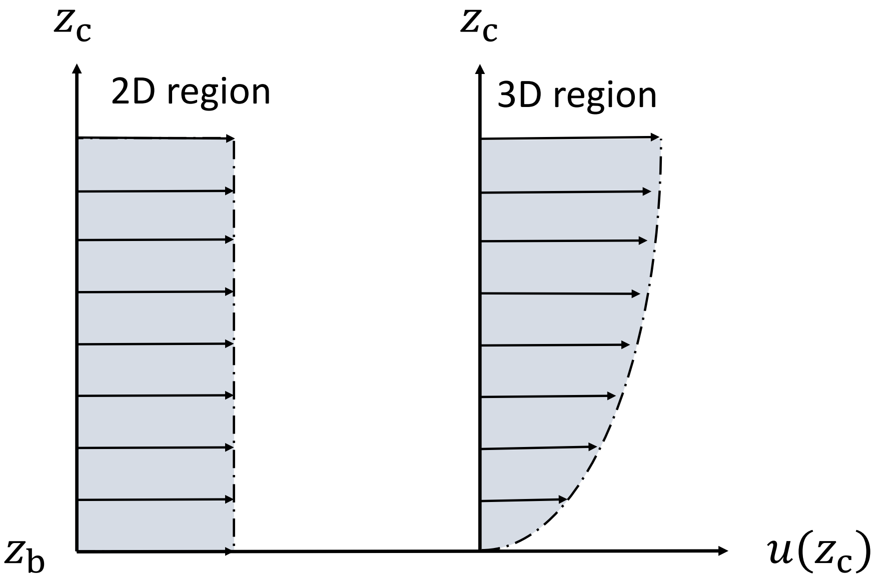


Figure 4.2: Velocity profiles in 2D and 3D sub-solvers respectively

$$u^{\mathrm{2D}} = \frac{hu^{\mathrm{2D}}}{h^{\mathrm{2D}}} \tag{4.1.5}$$

3D model applies a fully developed logarithmic profile in the flow velocity to avoid high velocity gradient near the bottom due to the wall shear stress. A logarithmic velocity profile given in equation (4.1.6) is a function of the depth-averaged velocity ($\bar{u}$), the friction velocity (u^*) and the flow depth (h). The friction velocity is defined in equation (3.5.9).

$$u(z) = \bar{u} + \frac{u^*}{\kappa}(1 + ln(\frac{z - z_\mathrm{b}}{h})) \tag{4.1.6}$$

with von-Karman's constant $\kappa = 0.41$ and the vertical position z.

The depth-averaged velocity ($\bar{u}$) can be substituted by the 2D mean velocity (u^{2D}). The friction velocity (u^*) is obtained from the deviatoric stress tensor at the bottom patch as the effects of variations of ν, which is solved from the 3D sub-solver. The flow depth can be easily taken from the 2D result h. The relative vertical position can be expressed by the difference between the present cell center position and the bottom elevation ($z_\mathrm{c} - z_\mathrm{b}^{\mathrm{3D}}$).

$$u^{\mathrm{2D}}(z_\mathrm{c}) = u^{\mathrm{2D}} + \frac{u^*}{\kappa}(1 + ln(\frac{z_\mathrm{c} - z_\mathrm{b}^{\mathrm{3D}}}{h^{\mathrm{3D}}})) \tag{4.1.7}$$

Velocities formulated by using equation (4.1.7) give an overall fitted velocity profile with no mass conservation guarantee. The flow discharge enclosed by velocity profiles indicated in figure 4.2 should be equal in both regions. However, the imposed discharge from 2D to 3D calculated via equation (4.1.8) differs from the 2D discharge hu^{2D}.

$$hu^{\mathrm{2D}}(z_\mathrm{c}) = \frac{\int u^{\mathrm{2D}}(z_\mathrm{c})\alpha_1^{\mathrm{2D}} \ \mathrm{d}\Delta z}{A_{\mathrm{xy}}} \tag{4.1.8}$$

Therefore a mass flux corrector β is introduced to express a rate of difference between these two flow discharges. In equation (4.1.9) velocities are corrected by multiplying itself with the corrector factor resulting in new velocities. New velocities are used again to calculate the updated corrector and the checking process will be repeated until a mass conservation is met.

$$\beta_\mathrm{i} = \frac{hu^{\mathrm{2D}}}{hu^{\mathrm{2D}}(z_\mathrm{c})} \quad (\mathrm{i} = 1, 2) \tag{4.1.9}$$

$$u^{\mathrm{2D}}(z_\mathrm{c}) = \beta u^{\mathrm{2D}}(z_\mathrm{c}) \tag{4.1.10}$$

Another special treatment for velocity field is to keep the velocity gradient between the air and the water phase is as mall as possible so that the maximum velocity in the water is taken as a reference to set the air velocity.

Calculated pressure from 2D regions (p_d^2D)

The pressure term is calculated as a hydrostatic pressure, where the density is the averaged density of the water and the air in 3D region (equation (3.4.7)).

$$p_\mathrm{d}^\mathrm{2D} = \rho g z_\mathrm{w}^\mathrm{2D} \tag{4.1.11}$$

Calculated turbulence parameters from 2D regions ($k^\mathrm{2D}(z_\mathrm{c})$ and $\omega^\mathrm{2D}(z_\mathrm{c})$)

Turbulence parameters, the turbulent kinetic energy k and the turbulent frequency ω, are estimated by imposing different profiles of k and ω. More details please refer to Mintgen (2017).

A slightly modified k profile is given in equation (4.1.12) which is originally taken from Nezu and Nakagawa (1993).

$$k^\mathrm{2D}(z_\mathrm{c}) = 4.0(u^*)^2 e^{-1.7 z_\mathrm{c}/h^\mathrm{2D}} \tag{4.1.12}$$

An own power-law has applied to express ω in equation (4.1.13).

$$\omega^\mathrm{2D}(z_\mathrm{c}) = \frac{6.0(u^*)(z_\mathrm{c}/h^\mathrm{2D})^{13/10}}{h^\mathrm{2D}} \tag{4.1.13}$$

4.1.2. Negative flow depth

Flow quantities are mapped from cells to cells based on these mapping functions. However, for coupling algorithms exchanging the data via the 2D-3D interface, the mapped values need to be interpolated at the interface. A linear interpolation scheme was applied in the characteristics-based method.

According to equation (4.1.7), the reference flow depths along the 3D column are required to produce the 3D velocity profile. The reference flow depth h_ref is determined by $(z_\mathrm{c} - z_\mathrm{b}^\mathrm{3D})$, where z_c is the face center elevation at the interface on 3D side and z_b is the bottom elevation at the interface. Theoretically, the bottom elevation at the interface from the 2D side $z_\mathrm{b}(\Gamma_\mathrm{2D})$ equals to the 3D side $z_\mathrm{b}(\Gamma_\mathrm{3D})$. Within the code implementation, $h_\mathrm{ref}^\mathrm{3D}$ is determined by equation (4.1.14).

$$h_\mathrm{ref}^\mathrm{3D} = z_\mathrm{c} - z_\mathrm{b}^\mathrm{2D}(\Gamma_\mathrm{2D}) \tag{4.1.14}$$

However, due to high irregular terrains in realistic cases, no exact value of bottom elevations will be prescribed at the boundary. Instead, a zero gradient interpolation scheme are more likely to be taken place resulting in $z_\mathrm{b}^\mathrm{2D}(\Gamma_\mathrm{2D}) = z_\mathrm{b}^\mathrm{2D}$, where z_b^2D is the bottom elevation of the neighbour 2D cell of the 2D interface). Therefore, in cases where 2D region is located at a higher elevation than 3D region, a negative flow depth will be obtained, resulting in mathematical error in equation 4.1.7.

For example, figure 4.3 illustrates a problematic situation that the first lowest face center elevation at the 3D interface z_c is smaller than $z_\mathrm{b}^{\mathrm{2D}}$, $h_{\mathrm{ref}}^{\mathrm{3D}}$ is then negative. This potential risk is avoided in the overlapping method.

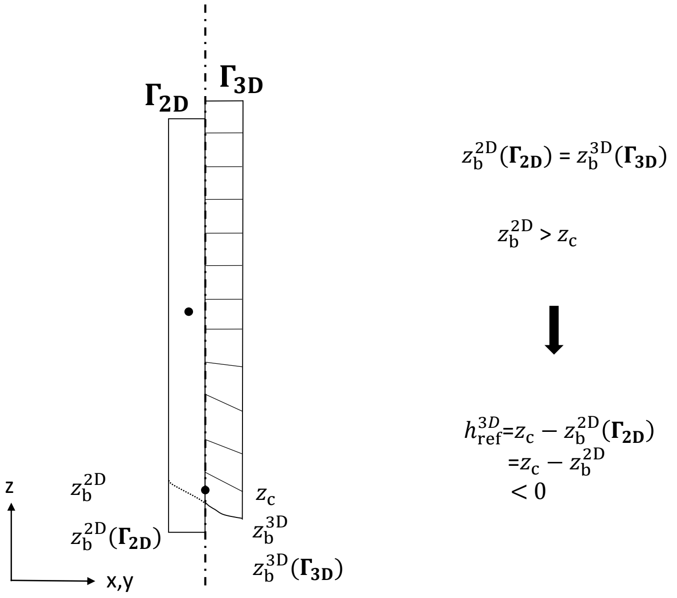

$\boldsymbol{\Gamma_{2D}}$
$\boldsymbol{\Gamma_{3D}}$
$z_\mathrm{b}^{\mathrm{2D}}(\boldsymbol{\Gamma_{2D}}) = z_\mathrm{b}^{\mathrm{3D}}(\boldsymbol{\Gamma_{3D}})$
$z_\mathrm{b}^{\mathrm{2D}} > z_\mathrm{c}$
h_{ref}^{3D}=$z_\mathrm{c} - z_\mathrm{b}^{\mathrm{2D}}(\boldsymbol{\Gamma_{2D}})$
=$z_\mathrm{c} - z_\mathrm{b}^{\mathrm{2D}}$
< 0
z
$z_\mathrm{b}^{\mathrm{2D}}$
$z_\mathrm{b}^{\mathrm{2D}}(\boldsymbol{\Gamma_{2D}})$
x,y
z_c
$z_\mathrm{b}^{\mathrm{3D}}$
$z_\mathrm{b}^{\mathrm{3D}}(\boldsymbol{\Gamma_{3D}})$

Figure 4.3: A problematic example of bottom elevation calculation

4.2. Characteristics-based algorithm

The multiple-regions problem with the non-overlapping mesh was implemented within the coupling framework provided by OpenFOAM by means of the well-known Dirichlet-Neumann domain decomposition method (Quarteroni and Valli (1999)). This method exchanges the information through the interface (the common boundary) between regions based on the characteristics of the flow information transfer directions. Tested results from this coupling algorithm will be noted as `CB 2D3D`. More details of this method can be found in Mintgen (2017).

4.2.1. Mesh representation

In the context of the domain decomposition method, the global domain Ω is subdivided into a number of non-overlapping smaller sub-domains: in figure 4.4 (a), Ω_{2D} represents the 2D region and Ω_{3D} represents the 3D region. The common boundary Γ connected to each region is denoted as the interface at the 2D side Γ_{2D} and the interface at the 3D side Γ_{3D}. Neighbouring regions communicate through the interface Γ_{2D} and Γ_{3D} therefore the mesh structure should be matched at the interface in a way that information from both sides can be transferred correctly.

As a consequence of the difference in structure of the 2D and 3D meshes, the meshes must match horizontally but not vertically. In contrast with 3D meshes representing the real geometry of the domain, 2D meshes are built by cuboid cells with an unit height (the default value is 1 [m]) and the boundaries normal to the z-direction (*i.e.* top and bottom) are totally flat. Figures 4.4 (b) and (c) outline the setting up of the coupling meshes, indicating that the adjacent cells have to match only in the xy-plane. Information transfer in the vertical dimension takes place by means of the spatial interpolation from Ω_{3D} to Ω_{2D} and the redistribution from Ω_{2D} to Ω_{3D}.

4.2.2. Types of boundary condition

The coupling was achieved by exchanging flow variables depending on the respective conditions. The Dirichlet-Neumann coupling is employed, based on an exchange of values and fluxes at the 2D-3D interface. When the information is entering the domain from the outside, the Dirichlet boundary condition is applied; that a defined value has to be specified explicitly to the boundary. In contrast, when the information is leaving the domain, the value at the boundary can be extrapolated by imposing the respective normal gradient on the boundary via the Neumann boundary condition. Hence the information transfer is always going from the side of the interface with the Neumann condition to the side with the Dirichlet condition.

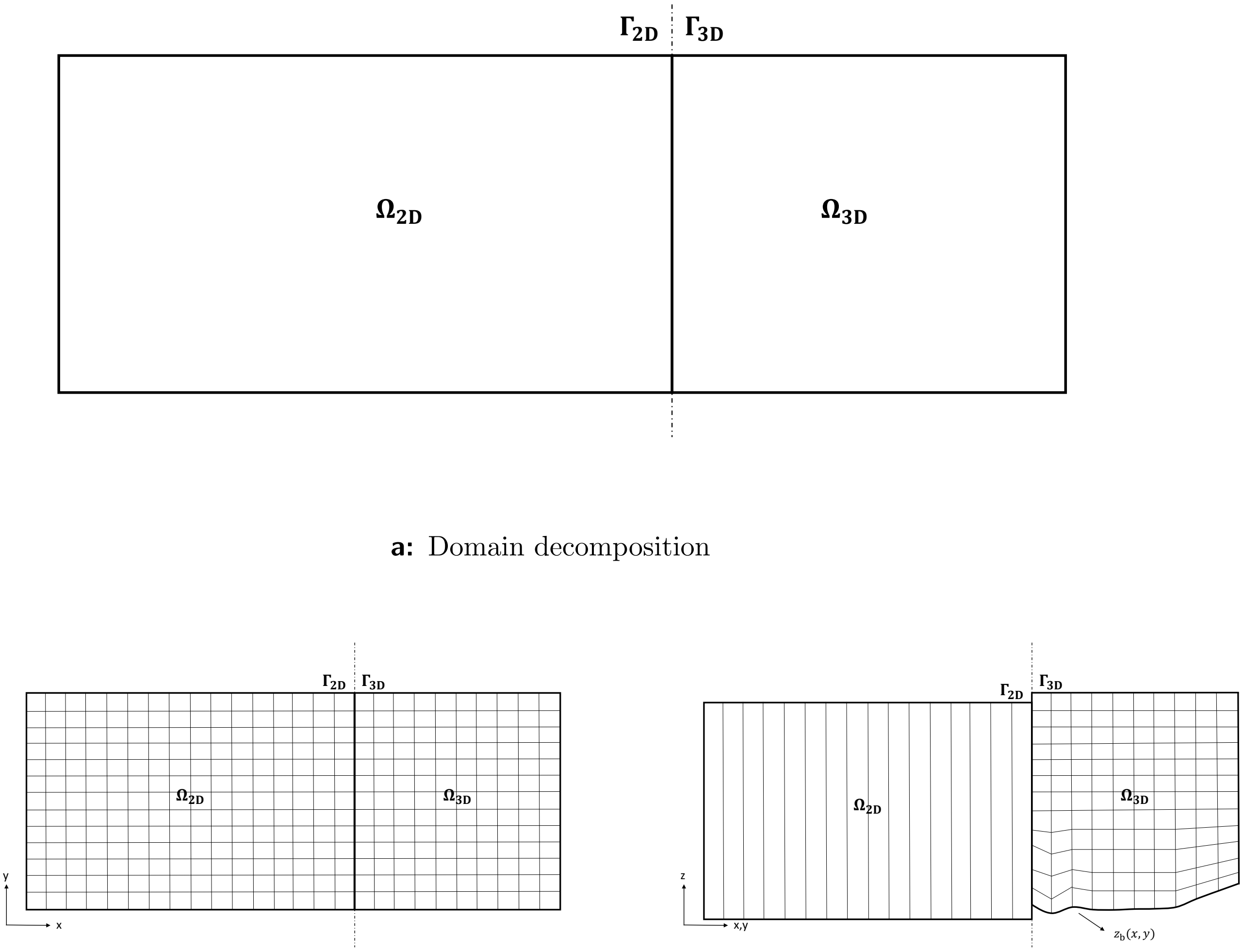
In OpenFOAM the Dirichlet-Neumann boundary condition is controlled by a weighted fraction value ω resulting in a mixed boundary condition. Equation (4.2.1) defines the mixed boundary function for the target cell (the left cell in figure 4.5) and parameters are identified in figure 4.5. Two adjoint cells are connected via the common face, where the target face value Φ_{f} is determined. Φ_c is the cell center value taken from the target cell, Φ_{ref} is the reference cell value taken from the neighbouring cell, $\frac{\Phi}{\mathbf{n}}_{\mathrm{ref}}$ is the reference gradient between Φ_{c} and Φ_{ref}, Δd_{c} and Δd_{ref} are distances from the cell center to the face center in the target cell and the neighbouring cell respectively.

$$\Phi_{\mathrm{f}} = \omega \cdot \Phi_{\mathrm{ref}} + (1-\omega) \cdot \Phi_c + \frac{\Phi}{\mathbf{n}}_{\mathrm{ref}} \cdot \Delta d_{\mathrm{c}} \tag{4.2.1}$$

where,

$$\frac{\Phi}{\mathbf{n}}_{\mathrm{ref}} = \frac{\Phi_{\mathrm{ref}} - \Phi_{\mathrm{c}}}{\Delta d_{\mathrm{ref}} + \Delta d_{\mathrm{c}}} \tag{4.2.2}$$

In the context of the implementation, the face value is the value at the interface denoted by



a: Domain decomposition

b: Coupling mesh top view

c: Coupling mesh side view

Figure 4.4: Non-overlapping coupling mesh representation

Φ_{if}, cell values are taken from the cells next to the interface, *i.e.* Φ_{2D} and Φ_{3D}. ω is one for the Dirichlet boundary condition and is zero for the Neumann boundary condition.

By using the non-overlapping mesh, two regions are bounded by the interfaces which are highlighted in red as shown in figure 4.6. Under the circumstances of subcritical flow from Ω_{3D} to Ω_{2D}, the flow depth is controlled by the hydrostatics pressure from downstream, therefore the pressure term is prescribed from z_{w}^{2D} which means a Dirichlet boundary condition is adopted to the p_{d} and a Neumann boundary condition is applied to α_1.

The Dirichlet-Neumann approach used in the coupling algorithm demands the information transferring directions, which is not only dependent on the flow directions but is also related to the flow conditions. Based on the flow direction from one region to the other and the

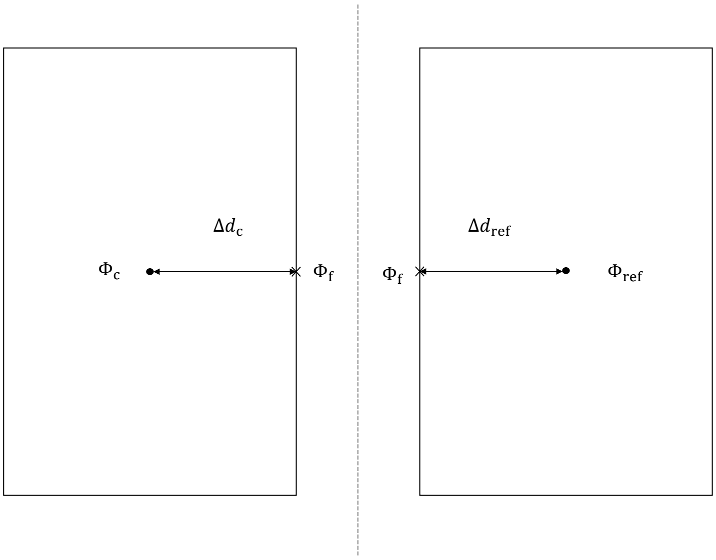


Figure 4.5: Sketch of neighbouring cells

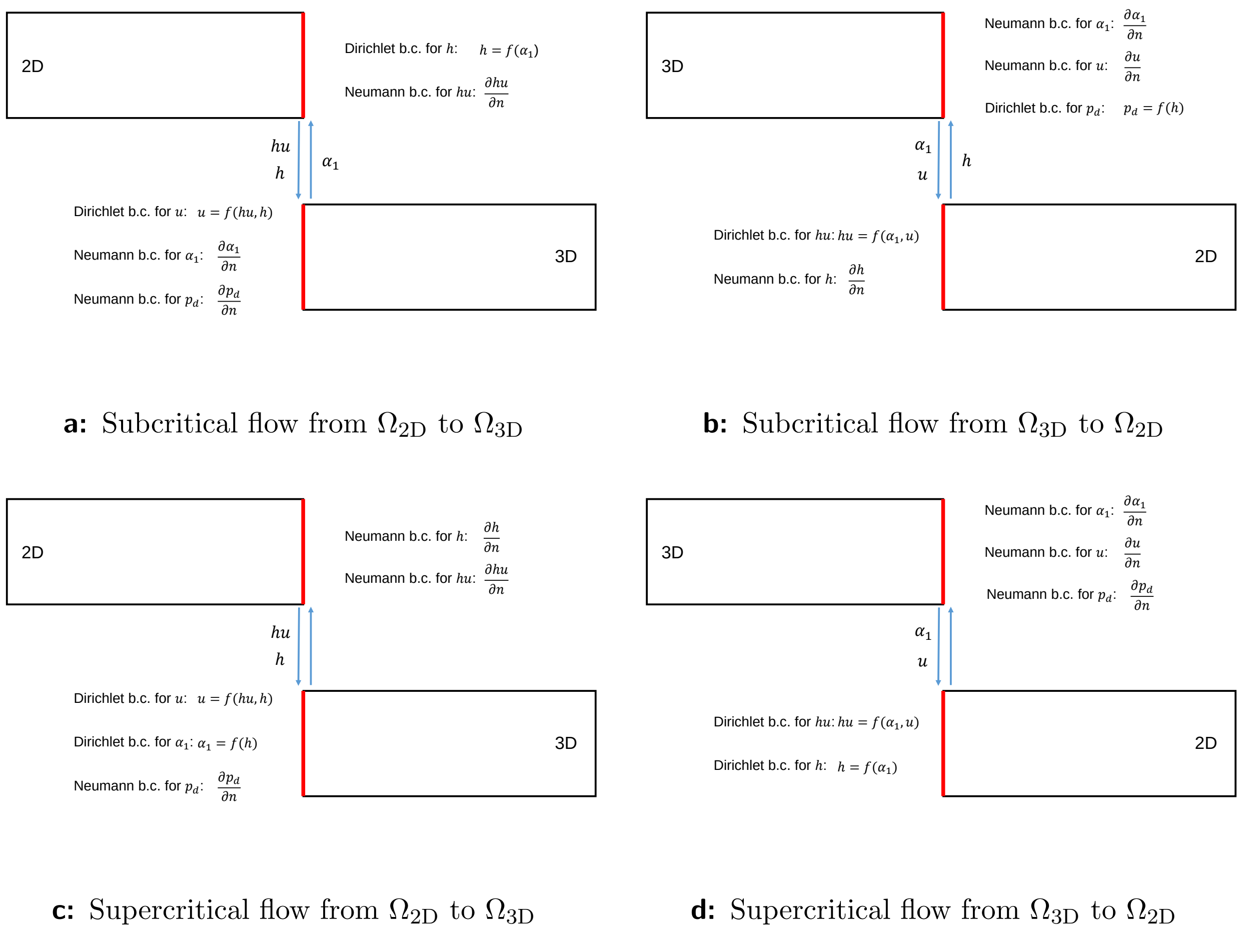
flow condition characteristics, four general sub-cases were considered during the coupling: subcritical flows from Ω_{2D} to Ω_{3D}, supercritical flows from Ω_{2D} to Ω_{3D}, subcritical flows from Ω_{3D} to Ω_{2D}, and supercritical flows from Ω_{3D} to Ω_{2D}. The flow condition and direction at the interface is determined by the Froude number Fr, which is calculated at the interface using the interpolated values from both regions (equation (4.2.3)). The flow is supercritical if $|Fr| > 1$ and when $|Fr| < 1$ the flow is subcritical. Meanwhile, the velocity normal vector $\mathbf{n}$ is considered to determine the flow directions, $Fr > 0$ represents an outflow and $Fr < 0$ implies an inflow.

$$Fr_{\text{if}} = \frac{(u^{\text{2D}}\Delta d^{\text{3D}} + u^{\text{3D}}\Delta d^{\text{2D}}) \cdot \mathbf{n}}{\sqrt{g(h^{\text{2D}}\Delta d^{\text{3D}} + h^{\text{3D}}\Delta d^{\text{2D}})}} \tag{4.2.3}$$

4.2.3. Hydraulic jump blocking

This coupling algorithm works well in most configurations. However, a problem may arise when the flow condition changes from supercritical to subcritical (*e.g.* hydraulic jump) especially in cases where a steep gradient wave front of back water is developed (*e.g.* the positive surge towards upstream). Such an upstream-travelling hydraulic jump passing through the 2D-3D interface cannot be properly modelled in this implementation. The tested example from the section 5.2 is displayed in figure 4.7. A fully closed outlet is set up for the outlet boundary based on a steady state supercritical flow condition. The back wave is then generated at the outlet and transfers upstream. Results from 5 - 20 [s] show the blockage of the upstream-travelling wave at the interface located at x = 50 [m] where the flow depth keeps increasing in the 3D region, while the purely 2D and 3D waves advance through the interface.

A hydraulic jump occurs when a supercritical flow and a subcritical flow meet. The reason


2D
Dirichlet b.c. for h: $h = f(\alpha_1)$
Neumann b.c. for hu: $\frac{\partial hu}{\partial n}$
hu
h
α_1
Dirichlet b.c. for u: $u = f(hu, h)$
Neumann b.c. for α_1: $\frac{\partial \alpha_1}{\partial n}$
Neumann b.c. for p_d: $\frac{\partial p_d}{\partial n}$
3D
3D
Neumann b.c. for α_1: $\frac{\partial \alpha_1}{\partial n}$
Neumann b.c. for u: $\frac{\partial u}{\partial n}$
Dirichlet b.c. for p_d: $p_d = f(h)$
α_1
u
h
Dirichlet b.c. for hu: $hu = f(\alpha_1, u)$
Neumann b.c. for h: $\frac{\partial h}{\partial n}$
2D
2D
Neumann b.c. for h: $\frac{\partial h}{\partial n}$
Neumann b.c. for hu: $\frac{\partial hu}{\partial n}$
hu
h
Dirichlet b.c. for u: $u = f(hu, h)$
Dirichlet b.c. for α_1: $\alpha_1 = f(h)$
Neumann b.c. for p_d: $\frac{\partial p_d}{\partial n}$
3D
3D
Neumann b.c. for α_1: $\frac{\partial \alpha_1}{\partial n}$
Neumann b.c. for u: $\frac{\partial u}{\partial n}$
Neumann b.c. for p_d: $\frac{\partial p_d}{\partial n}$
α_1
u
Dirichlet b.c. for hu: $hu = f(\alpha_1, u)$
Dirichlet b.c. for h: $h = f(\alpha_1)$
2D

a: Subcritical flow from Ω_{2D} to Ω_{3D}

b: Subcritical flow from Ω_{3D} to Ω_{2D}

c: Supercritical flow from Ω_{2D} to Ω_{3D}

d: Supercritical flow from Ω_{3D} to Ω_{2D}

Figure 4.6: Transfer variables of four coupling flow situations in exchange boundary conditions coupling

why the upstream-travelling hydraulic jump fails to pass the 2D-3D interface is the sudden change of the flow conditions from the supercritical to the subcritical. The flow is under the supercritical condition at the beginning of the simulation, the Neumann boundary conditions are employed at the upstream side interface and the Dirichlet boundary conditions are applied to the downstream side interface. After the surge is produced downstream, the flow condition changes to the subcritical and the flow depth is no longer controlled by the upstream but by the downstream due to the subcritical flow feature.

Unfortunately, the back wave transfer stops at the coupling interface as the code does not detect the flow condition change. A characteristic of the supercritical initial flow condition is that, both discharge and flow depth in the cells next to the interface in the downstream region are imposed from the cells next to the interface in the upstream region. No information passes on from the downstream side and values for calculating Fr_{if} are not updated from the back wave. Consequently, the calculated Fr_{if} is still larger than 1 at the interface resulting in a supercritical flow condition. In other words, flow quantities at the interface are always assigned from the upstream side to the downstream side.

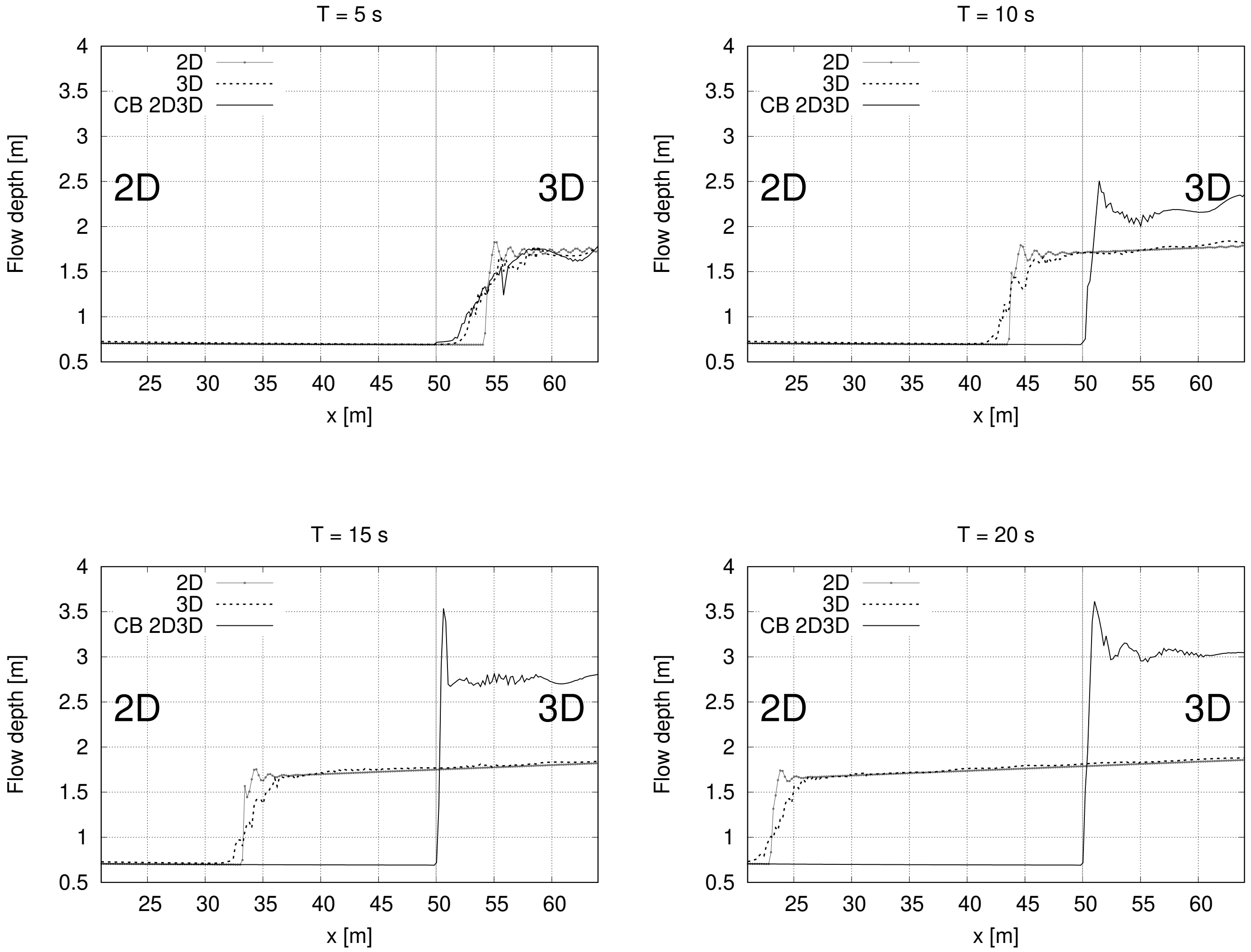


Figure 4.7: Surge travelling upstream at 5 - 20 [s] with 2D, 3D and CB 2D3D simulations

4.3. Tested ad-hoc modifications

During the development, some modifications were made as-needed to solve the hydraulic jump blocking problem with the characteristics-based method. Despite the fact that these ad-hoc tests turned out not to work, they helped to clarify the problem in a sense. These interim tests were all implemented based on the the non-overlapping mesh structure as the characteristics-based method.

4.3.1. Ad-hoc solution 1: Dirichlet boundary condition at interface

The first idea to overcome the hydraulic jump blocking uses Dirichlet boundary conditions for flow depth terms in both supercritical and subcritical flows, so that the increased hydraulic jump depth should be updated at the 2D interface. However, this does not improve the results of the characteristics-based method. Another idea to capture the back wave is using Dirichlet boundary conditions also for flow rate terms in all flow conditions, such that the wave information from downstream can be imposed on the upstream side of the interface. The surge wave is generated because of a sudden change of discharge; for this reason, only enforcing a specific flow depth at the interfaces will not correctly yield the wave information transfer. Dirichlet boundary conditions for discharge are then set regardless of the flow direction. As a result, all flow quantities use Dirichlet boundary conditions at both interfaces. To continue the simulation, the solution from the previous time to the new time, three general steps need to be performed. First of all, cell values will be computed at time n. Secondly, boundary values will be determined depending on cell values. Finally, cell values at the new time n+1 can be calculated.

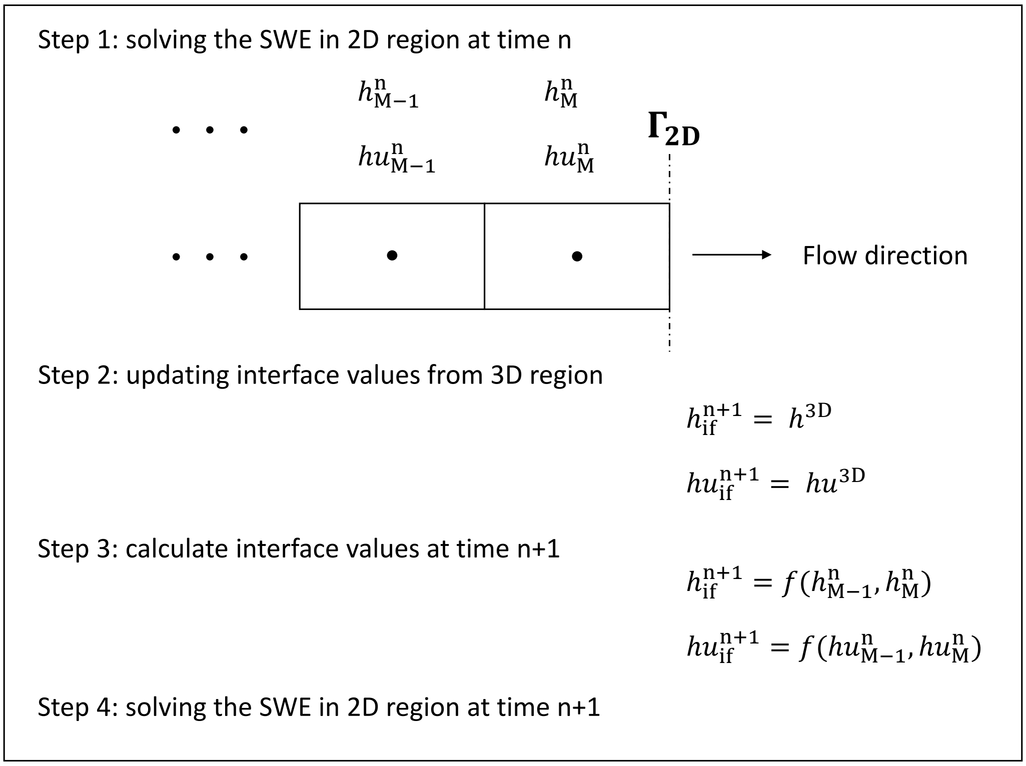

Step 1: solving the SWE in 2D region at time n
$h^{\mathrm{n}}_{\mathrm{M-1}}$
$h^{\mathrm{n}}_{\mathrm{M}}$
$hu^{\mathrm{n}}_{\mathrm{M-1}}$
$hu^{\mathrm{n}}_{\mathrm{M}}$
$\Gamma_{\mathbf{2D}}$
Flow direction
Step 2: updating interface values from 3D region
$h^{\mathrm{n+1}}_{\mathrm{if}} = h^{\mathrm{3D}}$
$hu^{\mathrm{n+1}}_{\mathrm{if}} = hu^{\mathrm{3D}}$
Step 3: calculate interface values at time n+1
$h^{\mathrm{n+1}}_{\mathrm{if}} = f(h^{\mathrm{n}}_{\mathrm{M-1}}, h^{\mathrm{n}}_{\mathrm{M}})$
$hu^{\mathrm{n+1}}_{\mathrm{if}} = f(hu^{\mathrm{n}}_{\mathrm{M-1}}, hu^{\mathrm{n}}_{\mathrm{M}})$
Step 4: solving the SWE in 2D region at time n+1

Figure 4.8: Solution procedure in 2D region for one time step

However, on the basis of applying the upwind scheme for the convective terms in the 2D sub-solver, flow information is propagated in a stream-wise manner. If Ω_{2D} is located further upstream than Ω_{3D}, problems may occur. A complete solution procedure in Ω_{2D} from time n to time n+1 is listed in figure 4.8. Quantities of interest (h and hu) for every cell are solved at time n. After that, an additional step (step 2) is performed as a part of the coupling algorithm, in which 2D interface values will be assigned from 3D values. Then all boundary values including the 2D interface values will be calculated by the 2D sub-solver (step 3). Cell values combined with boundary values are then used to program the new time step values. When the flow travels from Ω_{2D} to Ω_{3D}, step 2 and step 3 conflict that all imposed data from

3D sides will not give a smart impact on the upstream field during the simulation. Thus, the back wave information is still missing in Ω_{2D}.

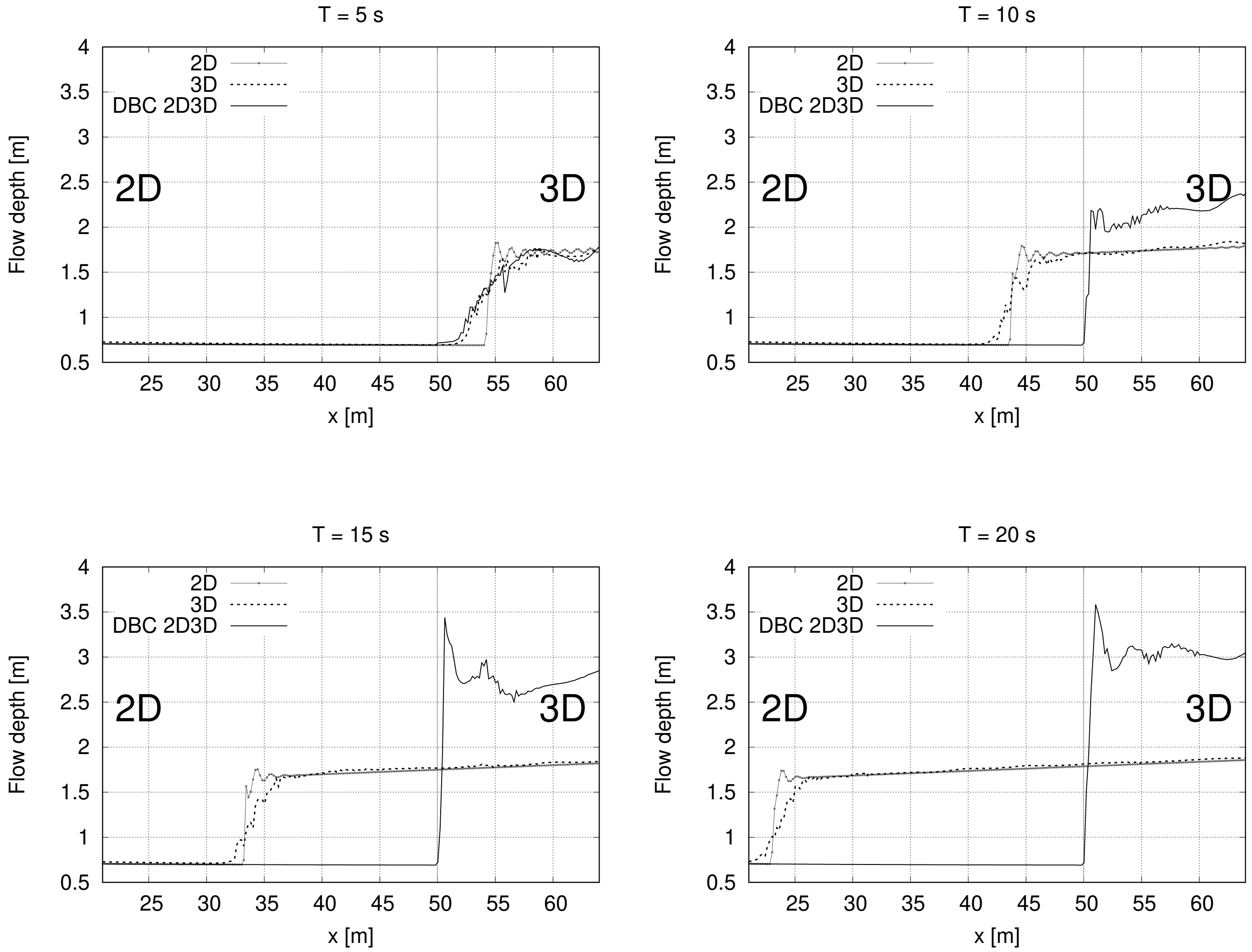


Figure 4.9: Surge travelling upstream at 5 - 20 [s] with 2D, 3D and DBC 2D3D simulations

As illustrated in figure 4.9, a similar behaviour is expressed as in figure 4.7; the back wave fails to pass the interface. Results from the present ad-hoc coupled solver is noted as `DBC 2D3D`.

4.3.2. Ad-hoc solution 2: Discontinous wave correction

The hydraulic jump can be classified as a stationary hydraulic jump or a the moving hydraulic jump: The phenomenon that the flow transitions from the rapidly flowing water to the slowly moving water is termed the stationary hydraulic jump. The moving hydraulic jump also called the tidal bore when an obstacle is suddenly introduced against the water flowing downstream. Hydraulic jumps cause energy losses which means the momentum equation is used instead of the energy equation in the analysis. In the open channel unsteady rapidly varied flow, the discontinuous wave is an important feature. Depending upon the substantial large change in the flow rate in a short time, the flow depth in the open channel changes abruptly, which produces a surge travelling in the upstream or the downstream direction. This kind of surge shows a discontinuity of the water surface. Typical examples of discontinuous waves are:

- Dam break wave due to failure of dams
- Tidal bore in estuaries or inlets
- Surge wave generated by starting or stopping a hydro power plant

The upstream-travelling surge is the problem of interest in this study. A control volume of rectangular open channel with a moving upstream discontinuous wave is displayed in figure 4.10. Variables at the inlet are labelled with subscript `in` and variables at the outlet are labelled with subscript `out`. The wave speed at the wave surface is called the wave celerity u_{w} [$\mathrm{m} \cdot \mathrm{s}^{-1}$].

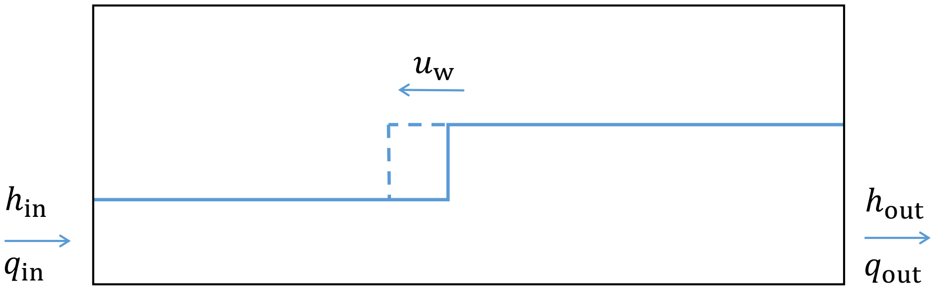


Figure 4.10: Upstream positive surge in an rectangular open channel

A sudden decreased discharge after the surge is derived based on the continuity equation:

$$q_{\mathrm{in}} = q_{\mathrm{out}} + u_{\mathrm{w}}(h_{\mathrm{out}} - h_{\mathrm{in}}) \tag{4.3.1}$$

and the wave celerity is calculated as a result of the momentum equation (full derivation can be found in appendix D):

$$u_{\mathrm{w}} = \sqrt{g h_{\mathrm{out}} \left(\frac{h_{\mathrm{in}} + h_{\mathrm{out}}}{2 h_{\mathrm{out}}}\right)} \tag{4.3.2}$$

However, in our Shallow Water solver the wave speed is determined only by the flow depth at the wave (equation (4.3.3)), thus, the sudden change in discharge cannot be represented.

$$u_{\mathrm{w}}^{2\mathrm{D}} = \sqrt{gh^{2\mathrm{D}}} \tag{4.3.3}$$

In order to eliminate the wave speed difference between the shallow water wave and the discontinuous wave, wave celerity correction terms are added to calculate the Froude number and the specific discharge at the bore.

The corrected discharge is:

$$hu_{\mathrm{w}} = hu^{2\mathrm{D}} - u_{\mathrm{w}}(h^{3\mathrm{D}} - h^{2\mathrm{D}}) \tag{4.3.4}$$

ans the corrected Fr is:

$$Fr_{\mathrm{w}} = \frac{u^{2\mathrm{D}} - u_{\mathrm{w}}}{\sqrt{gh^{2\mathrm{D}}}} \tag{4.3.5}$$

Fr_{w} is then used to determine the flow condition at the interface and hu_{w} is used to check the flow direction.

Figure 4.11 expresses the failure of the back wave to travel though the interface on the same test case. `DWC 2D3D` is used to indicate results from the discontinuous wave correction method. This method is unable to work because the corrected Fr_{w} only affects the boundary type at interfaces. When a hydraulic bore is moving upstream from $\Omega_{3\mathrm{D}}$ to $\Omega_{2\mathrm{D}}$, a subcritical flow condition can be determined by the corrected Fr_{w} at the interface. However, the flow condition in $\Omega_{2\mathrm{D}}$ is still supercritical. As in subsection 4.3.1, information in $\Omega_{2\mathrm{D}}$ will not be effected by $\Gamma_{2\mathrm{D}}$.

4.3.3. Ad-hoc solution 3: Hydraulic jump detection

Imposing values at the interface are not helping in imposing the information transfer direction. The main reason leads to this failure is the lack of information from downstream. In order to integrate the code ability in hydraulic jump detection, a new determinant is added.

As mentioned above, the direction of flow depth information transfer is decided by Fr while the discharge or the flow velocity information is always transfer stream-wise as long as no input flow rate is coming from the outlet of the channel. Although the surge wave gives rise to a partially negative flow rate (towards upstream), the total flow rate is always from upstream (2D region) to downstream (3D region). The discharge is the original drive element. The block of discharge information is the key factor stopping the upstream-travelling hydraulic jump.

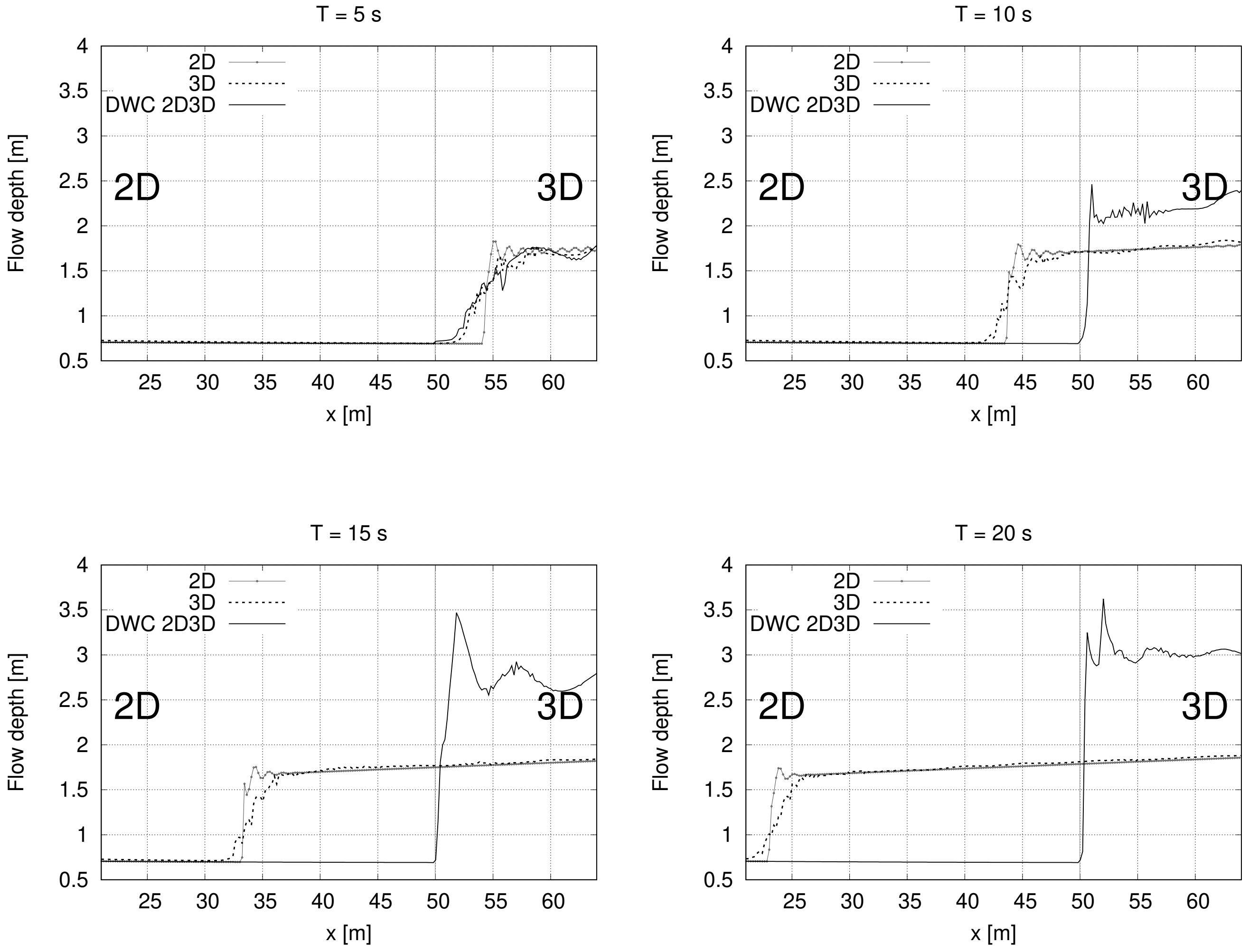


Figure 4.11: Surge travelling upstream at 5 - 20 [s] with 2D, 3D and DWC 2D3D simulations

For sake of bringing the back wave information to the interface, whether a hydraulic jump is generated shall be estimated. Conjugate depths (equation 4.3.6) are the depths (y_1) upstream and the depth (y_2) downstream of the hydraulic jump whose momentum functions are equal for a given unit discharge q.

$$y_2 = \frac{y_1}{2}(\sqrt{1 + 8Fr_1^2} - 1) \tag{4.3.6}$$

Many empirical relationships for length of the normal hydraulic jump can be found. In the ad-hoc code, the hydraulic jump length equation 4.3.7 is implemented (dan Beeba (1917)).

$$L_{\mathrm{j}} = 5(y_2 - y_1) \tag{4.3.7}$$

The new criteria is proceeding as follows. The flow information changes if and only if an upstream-travelling hydraulic jump is detected from downstream of the interface. A flow chart and the sketch of influence zone are shown in figures 4.12 and 4.13.

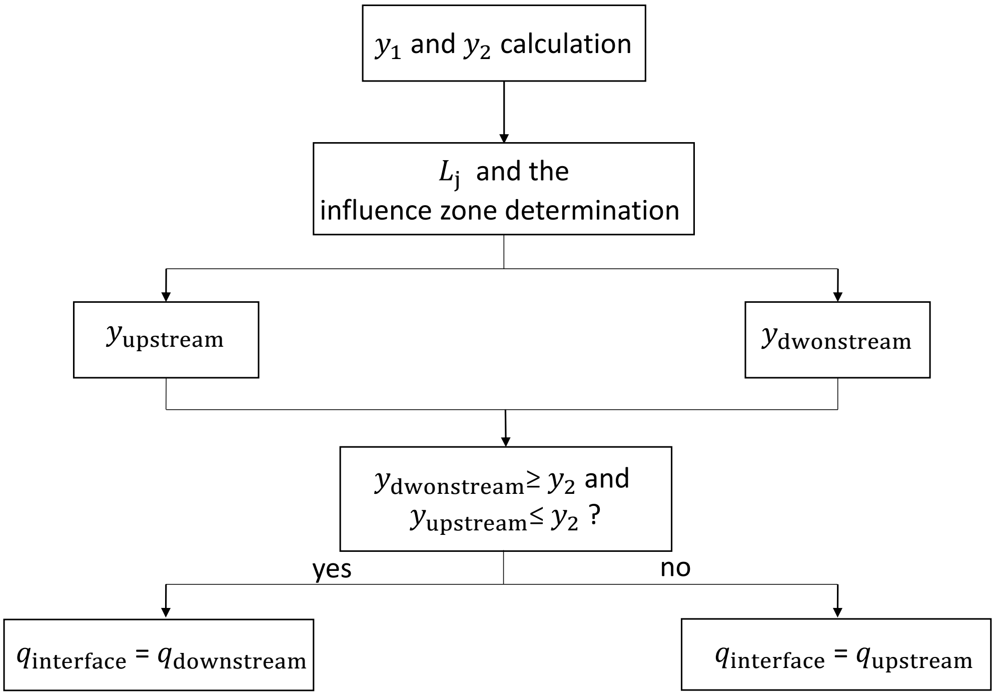

y_1 and y_2 calculation
L_j and the influence zone determination
y_upstream
$y_\text{dwonstream}$
$y_\text{dwonstream} \geq y_2$ and $y_\text{upstream} \leq y_2$?
yes
no
$q_\text{interface} = q_\text{downstream}$
$q_\text{interface} = q_\text{upstream}$

Figure 4.12: Flow chart of the new criteria for discharge information direction determine

- Calculate the corresponding conjugate depth y_2 based on the steady state upstream flow depth y_1 using equation (4.3.6). y_1 is obtained by taking the steady state flow depth far upstream of the interface.
- Define the estimated hydraulic jump length L_j by applying equation (4.3.7).
- Fix the influence zone of hydraulic jump where L_j away from the interface in both upstream and downstream directions. A hydraulic jump would be supposed to appear or disappear in the influence zone if some determinants are satisfied. The influence zone is marked in the dashed box in figure 4.13.
- Identify flow depths at the influence zone boundaries. $y_\text{downstream}$ is the flow depth that L_j away downstream from the interface, while y_upstream is the flow depth that L_j away upstream from the interface.
- Determine whether $y_\text{downstream}$ or y_upstream reaches the corresponding conjugate depth y_2. If $y_\text{downstream} > y_2$, an upstream-travelling hydraulic jump is entering the influence zone. If $y_\text{upstream} > y_2$, a hydraulic jump is leaving the influence zone.
- When $y_\text{downstream} > y_2$ and $y_\text{upstream} < y_2$, the upstream-travelling hydraulic jump in the influence zone is detected and the flow discharge information at the interface $q_\text{interface}$

shall be updated from the downstream side $q_{\text{downstream}}$. Otherwise, $q_{\text{interface}}$ remains the same as it is before (*i.e.* $q_{\text{interface}} = q_{\text{upstream}}$).

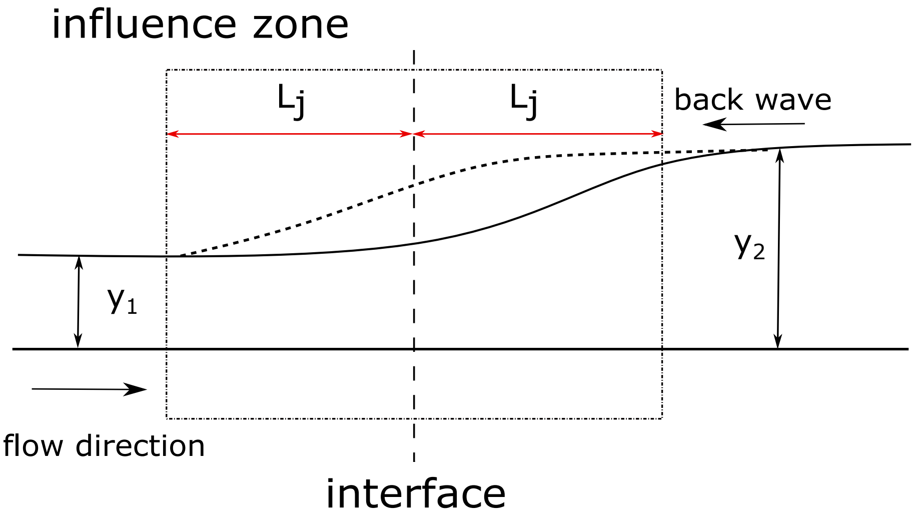


Figure 4.13: Sketch map of influence zone in open channel flow

This method is able to let the back wave moving through the interface but an obvious blockage can be noticed at the early time during the wave transfer phase. `HJD 2D3D` is used as an abbreviation of the hydraulic jump detection. In figure 4.14, the `HJD 2D3D` back wave is stopped by the interface at T = 10 [s] while purely 2D and 3D waves advance without any hindrance. At T = 15 [s], the `2D3D` wave passes the interface and speeds up in Γ_{2D}. The wave arriving time lag in the coupled results is compensated in a later time. The reason why a severe delay appears in the coupled simulation is the hydraulic jump length determination. It is evident that the hydraulic jump length equation (4.3.7) implemented here is shorter than the real length in the test case. However, a proper hydraulic jump length is hardly defined only a one single equation.

Moreover, when applying such method to a more complex problem, the back wave may get totally blocked at the interface. For instance, when an obstacle is located in a river with irregular bottom elevations, part of the fluids hit the obstacle and is reflected upstream while others move around the obstacle and travel downstream. Under such configuration, the calculated conjugate depth is not applicable any more and wrong prediction of the hydraulic jump length will be produced.

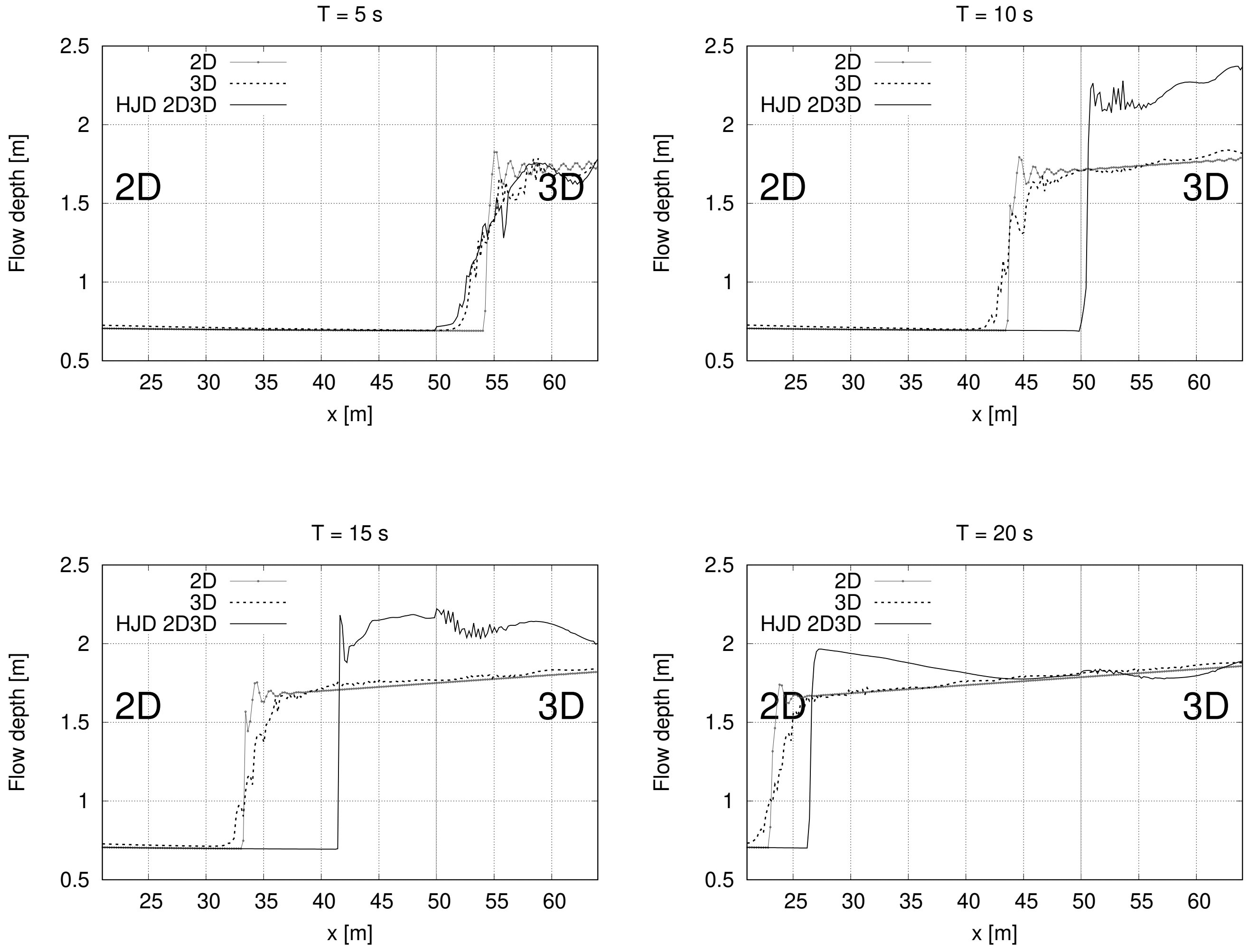


Figure 4.14: Surge travelling upstream at 5 - 20 [s] with 2D, 3D and `HJD 2D3D` simulations

4.4. Overlapping algorithm

This section describes my development of the improved coupling algorithm. The main reason for the back wave transfer failure is the lack of information at the interface from downstream. Communication between the regions was accomplished by interpolating the dependent variables at grid boundaries in previous implementations.

OpenFOAM provides an overset framework Ltd (2017-2019), which is a generic implementation of the Chimera method (Steger et al. (1983) and Benek et al. (1986)). This method was originally developed to treat moving objects by generating different geometrical features independently and coupling each component via interpolation. An overlapping mesh usually contains a dynamic object of interest superimposed on a static background mesh. To meet our needs, an improved coupling algorithm using a simplified overlapping mesh is developed here.

4.4.1. Cell-to-cell overlapping

In order to develop an overlapping coupling algorithm, overlapping domain decomposition methods are required to split the initial system into smaller subsystems, which are overlapped by more than the interface. The main feature that distinguishes the overlapping domain decomposition methods from the non-overlapping domain decomposition methods is that no special treatment needs to be made at the interface of the subsystems. Each subsystem is solved independently, and solutions between the subsystems are coordinated through an iterative procedure.

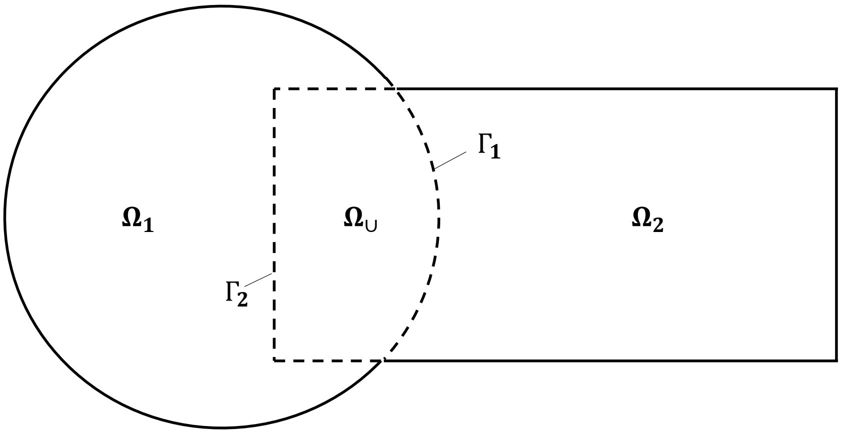


Figure 4.15: Typical overlapping sub-domains

The Schwarz algorithm is an iterative method for identifying the approximate solution for the entire domain. The typical structure of this method can be explained with the help of figure 4.15, where a union of a circle and a rectangle is divided into the sub-domain Ω_1 and the sub-domain Ω_2, sharing the overlapping region $\Omega_\cup$. The overlapping region $\Omega_\cup$ is enclosed by the internal interfaces Γ_1 and Γ_2, which are originally elements from the circle's and the rectangle's boundary, respectively. Overlapping regions are used to transfer data over the boundaries by means of interpolation schemes. For each time step, the iteration process for Ω_1 and Ω_2 are repeated until the solutions converge. The convergence of the solution at these interfaces ensures the convergence of the solution in the whole domain. During each iteration, the previous iteration's solution from Ω_2 on Γ_1 functions as a boundary condition for solving Ω_1. Ω_2 is then solved by imposing Γ_2 as a boundary condition, which is taken from the current iteration's solution from Ω_1. This boundary condition renewal is also called the Schwarz iteration.

After the aforementioned trial investigations in section 4.3, it was evident that interpolated data are not accurate enough to transmit the necessary information in certain extreme applications (*e.g.* hydraulic bore), and any improper value at the boundary face will lead to numerical instability. Imposing the downstream data on the interface will not yield definite information on the upstream field during the hydraulic bore simulation, for instance. To obviate interpolation for boundary data, the cell-to-cell overlapping technique provides a more robust connection between the regions because it incorporates the coupling method to update the downstream information in the upper part of the interface. In this manner, the hydraulic jump feature should be captured by the upstream region; and the upstream-travelling surge moves through the interface.

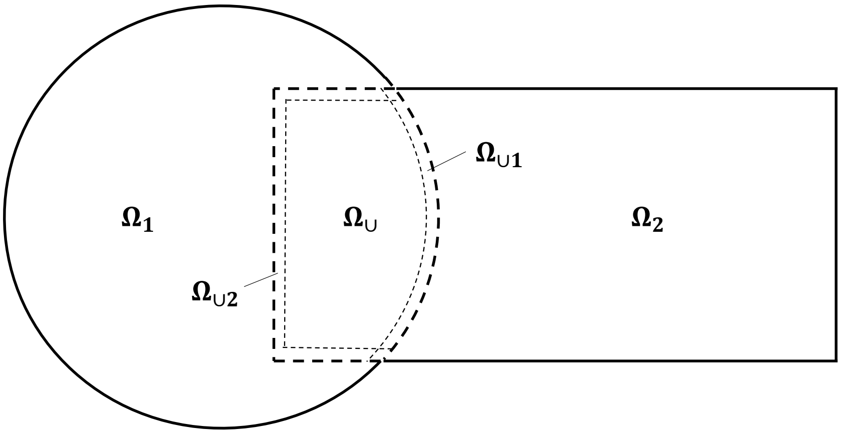
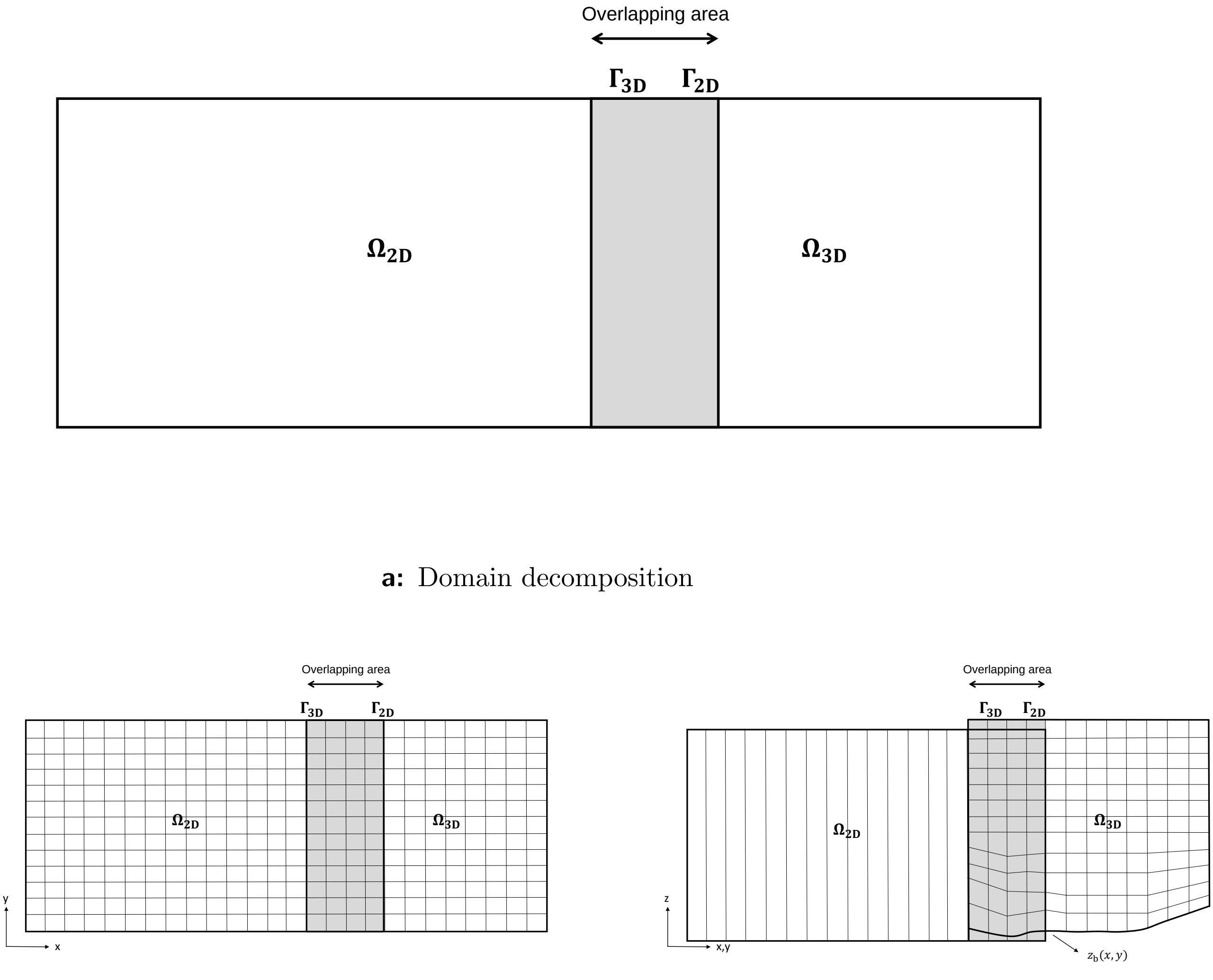


Figure 4.16: Cell-to-cell overlapping sub-domains

Typically, iterative procedure significantly slows down the simulation. The sub-solver in each sub-domain iterates on each time step and no interpolation is occurred during the data exchange in overlapping regions. To speed up the simulation, a simplified approach using only one Schwarz iteration and cell-to cell overlapping is applied here, allowing a direct exchange between the subsystems. Cell values from the internal interface's neighbour cells (*e.g.* $\Omega_{\cup 1}$ and $\Omega_{\cup 2}$ in figure 4.16) instead of face values on the internal interface (*e.g.* Γ_1 and Γ_2 in figure 4.16) will be updated during the Schwarz iteration. The solution procedure then becomes: (1) update values of $\Omega_{\cup 1}$ in Ω_1 with the previous time step values of $\Omega_{\cup 1}$ from Ω_2, (2) solve Ω_1 and extract the current time step values of $\Omega_{\cup 2}$, (3) update these values for Ω_2, (4) solve Ω_2 and extract the current time step values of $\Omega_{\cup 1}$ for the next time step.

4.4.2. Mesh representation

Grid-embedding does not require common boundaries but rather common regions between sub-domains. In this dissertation, no moving body is considered that enables perfect matching common regions and cell-to-cell mappings between sub-domains without any interpolation prerequisite. Due to the overlapping methodology, the mesh representation illustrated in figure 4.4 is not suitable because an overlapping mesh structure is necessary. The present coupling mesh shown in figure 4.17 overlaps a certain area (marked in grey) such that grids in the overlapping areas should match. More specifically, all grids next to the interfaces Γ_{2D} and Γ_{3D} must match in the xy-plane, though they do not have to match in the vertical direction. Accordingly, 2D and 3D meshes share the same geometry and the same grid spacing in horizontal directions inside the overlapping areas. With this approach, dependent data can be transferred directly between 2D and 3D cell columns.



a: Domain decomposition

b: Coupling mesh top view

c: Coupling mesh side view

Figure 4.17: Overlapping coupling mesh representation

However, the arbitrary grid shape causes considerable difficulties in ensuring a perfect match between 2D and 3D grids in the overlapping area. Quadrilateral grids in that area are then employed to simplify the mesh generation process.

4.4.3. Implementation in OpenFOAM

The overlapping method can be implemented in the coupled solver by imposing communicating variables in one region from the overlapping area onto the corresponding area in the other region. However, employing the OpenFOAM coupling framework, sub-domains are solved independently and communications between them occur at the interface. In other words, inner cells' data cannot be exchanged directly between sub-domains. In order to transfer the data cell-to-cell, interfaces play the role of a cell value receiver and transmitter. Accordingly, the Dirichlet-Dirichlet approach is applied here.

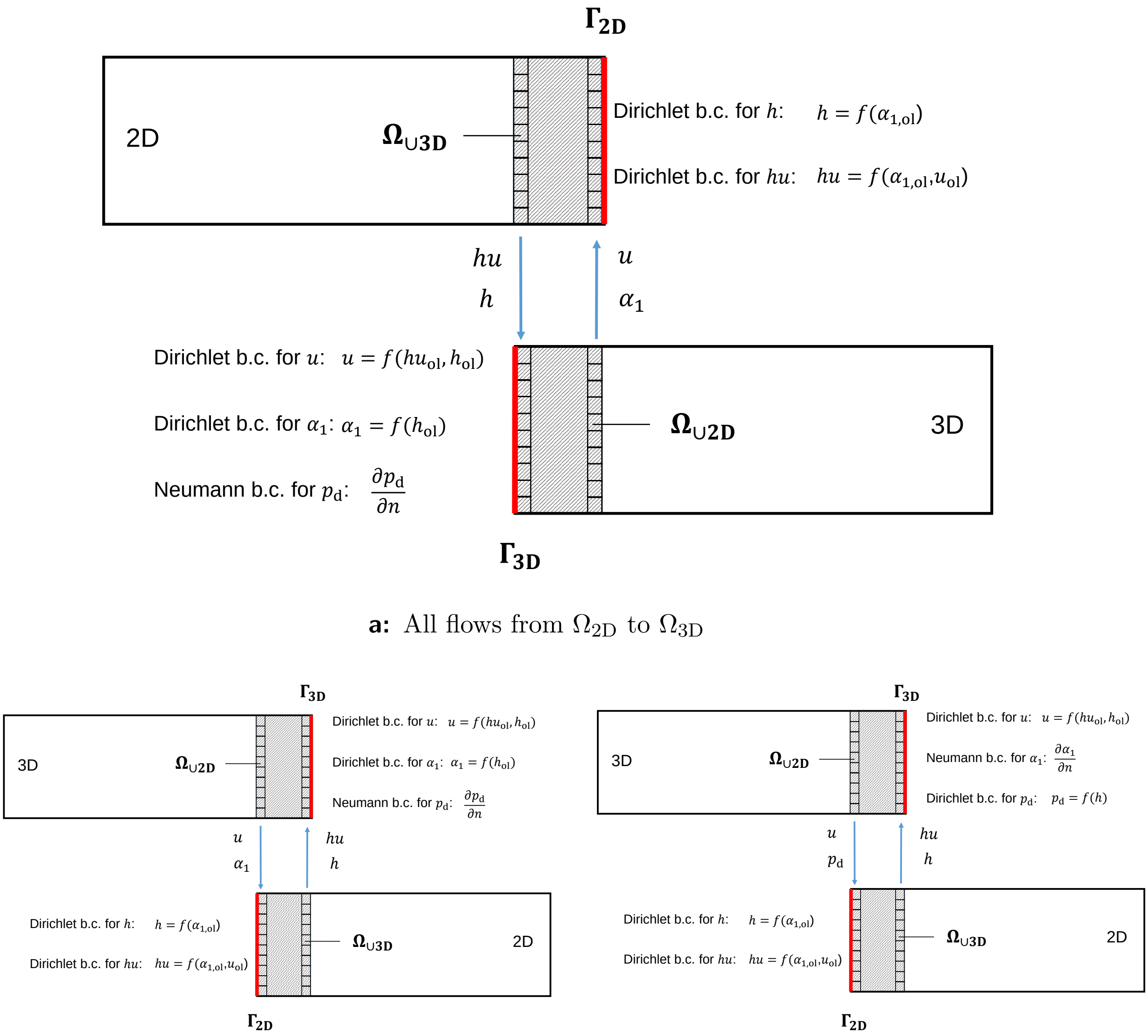
Nevertheless, a special treatment on the pressure term is taken into account in subcritical flow conditions. As introduced in chapter 3, a modified pressure term p_d instead of the total pressure p is used in the momentum equations of the 3D solver `interFoam` and the flow depth is governed by the hydrostatic pressure from downstream. Under the subcritical flow condition, a specified pressure should be prescribed at the outlet and the interface indicator α_1 will then automatically adjusts to the flow depth calculated from the pressure. Mintgen (2017) experienced that a fixed α_1 at the 3D outlet under such situation leads to an unbounded simulation. Consequently, the Dirichlet condition is activated for the pressure term p_d. This special treatment is also considered in my coupling implementation. In other words, the Dirichlet condition is set for the p_d and the Neumann condition is set for the α_1 if and only if subcritical flow conditions occur from Ω_3D to Ω_2D. Flow conditions and flow directions at the 3D interface Γ_3D are determined by equation (4.4.1):

$$Fr_\mathrm{if}^\mathrm{3D} = \frac{u_\mathrm{ol}^\mathrm{2D} \cdot \mathbf{n}}{\sqrt{g h_\mathrm{ol}^\mathrm{2D}}} \tag{4.4.1}$$

where the subscript $_\mathrm{ol}$ is the value in the overlapping area and $\mathbf{n}$ is the normal vector of the velocity. The $Fr_\mathrm{if}^\mathrm{3D}$ between 0 and 1 introduces a subcritical flow from Ω_3D to Ω_2D.

Figure 4.18 presents an overview of the overlapping approach in the coupled solver with three sub-cases. There is no difference in the exchange of variables between the supercritical and the subcritical flows when the flow goes from Ω_2D to Ω_3D. Flow quantities such as the flow depth h, the discharge hu, the VOF indicator α_1 and the flow velocity u are transferred via the Dirichlet boundary condition and are updated by the cell-to-cell overlapping, while the Neumann boundary condition is set for the pressure term p_d (figure 4.18 a). The same settings are implemented in the supercritical flow from Ω_3D to Ω_2D (figure 4.18 b). When a subcritical flow is simulated, the pressure p_d is used to update the flow depth h thus the Dirichlet boundary condition is specified for p_d (figure 4.18 c). Accordingly, the VOF indicator α_1 uses the Neumann boundary condition.

In technical terms, the overlapping cells for the data exchange should have the same number as the faces at the communicator interface. For example, in $\Omega_\mathrm{\cup 3D}$ each cell must have a corresponding face at Γ_3D; therefore, values in $\Omega_\mathrm{\cup 3D}$ can be generated correctly. The new


Γ_{2D}
2D
Ω_{U3D}
Dirichlet b.c. for h: $h = f(\alpha_{1,ol})$
Dirichlet b.c. for hu: $hu = f(\alpha_{1,ol}, u_{ol})$
hu
h
u
α_1
Dirichlet b.c. for u: $u = f(hu_{ol}, h_{ol})$
Dirichlet b.c. for α_1: $\alpha_1 = f(h_{ol})$
Neumann b.c. for p_d: $\frac{\partial p_d}{\partial n}$
Ω_{U2D}
3D
Γ_{3D}
Γ_{3D}
3D
Ω_{U2D}
Dirichlet b.c. for u: $u = f(hu_{ol}, h_{ol})$
Dirichlet b.c. for α_1: $\alpha_1 = f(h_{ol})$
Neumann b.c. for p_d: $\frac{\partial p_d}{\partial n}$
u
α_1
hu
h
Dirichlet b.c. for h: $h = f(\alpha_{1,ol})$
Dirichlet b.c. for hu: $hu = f(\alpha_{1,ol}, u_{ol})$
Ω_{U3D}
2D
Γ_{2D}
Γ_{3D}
3D
Ω_{U2D}
Dirichlet b.c. for u: $u = f(hu_{ol}, h_{ol})$
Neumann b.c. for α_1: $\frac{\partial \alpha_1}{\partial n}$
Dirichlet b.c. for p_d: $p_d = f(h)$
u
p_d
hu
h
Dirichlet b.c. for h: $h = f(\alpha_{1,ol})$
Dirichlet b.c. for hu: $hu = f(\alpha_{1,ol}, u_{ol})$
Ω_{U3D}
2D
Γ_{2D}

a: All flows from Ω_{2D} to Ω_{3D}

b: Supercritical flows from Ω_{3D} to Ω_{2D}

c: Subcritical flows from Ω_{3D} to Ω_{2D}

Figure 4.18: Transfer variables of all flow situations in the overlapping method coupling

coupling procedure for one time-step is:

- Extract Ω_{U2D} values from Ω_{3D} at time n-1
- Update Ω_{U2D} values to Γ_{2D} at time n
- Transfer Γ_{2D} values to their neighbouring inner cells in Ω_{2D} at time n
- Solve Ω_{2D} at time n
- Extract Ω_{U3D} values from Ω_{2D} at time n
- Update Ω_{U3D} values to Γ_{3D} at time n
- Transfer Γ_{3D} values to their neighbouring inner cells in Ω_{3D} at time n
- Solve Ω_{3D} at time n

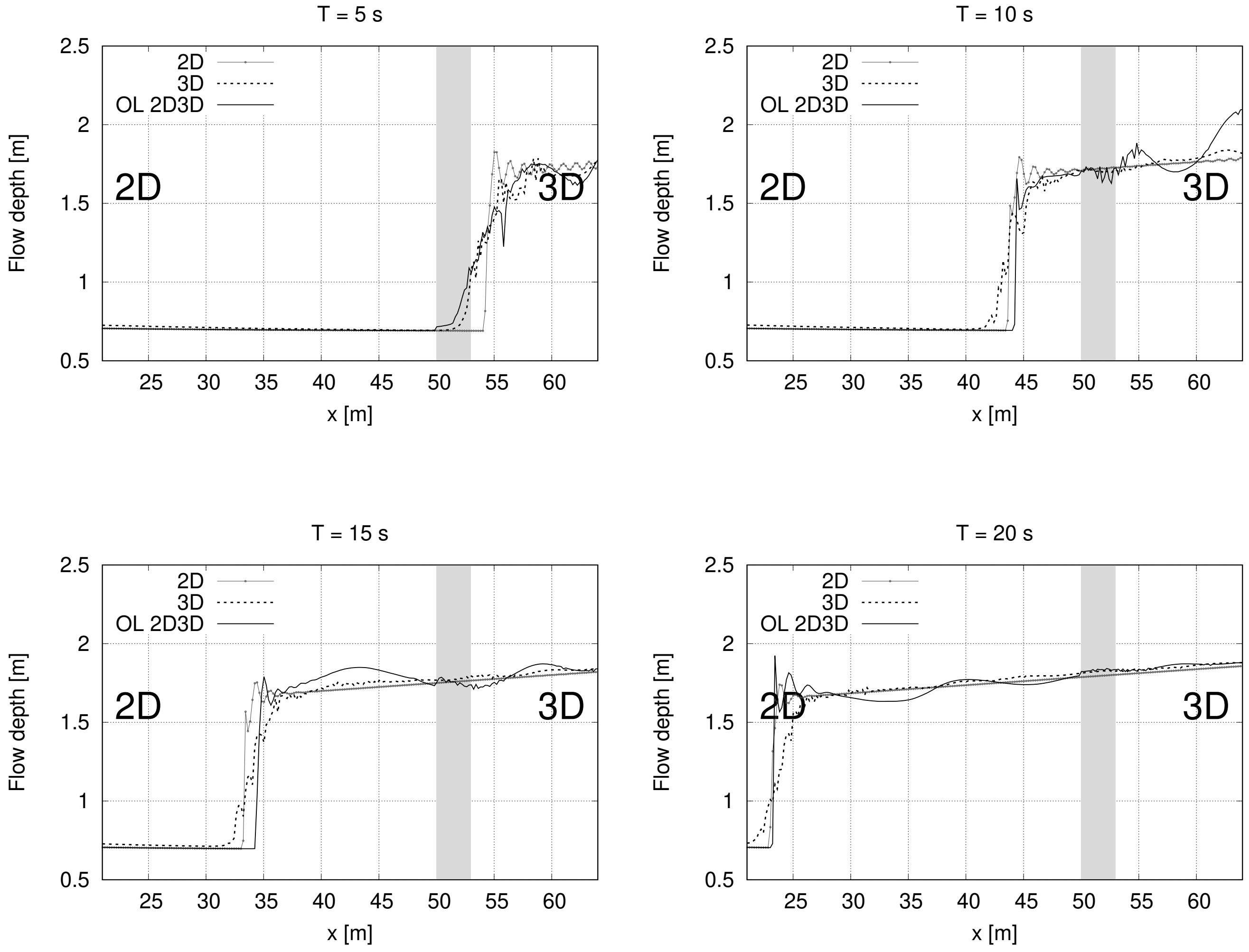


Figure 4.19: Surge travelling upstream at 5 - 20 [s] with 2D, 3D and `OL 2D3D` simulations

The overlapping method is examined by using the same test case. Results from the overlapping method are denoted as `OL 2D3D` and the overlapping distance is 3 [m]. Comparing the surge travelling results of the coupling method (figure 4.19) with previous results from the other coupling methods (figures 4.7, 4.9, 4.11 and 4.14) demonstrates an improved surge transfer behaviour from Ω_{3D} to Ω_{2D}. Before the wave approaches the interface (*e.g.* T = 5 [s]), flow depths from all coupled solvers fit well. After the wave touches the interface, the flows from `CB 2D3D`, `DBC 2D3D`, `DWC 2D3D` and `HJD 2D3D` are blocked at the interface and deflect back to the downstream, while the flow in the `OL 2D3D` passes the interface and advances upstream. The overlapping method can handle the upstream-travelling wave problem moving from a high dimension to a low dimension.

4.5. Summary

A series of 2D-3D coupling algorithms based on OpenFOAM is presented in this chapter. The main framework is referred to the characteristics-based coupling algorithm (Mintgen (2017)). To overcome the hydraulic jump blocking failure of the characteristics-based coupling algorithm, some ad-hoc modifications of the algorithm have been tested and the resulting new algorithm, the overlapping coupling algorithm, is formulated. The new 2D-3D solver is named as the `shallowInterFoamOL`. This solves the hydraulic jump blocking problem algorithm.

5. Validation tests

Wave magnitudes and propagation velocities are of great significance in free surface flows. This chapter investigates the properties of the numerical scheme in terms of wave formation and propagation in rectangular open channel flows. To analyse the method's ability, a sequence of test cases with simple geometries will be introduced as a validation of the proposed coupling method.

This chapter is organized as follows. The motivation of all test cases will be given in the first section as well as an overview of relevant parameters. Sections 5.2, 5.3 and 5.4 introduce validation cases based on different flow conditions and configurations so that distinct wave motions are originated. The conclusions are drawn in section 5.5.

5.1. Introduction

Different types of water waves are categorized based on their formations and behaviors, among which (1) breaking waves and (2) shallow water waves are two popular topics in open channel flow simulations. Three scenarios are therefore established upon these two types.

The former type (1) of waves occur when their amplitude reach a critical level and the huge swells collapse on top of themselves. No wave crest are associated with them. The process of wave breaking causes large amount of turbulent kinetic energy that 3D models are preferred to solve breaking wave problems accurately. One kind of the typical breaking wave, the upstream-travelling surge, cannot be solved correctly by the coupling method proposed by Mintgen (2017). The upstream-travelling surge is produced when a plenty of supercritical flows change to subcritical flows and travel backwards at high speeds. The flow condition change from supercritical to subcritical over the coupling region was failed to determine by the characteristics-based coupling method. The main challenge of the new improved coupling method is therefore to tackle the sudden change of flow conditions. Section 5.2 studies the method's capacity for this upstream-travelling surge.

The later water wave type (2) of interest, the shallow water wave, has a considerable large applications in both research and engineering filed. It is defined that the flow depth is 1/20 lesser than the wavelength and travels at the free surface of a fluid under the force of gravity. Within the scope of validating the coupling method, a solitary shallow water wave is selected to analyse the stability and accuracy of the coupling method. The solitary wave is localized gravity wave having finite amplitude and propagate with constant speed and wave shape (Hereman (2009)). Two sub-cases will be discussed about the solitary wave: the downstream-travelling wave in supercritical flows in section 5.3 and the upstream-travelling wave in subcritical flows in section 5.4.

Each scenario is simulated by three kinds of models: (1) the two-dimensional shallow water equations, (2) the three-dimensional RANS equations and (3) the 2D-3D coupling method. The methods are marked in the following contents and figures: the proposed overlapping coupling method is marked as the `OL 2D3D`; the 2D and the 3D methods use `2D` and `3D` respectively as abbreviations. To simplify the test cases, channel widths in all tests are ignored due to the fact that the flow dynamic in the span-wise direction is of less interest in wave transfer modelling here. Table 5.1 summarises basic parameters for three cases.

	Case 1	Case 2	Case 3
I_s [-]	0.005	0.005	0.001
k_st [$\mathrm{m}^{1/3} \cdot \mathrm{s}^{-1}$]	59	73	52.3
k_s [m]	0.001	0.001	0.008
q [$m^2 \cdot$ s]	2.23	3.886	2.49
h [m]	0.77	0.72	1.1
Solitary wave center in X-direction [m]	-	40	30
Fr before the wave [-]	1.19	2	0.72
grid resolution [m]	0.05	0.1	0.05
Overlapping region in X-direction [m]	50 - 53	50 - 53	20 - 23

Table 5.1: Parameters of test cases

where

- Case 1: Upstream-travelling surges over a supercritical flow in section 5.2
- Case 2: Downstream-travelling wave over a supercritical flow in section 5.3
- Case 3: Upstream-travelling wave over a subcritical flow in section 5.4

and where I_s is the channel slope, k_st [$\mathrm{m}^{1/3} \cdot \mathrm{s}^{-1}$] is the Strickler coefficient, k_s is the sand roughness, q [$\mathrm{m}^2 \cdot \mathrm{s}^{-1}$] is the specific discharge, h [m] is the flow depth and Fr is the Froude number.

5.2. Upstream-travelling surges over a supercritical flow

Upstream-travelling surges or shock waves show fierce behaviors over an incoming supercritical flow, which failed to be modelled by Mintgen (2017). Such shock wave starts with a sudden strong swelling in the free surface level, then the sharp surge front rolls and submerges the incoming flow. In this section, a series of surge propagation simulations based on the steady supercritical flow are carried out to present the `OL 2D3D` performance. This test case was also selected as the benchmark case by the characteristics-based coupling method and other ad-hoc modification coupling methods described in chapter 4.

5.2.1. General description

A general set-up of the study case is displayed in figure 5.1, where the red column implies the coupling interface in the non-overlapping coupling methods introduced in chapter 4 or the coupling overlapping area in the overlapping method. Each test case consists of two sub-simulations: the steady state flow development in the channel and the backward surge propagation. To develop the supercritical steady state flow, a certain amount of flow goes through the steep slope channel from the left to the right without any control operation until the water level stabilises. The incoming supercritical flow stems from a specific flow rate of 2.23 [$\mathrm{m}^2 \cdot \mathrm{s}^{-1}$] and a flow depth of 0.77 [m] at the channel inlet section resulting in the Froude number Fr of 1.19 before the surge is generated. On this basis, a closed outlet in an instant, for example a fully closed hydraulic gate or a high enough obstacle, engenders the surge over the incoming supercritical flow which travels upstream.

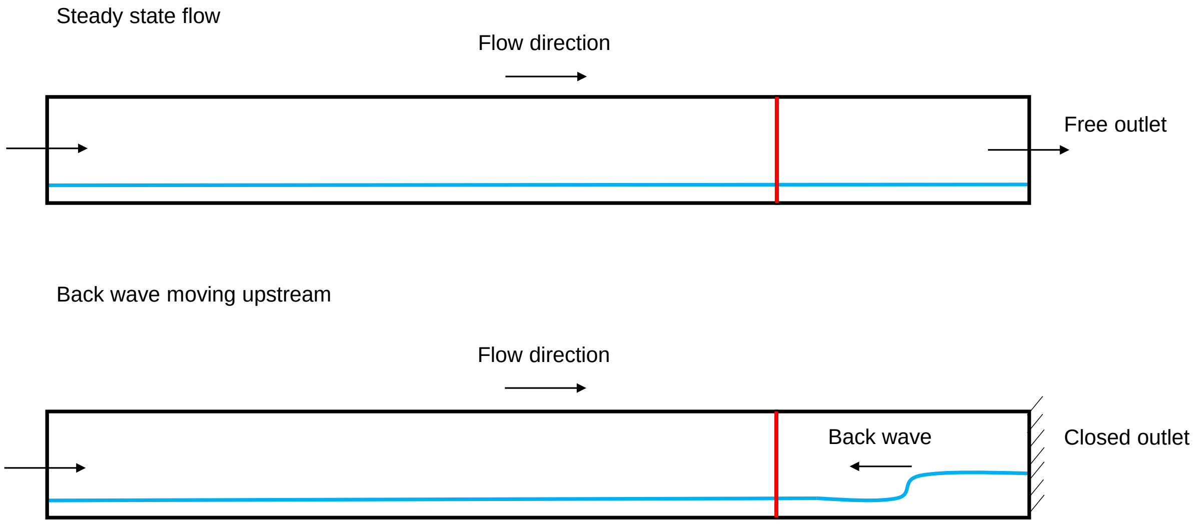


Figure 5.1: Surge transfer simulations set-up

5.2.2. Numerical set-up

The prospective channel should be long enough to observe the wave travelling process and the side walls should be high enough to cover the shock wave amplitude. Moreover, a slightly steep channel slope is applied, which, on the one hand, helps in supercritical flow generation; on the other hand, the resultant upstream-travelling surge is able to transfer further towards upstream. As a consequence, a rectangular open channel with 65 [m] length, 4 [m] height and 0.005 slope is considered to represent the geometry system. Since the channel width is not relevant to our interests, a single cell is specified in the y-direction and the patches perpendicular to the y-axis are assigned with `empty` boundary type. As mentioned in section 4.1, the 2D OpenFOAM mesh requires only one cell in the vertical direction, *i.e.* z-direction, thus the 2D cell height is prescribed to be the same height as the geometry system (Δ z = 4 [m]). Referring to Mintgen (2017), a uniform grid spacing of 0.05 [m] produced accurate results in such scenarios and is therefore adopted in each direction.

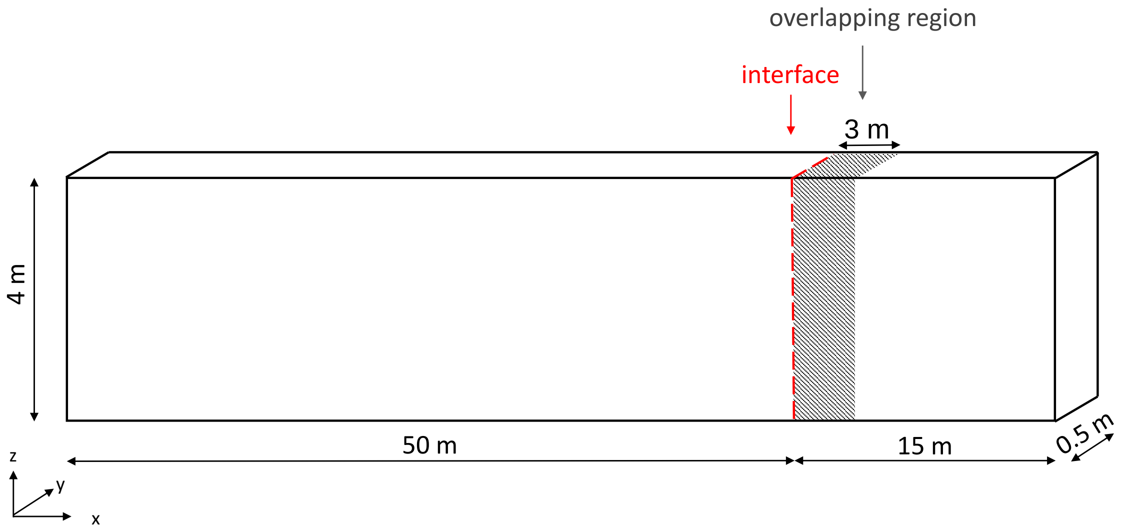


Figure 5.2: Side view of the computational domain

As sketched in figure 5.2, the 2D-3D interface (marked in red dotted line) locates 15 [m] upstream from the outlet for the characteristics-based and other ad-hoc modification coupling methods, and a 3 [m] overlapping distance (*i.e.* from 50 to 53 [m]) is used for the `OL 2D3D`. The upstream region starts from 0 to 53 [m] and the downstream region covers from 50 to 65 [m].

The `OL 2D3D` numerical cases are validated in two types of coupling connections: (1) the Ω_{2D} situated upstream of the Ω_{3D} denoted as `2D->3D` and (2) the other way around denoted as `3D->2D` connection. As introduced in section 4.1, different variables are solved in the pure 2D and 3D solvers, resulting in different initial and boundary condition set-ups. The coupling initial and boundary set-ups are derived from the intersections of the set-ups in the matched non-coupled models.

To make it clearer, flow chart of the inlet and the outlet boundary conditions is additionally detailed in figure 5.3 as an example to illustrate the coupled model set-up. We use blue and green outlines to represent the pure 2D and 3D boundary conditions respectively. Each sub-region of coupled models adopts its corresponding non-coupled boundary conditions indicated by the same color, and the coupling region marked in red merges two sub-regions. Consequently, three categories of boundary conditions are introduced in coupled models: the customized coupling boundary conditions (in red), the 2D boundary conditions (in blue) and the 3D boundary conditions (in green).

Assuming the flow direction is from the left to the right, where the left boundary is the inlet and the right boundary is the outlet. Four group of variables are then assigned to the target boundary, which are variables at the 2D inlet (ϕ_{in}^{2D}), the 2D outlet (ϕ_{out}^{2D}), the 3D inlet (ϕ_{in}^{3D})

and the 3D outlet ($\phi_{\mathrm{out}}^{3\mathrm{D}}$). When an upstream $\Omega_{2\mathrm{D}}$ encounters a downstream $\Omega_{3\mathrm{D}}$, $\phi_{\mathrm{in}}^{2\mathrm{D}}$ and $\phi_{\mathrm{out}}^{3\mathrm{D}}$ are therefore adopted for such a `2D->3D` connection. Vice versa, the `3D->2D` connection prescribes $\phi_{\mathrm{in}}^{3\mathrm{D}}$ and $\phi_{\mathrm{out}}^{2\mathrm{D}}$ onto its boundary patches.

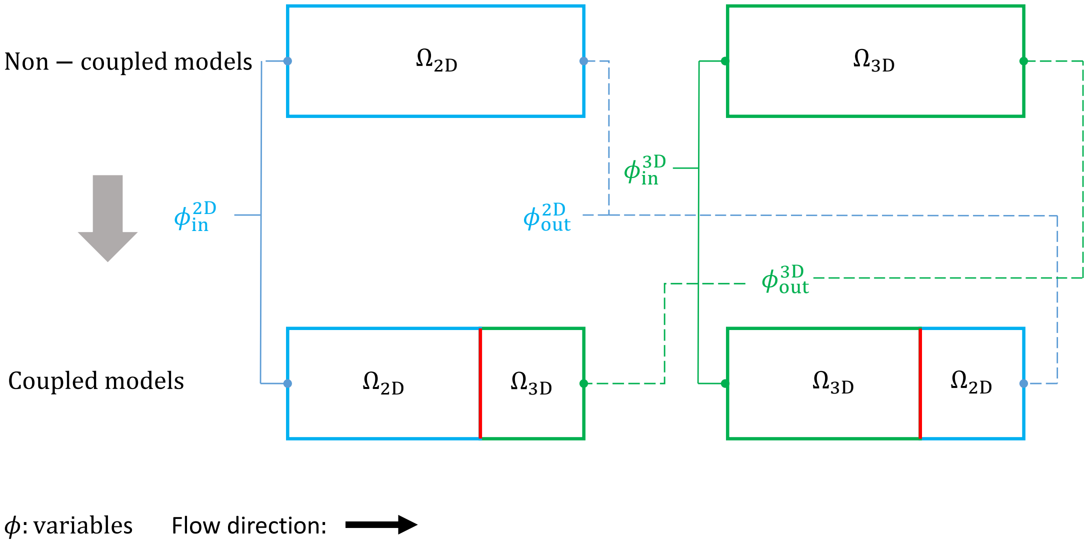


Figure 5.3: General boundary conditions set-up for coupled models

Steady state development

In order to minimize disturbances in the flow, a fully developed flow is pre-generated. In practice, a fully developed 3D flow can be generated at a certain distance downstream of the inlet, known as the establishment length (L_{e}). An estimation of L_{e} in an open channel varies from 50 to 150 times of the flow depth (H. Bonakdari (2014)), resulting in a maximum L_{e} of 115.5 [m] in this case. This L_{e} is fairly long that increases the computational costs. To get a fully developed flow for surges generation, a steady state flow is initially implemented before the surge transfer. In supercritical flows all flow information travels in a stream-wise direction, prescribed flow properties are defined at the inlet whereas Neumann conditions are employed on the outlet. To speed up the simulation, a wet bed condition with 0.1 [m] water depth is predefined initially.

The numerical tests performed in this chapter act as laboratory tests and no control experiment data is available for parameters calibration. 3D simulations are therefore used as the reference to adjust 2D parameters (mainly for the friction terms). Followed by the schematic diagram of boundary conditions set-up in figure 5.3, the 2D and 3D set-ups are firstly determined and subsequently, the coupled set-up can be simply achieved by patching the corresponding initial and boundary conditions in the $\Omega_{2\mathrm{D}}$ and the $\Omega_{3\mathrm{D}}$. Variables set-up for different regions are described below.

3D region set-up 3D inlet is delicately set up based on equations presented in the previous chapter of section 4.1. The inlet volume fraction indicator distribution is prescribed using equation (5.2.1) by inserting the inlet flow depth of 0.77 [m] and the half cell height which is 0.025 [m]. z_c is the variable of the face center in the vertical direction along the 3D inlet patch.

$$\alpha_1^{3\mathrm{D}}(z_\mathrm{c}) = \begin{cases} 0 & 0.77 < z_\mathrm{c} - 0.025 \\ 1 & 0.77 > z_\mathrm{c} + 0.025 \\ \frac{0.77 - z_\mathrm{c}}{0.05} + 0.5 & \quad other \end{cases} \tag{5.2.1}$$

Theoretically, a logarithmic velocity profile fits better in the 3D domain than a uniform mean velocity profile. According to equation (4.1.7), the friction velocity is necessary but challenging to properly define at the $\Omega_{3\mathrm{D}}$ inlet. With this respect, a uniform mean velocity profile of 2.896 [$\mathrm{m}^3 \cdot \mathrm{s}^{-1}$] is employed at the $\Omega_{3\mathrm{D}}$ inlet patch.

The channel bottom patch utilizes a non-slip boundary condition and the $k - \omega\ SST$ turbulent model. For the channel roughness, the sand grain roughness $k_\mathrm{s} = 0.001$ [m] is chosen. The pressure term uses a buoyant pressure boundary type which is suitable for the hydrostatic pressure contribution. It sets a fixed-gradient pressure based on the atmospheric pressure gradient.

For other boundary patches setting, free boundary conditions are set to develop the supercritical flow, hence the zero-gradient condition is specified in all variables.

2D region set-up 2D region set-up is rather simple that a fixed value for the flow depth 0.77 [m] and for the flow rate 2.23 [$\mathrm{m}^3 \cdot \mathrm{s}^{-1}$] are explicitly imposed at the $\Omega_{2\mathrm{D}}$ inlet patch. After calibration with 3D results, the Strickler value is estimated as 59 [$\mathrm{m}^{1/3} \cdot \mathrm{s}^{-1}$] for the bottom friction modelling. At the $\Omega_{2\mathrm{D}}$ outlet, the zero-gradient is applied in all variables in the steady flow generation case.

Surge generation

To produce the upstream-travelling surge, the outlet of the channel is closed suddenly. In other words, when the supercritical flow reaches the steady state, a fully closed outlet boundary condition is set by imposing a fixed zero velocity at the outlet of the $\Omega_{3\mathrm{D}}$ and a fixed zero flow rate at the $\Omega_{2\mathrm{D}}$ outlet. Wall functions for turbulent parameters also apply to the closed 3D outlet. The simulation time is reset from this point.

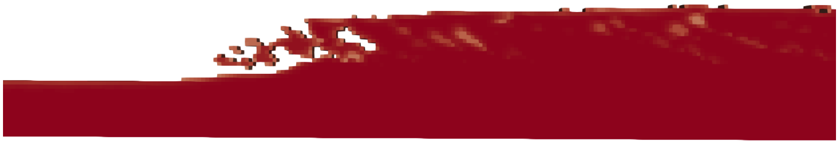


Figure 5.4: Surge profile at 20 [s]

A screen shot obtained from the 3D results at 20 [s] is shown in figure 5.4 as an example illustrating the upstream-travelling surge with air bubbles. More results with all models will be detailed in the following.

5.2.3. Surge transfer from Ω_{3D} to Ω_{2D}

In order to test the capability of the overlapping algorithm in surge transfer from the Ω_{3D} to the Ω_{2D}, the flow direction is therefore from the Ω_{2D} to the Ω_{3D} leading to a 2D->3D coupling connection. The overlapping region located at x = 50 - 53 [m] splits the whole domain into the upstream part (*i.e.* the Ω_{2D}: x = 0 - 53 [m]) and the downstream part (*i.e.* the Ω_{3D}: x = 50 - 65 [m]). Under such configuration, the surge is produced at the outlet in the Ω_{3D} and moves upstream. Concerning our interests of the study, the whole process of the OL 2D3D can be categorized into four phases: (1) surge generation in the Ω_{3D}; (2) surge movement in the Ω_{3D}; (3) surge transfer from the Ω_{3D} to the Ω_{2D} and (4) surge travelling in the Ω_{2D}.

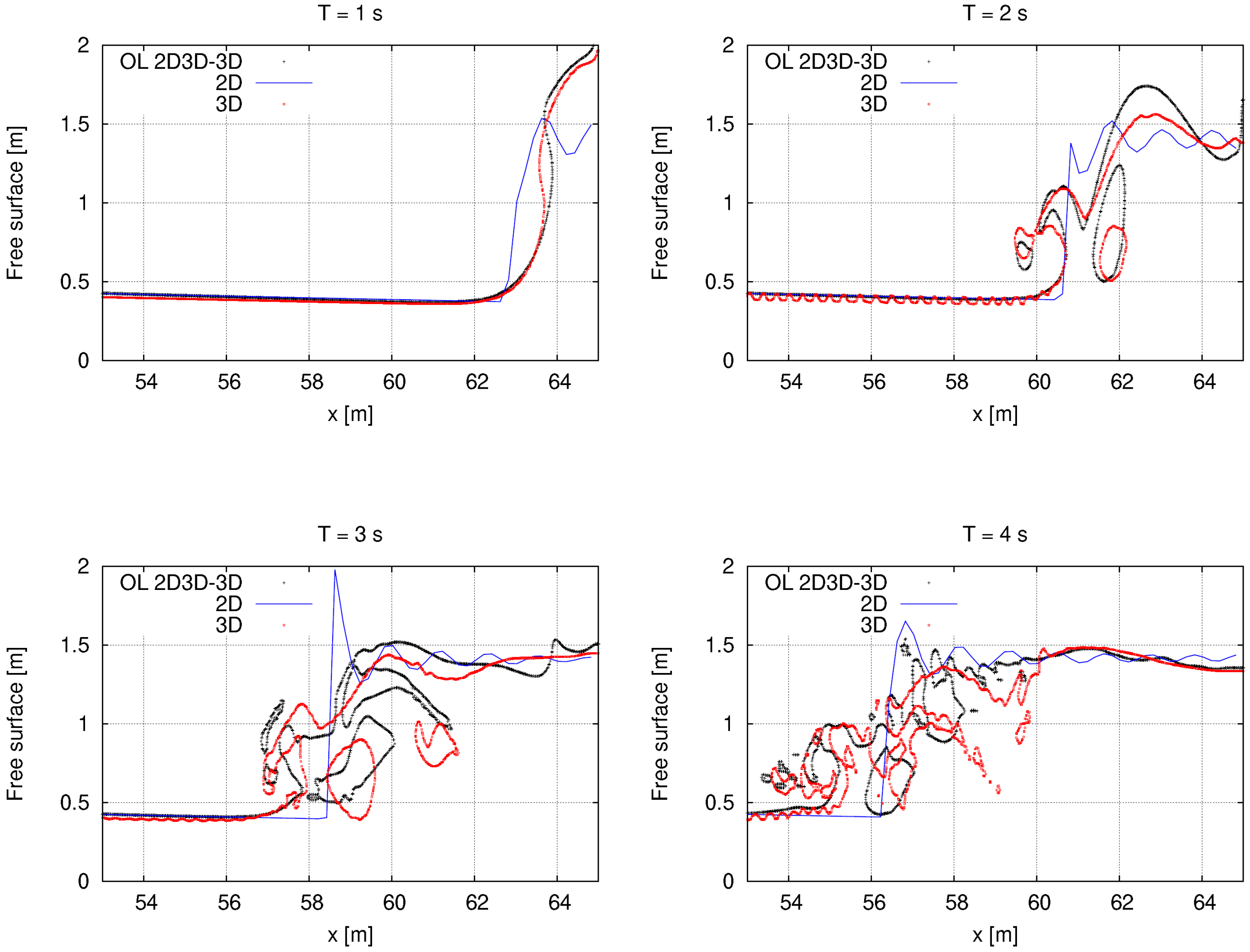


Figure 5.5: Surge travelling in the Ω_{3D} at 1 - 4 [s]

Figure 5.5 reproduces phases (1) and (2) in the Ω_{3D}, where the free surface behaviors from the three models are investigated in the downstream region. During the first 4 [s] the `OL 2D3D` back wave stays in the Ω_{3D}, results from the 3D sub-solver of the `OL 2D3D`, denoted as `OL 2D3D-3D`, are compared with the `2D` and the `3D` results. As a consequence of the totally blocked outlet, the incoming water was stopped by violence at the downstream boundary. Due to the sudden deceleration in flow velocities, a sharp rise in the water elevation is observed at T = 1 [s] associated with an energy transfer into potential energy. The pounding water body swiftly topples over and rolls away from the blocked outlet. The hydraulic jump roll is subsequently formed, characterised by some complicated turbulent air-water flow features. During T = 2 - 4 [s], the surge is thus generated and propagates upstream. These turbulent jumps are well represented by the `OL 2D3D-3D` results compared to the pure `3D`, including the splashing spray and some intense entrained air bubbles at the roller and the jump toe. These entrained air bubbles are broken up into smaller bubbles by the turbulent shear and tend to disappear eventually (Brocchini et al. (2002)). Such features of wave discontinuity are not able to be captured by the `2D` solver, resulting in a steeper wave front and a wavy surface.

However, the appearance of wiggles in the `3D` free surface is observed at T = [2]. This impact is usually arisen from simulation of the non-linear wave propagation using the `interFoam`. As introduced in chapter 2 section 3.4.2, `interFoam` used the VOF model to represent the two-phase flow interface and the VOF model is solved by the currently implemented MULES scheme (a numerical interface compression method). A comprehensive investigation of the `interFoam` performance on progressive waves simulations has been studied by B. E. Larsen (2019), which stated that the wiggles are caused by the numerical interface compression method (MULES). These wiggles decrease with higher spatial and temporal resolutions, which requires a much more computational expense. Applying the more diffusive schemes also help in mitigating the undesirable effects, nevertheless, the more diffusive schemes are more like to smear the interface leading to more vague results. Recently, Roenby et al. (2018) have developed a new algorithm (isoAdvector) coupled in the `interFoam` to remedy the aforementioned effects. Considering the relatively small wiggles in this case and the core attention in surge transfer, these wiggles will not be deeply discussed in this work.

In 'phase (3)' the surge travels through the overlapping region which is exhibited in figures 5.6 to 5.9. The 2D sub-solver results are included from now on since the surge enters the Ω_{2D}, denoted as `OL 2D3D-2D` at X = 53 [m]. To better understand the `OL 2D3D` performance, results from the two sub-solvers of the `OL 2D3D` are both displayed in the overlapping region which is highlighted in grey. Overall, the results from the three models are in general agreement. The `OL 2D3D-3D` and the `3D` surges are well matched before they leave the overlapping region (*e.g.* at T = 5 and 6 [s]). Because of the steeper `2D` wave front, its jump toe slightly lags behind the `3D` jump toe so that the `3D` wave information passes the overlapping region earlier (*e.g.* at T = 7 [s]). In reference to the coupling algorithm, the right overlapping interface receives the `3D` information, in the meantime the left overlapping interface receives the `2D` information. Caused by the lag in the `OL 2D3D-2D` wave front, no `2D` wave information is detected at the left overlapping interface at T = 7 [s]. On the contrary, the incoming supercritical flow information is assigned to the left overlapping interface as the inlet input of the `OL 2D3D-3D`, which results the reflected wave travelling downstream. The reflected wave grows and induces an objectionable surface in the overlapping region (*e.g.* at T = 8 [s]).

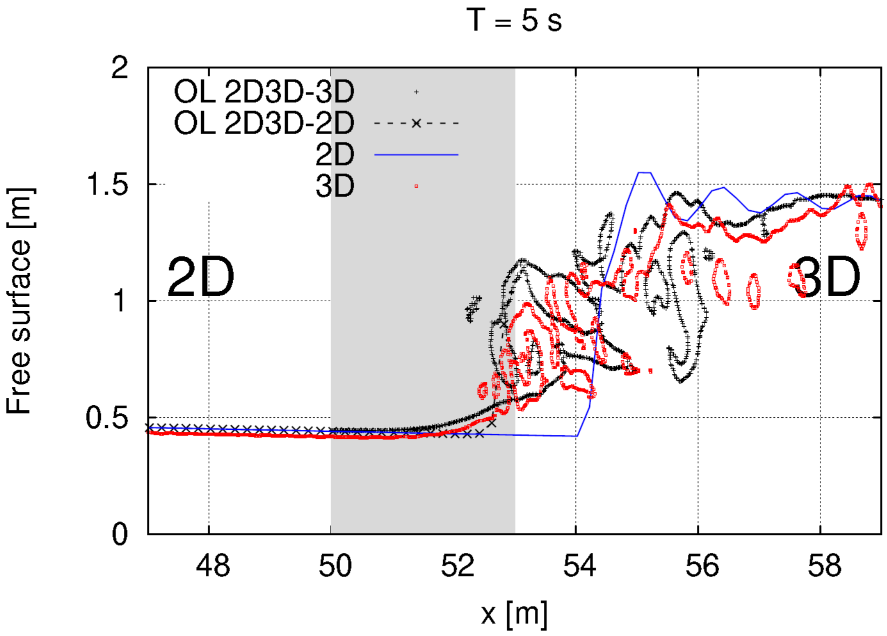


Figure 5.6: Surge transfer from the Ω_{3D} to the Ω_{2D} at 5 [s]

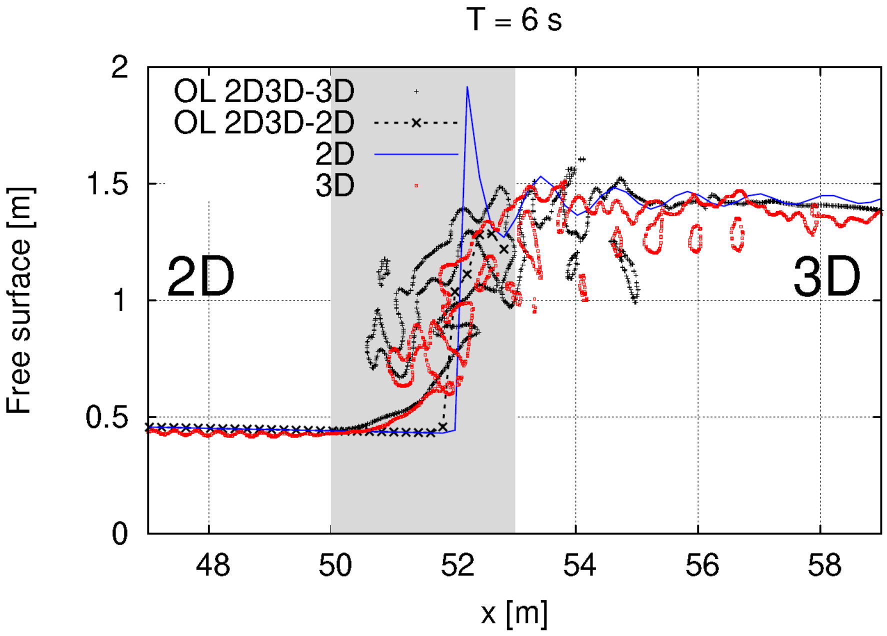


Figure 5.7: Surge transfer from the Ω_{3D} to the Ω_{2D} at 6 [s]

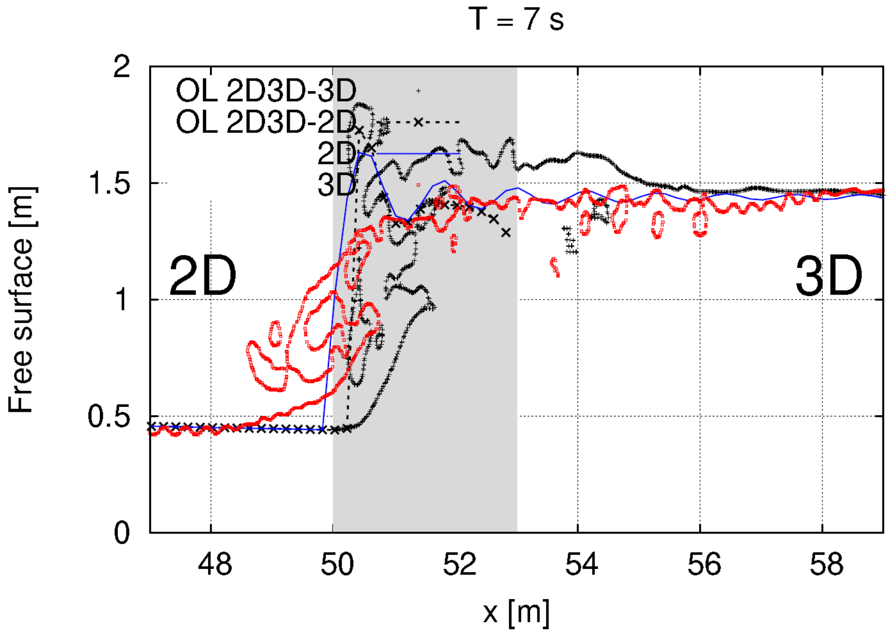


Figure 5.8: Surge transfer from the Ω_{3D} to the Ω_{2D} at 7 [s]

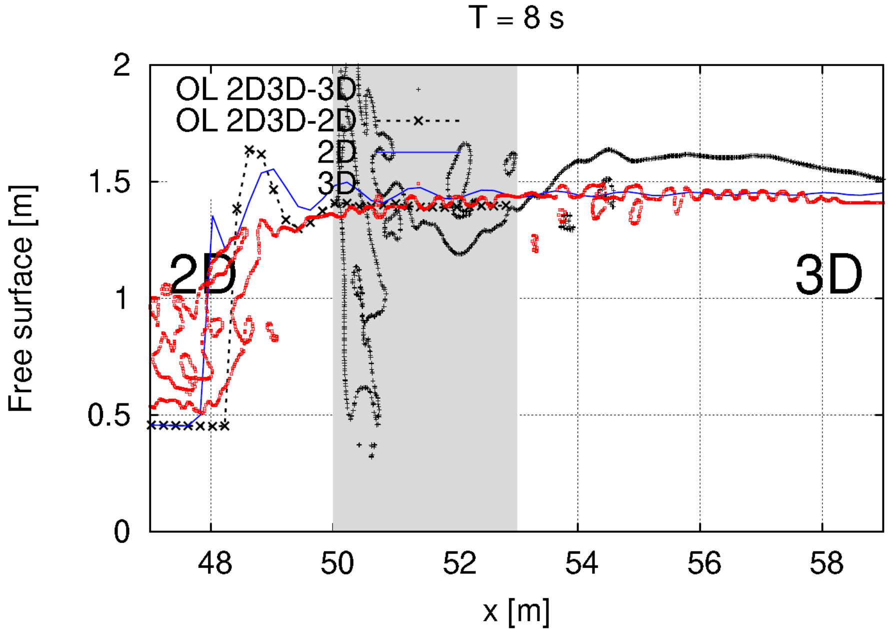


Figure 5.9: Surge transfer from the Ω_{3D} to the Ω_{2D} at 8 [s]

However, as the surge propagates far from the overlapping region, the aforementioned violent disturbances neutralizes shown in figure 5.10, which can be interpreted by the on going information updating in the 2D-3D coupling process. Meanwhile wave front features completely cross the overlapping region by all the three models, the upstream advancement wave is dominated by downstream quantities. In the phase (4), when the wave front is totally placed in the Ω_{2D}, a good match of the wave front can be noticed between the `2D` and the `OL 2D3D` at T = 20 [s].

Figures 5.5 to 5.10 demonstrate the surge transfer proceeding along the open channel. Before the `OL 2D3D` surge accesses the overlapping region and only stays in the Ω_{3D}, its performance lies in the consistency with the 3D flow. After it runs through the overlapping region to the Ω_{2D}, a 2D free surface flow manner is expressed. Surge transfer through the overlapping region reveals wave disturbances between two interfaces. In consequence, wave transfer crossing and after crossing the overlapping field are additionally inspected in figure 5.11.

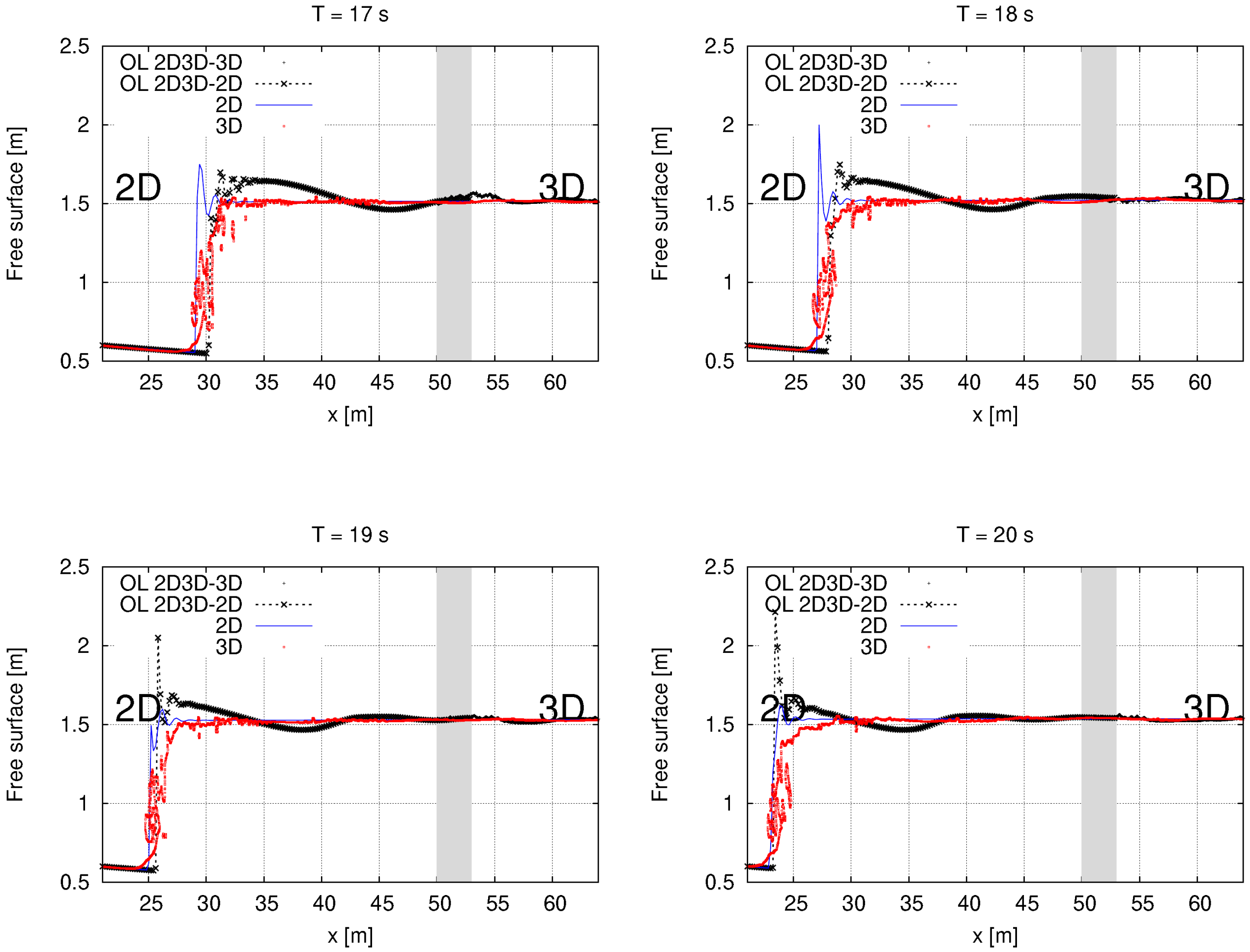


Figure 5.10: Surge travelling in the Ω_{3D} at 17 - 20 [s]

To assess this process via a more comparable wave shape, the spatial averaged flow depth rather than the free surface is visualized. The averaged flow depth is calculated by integrating values of α_1 over each column in the Ω_{3D}. At T = 6.5 [s], the `3D` wave front first partially passes the overlapping region, while the `OL 2D3D` and the `2D` wave fronts are entirely situated inside the overlapping zone. In other words, no discharge and flow depth change are sensed

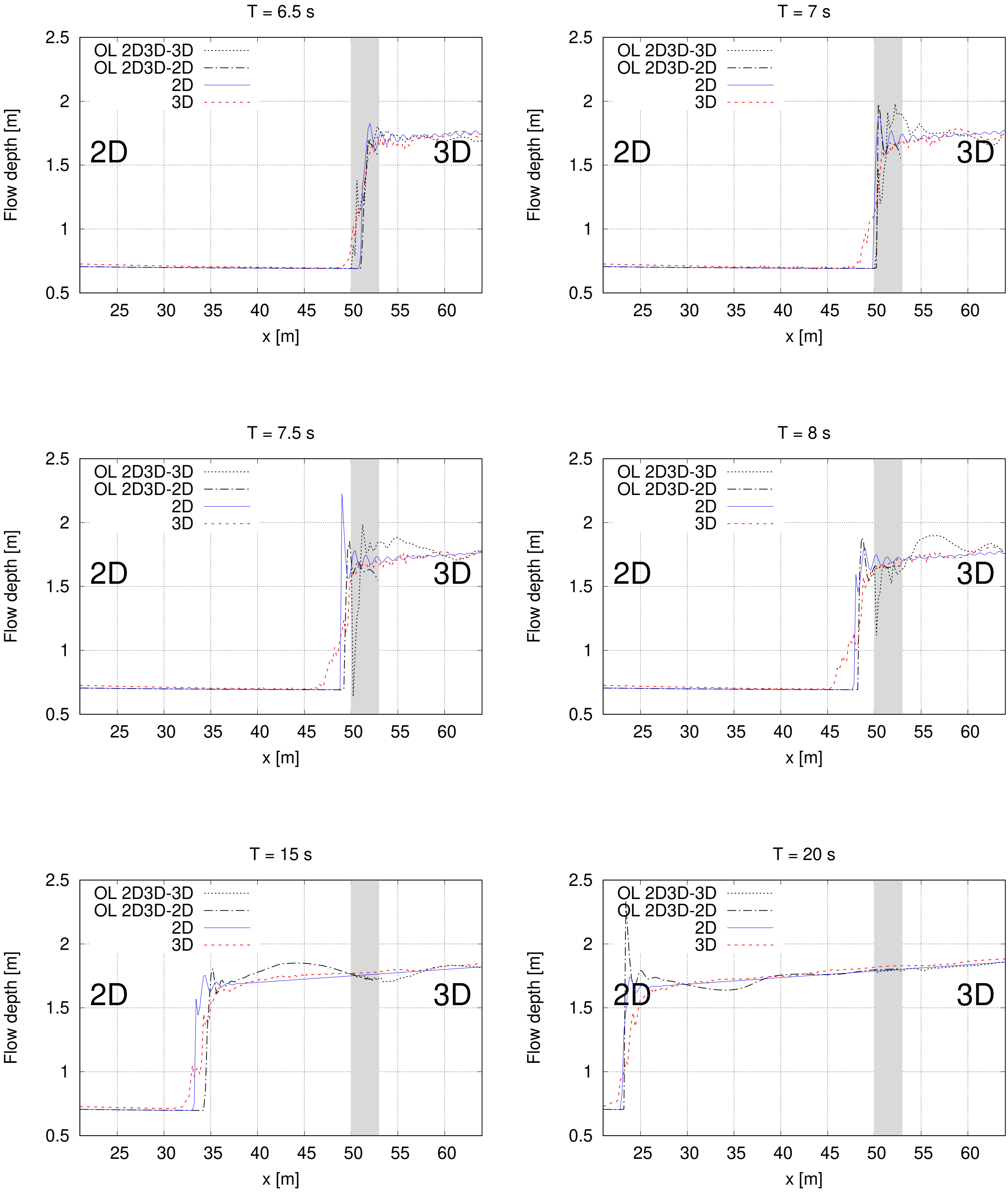


Figure 5.11: Surge travelling at 6.5 - 20 [s] in term of the flow depth

at the `OL 2D3D-3D` inlet patch but a decreased discharge and an increased flow depth are prescribed at the `OL 2D3D-2D` downstream boundary. A reflected wave is sequentially generated at T = 7 [s] as a consequence of the discharge gap imposed on both overlapping interfaces. This resulting reflected wave prevents the wave from advancing upstream, which is thus behind the pure `2D` wave front. Such reflected wave also conflicts with the approaching surge from the downstream in the Ω_{3D} and produces a prominent fluctuating wave surfaces at T = 7.5 [s]. Impacts from the non-matching information at the overlapping interfaces during the surge transfer phase mitigate as time goes on. Results of later time stamps shown in figure 5.11 indicate a better agreement with aspects mentioned above. Once the full wave front enters the Ω_{2D}, the steepening of the `OL 2D3D` wave front is inherited from the `2D` feature and stays the same. The wave arriving lag becomes smaller and finally vanishes at T = 20 [s]. The amplitude of unexpected disturbances also reduces and the wave surface in average reaches a satisfying match among the three models.

5.2.4. Sensitive study

To enhance the understanding of the relationships between input variables and output results of a model, uncertainty and sensitive study is a critical step in systematic reviews. Within the help of the study, the robustness of a model and the impact of a variable can be evaluated. One-at-a-time is one of the most common approaches to examine how this uncertainty factors affect the results by changing one factor meanwhile keeping other variables unchanged.

Input of a numerical model can be categorized as (1) mesh parameters (*e.g.* mesh resolution, overlapping configurations), (2) physical properties (*e.g.* flow properties), (3) numerical settings (*e.g.* time step, simulation time and numerical schemes). (1) The coupling position and the overlapping distance can be considered as two major factors influencing to the overlapping coupling algorithm configuration, however, the coupling position is highly case dependent and will not be discussed here. We do not investigate the impact of grid size mesh resolution either since the effects in coupled models is not much different than in non-coupled models. (2) The most interesting flow property in this test case is the hydraulic jump feature and the hydraulic jump is mainly classified by the approaching Froude number. (3) In section 3.3, we briefly introduced a condition, the CFL condition, for the stability of unstable numerical methods. The CFL number defined in equation (3.4.12) gives the upper limit for the time step if the mesh resolution is fixed. On the premise that the simulation converges, by increasing the CFL number we shorten the computation time at a price of decreasing the accuracy.

Due to the facts above, three variables are accordingly selected as uncertainty factors: (1) the distance of the overlapping distance (`OL Distance`), (2) the incoming Froude number (Fr) and (3) the maximum Courant number (`CFL` number).

Distance of the overlapping region

The overlapping distance (OL Distance) can be freely adjusted during the mesh generation process. To analyse the influence of the overlapping distance, a series of tests with different overlapping distances was conducted. Equation 4.3.7 is used to estimate a 5 [m] hydraulic jump length in the test case. To cover a wide range of variance considering the jump length and the grid spacing, the overlapping distance varying from 0.5 to 7 [m] are tested in this study, noted as OL Distance 0.5, OL Distance 1, OL Distance 3, OL Distance 4, OL Distance 5, OL Distance 7 respectively in following figures.

A glance of surge travelling process from Ω_{3D} to Ω_{2D} in terms of flow depth with different overlapping distances are presented in figures 5.12 and 5.13: from T = 5 - 5.5 [s], the surge is entering the overlapping region (figure 5.12) and from T = 7 - 7.5 [s], the surge is leaving the overlapping region (figure 5.13). In overlapping regions, only `OL 2D3D-2D` results are displayed. The left overlapping interface keeps the same at X = 50 [m]. The grey column implies the largest overlapping region applied in this test which is from 50 to 57 [m].

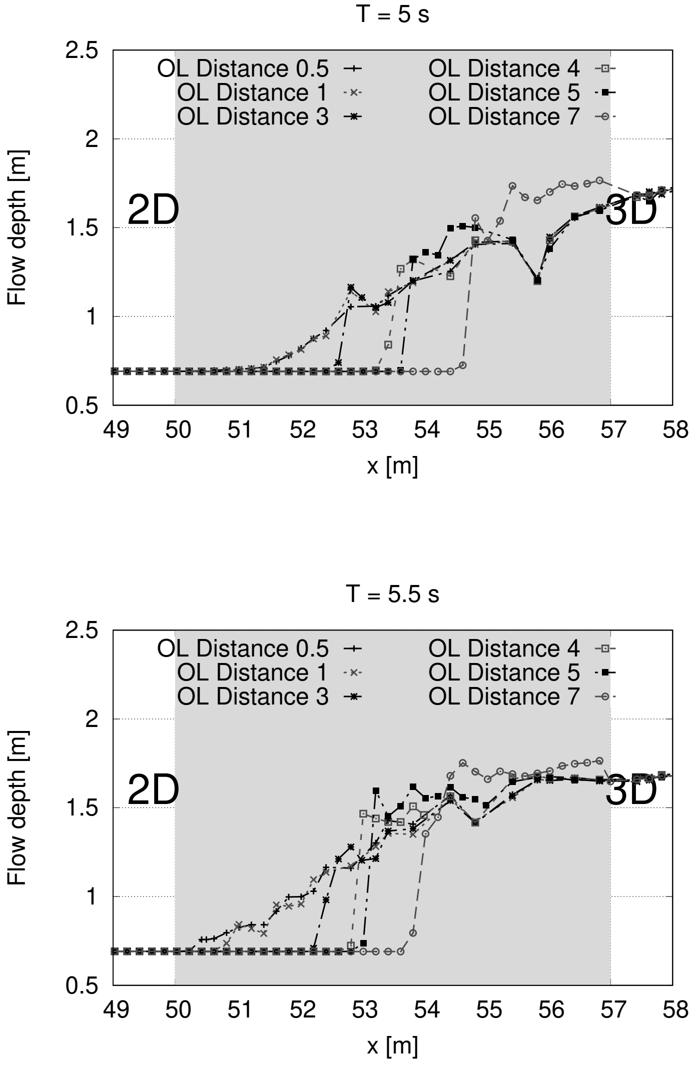


Figure 5.12: Surge travelling from the Ω_{3D} into the overlapping region with different overlapping distances at 5 - 5.5 [s]

Before all surge fronts passed the overlapping region, T = 5 - 7 [s], the location of surge fronts has a linear relationship with the OL Distance: the larger OL Distance leads to the more delayed surge front. Once the surge travels through the left interface, reflected waves are generated and as a consequence, disturbances appear in the Ω_{3D}. At T = 7.5 [s], the reflected wave is situated between X = 50 - 50.5 [m], thus disturbances are only show up with OL Distance 0.5.

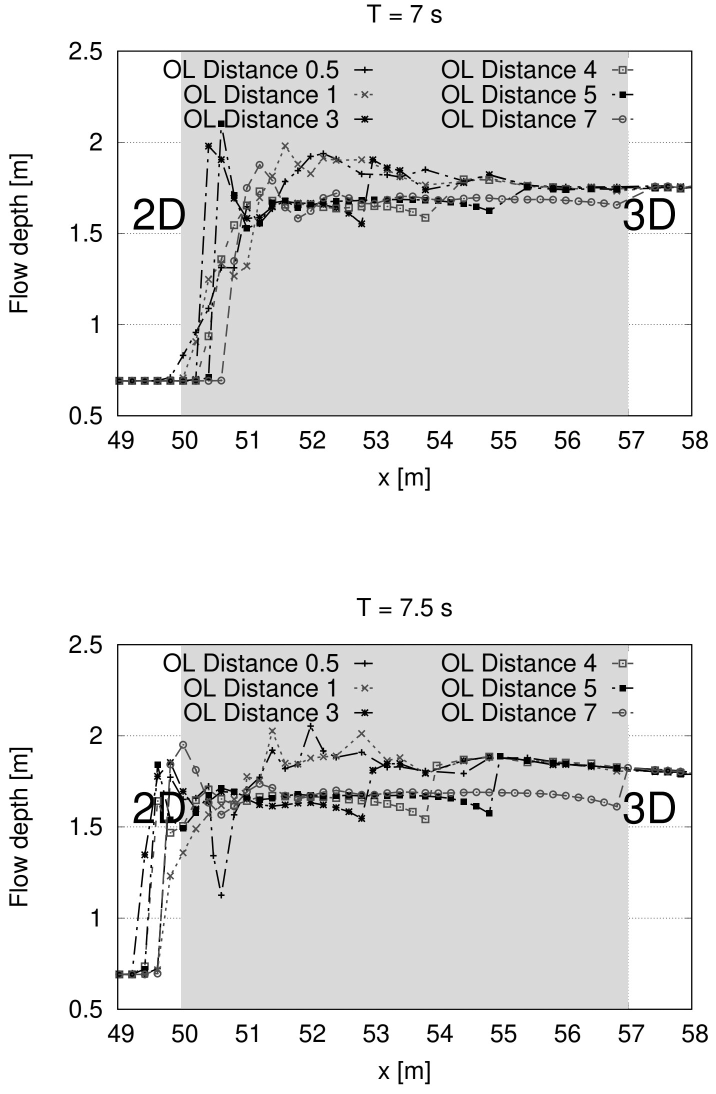
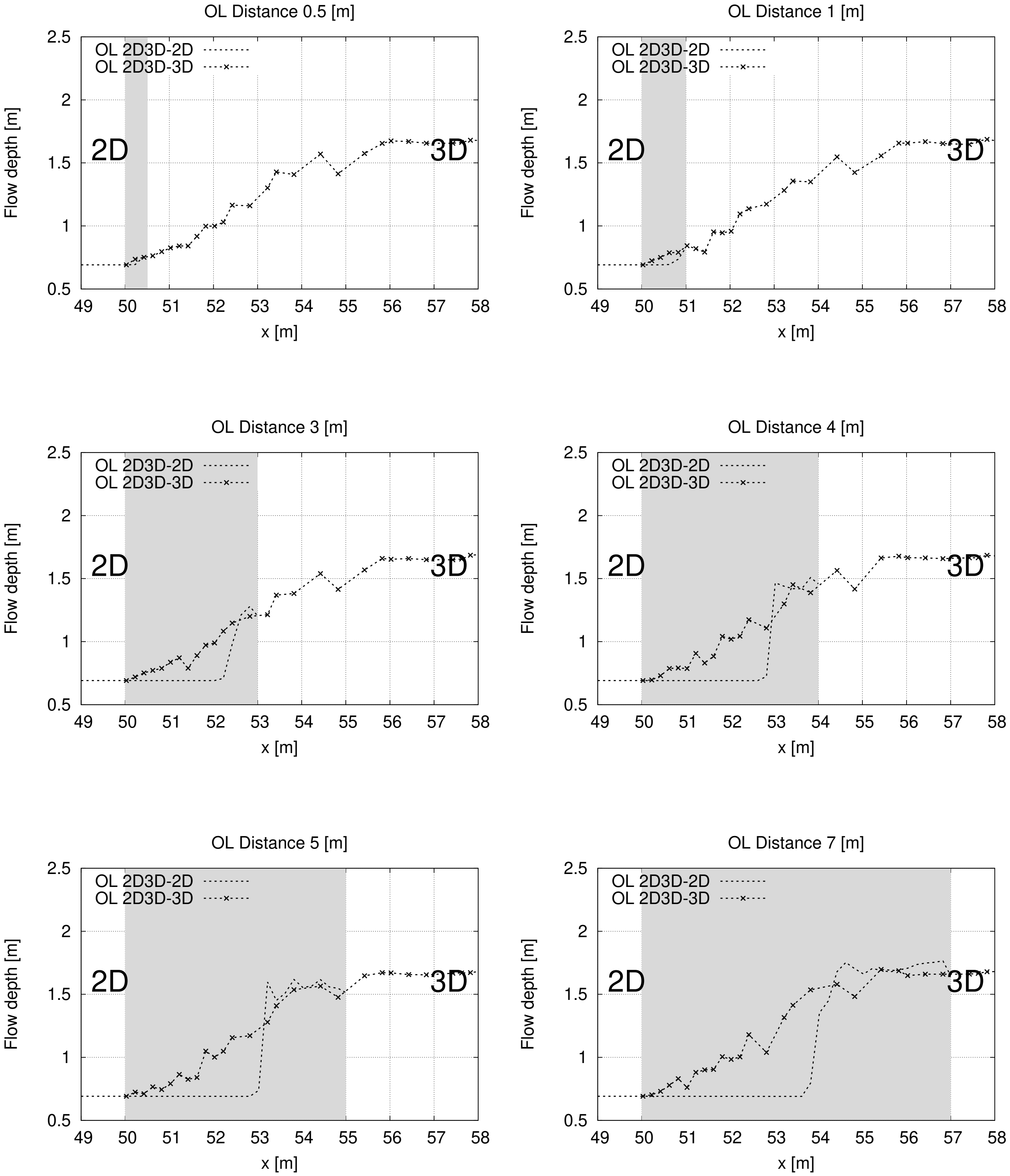


Figure 5.13: Surge travelling from the overlapping region into the Ω_{2D} with different overlapping distances at 7 - 7.5 [s]


OL Distance 0.5 [m]

OL Distance 1 [m]

OL Distance 3 [m]

OL Distance 4 [m]

OL Distance 5 [m]

OL Distance 7 [m]

Figure 5.14: Surge transfer with different overlapping distances at T = 5.5 [s]

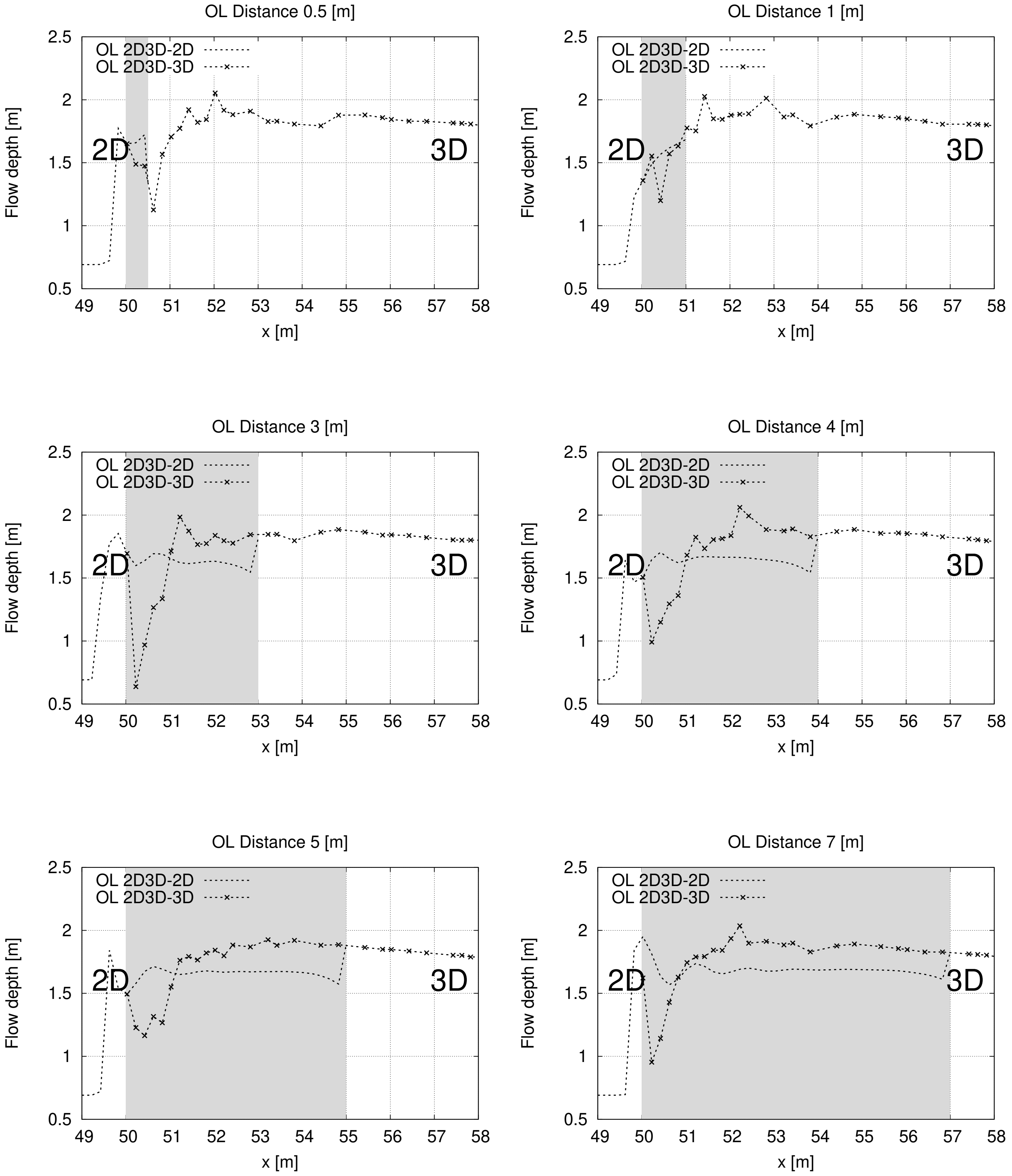


Figure 5.15: Surge transfer with different overlapping distances at T = 7.5 [s]

Elaborated comparisons in flow depth at two specific time stamps (*i.e.* 5.5 and 7.5 [s]) are further given in figures 5.14 and 5.15. At T = 5.5 [s], for each overlapping distance, the `OL 2D3D-3D` jump toe touches the left overlapping interface, while the `OL 2D3D-2D` jump toe stays a certain distance downstream from the left overlapping interface. That means the same incoming supercritical flow information is assigned to all these Ω_{3D} inlet patches at X = 50 [m] and different flow data are imposed to the Ω_{2D} outlet patches from X = 50.5 - 57 [m]. A steepened wave front is generated followed by the wave detection. The distance between `OL 2D3D-2D` and `OL 2D3D-3D` jump toes become smaller when the two overlapping interfaces stay closer.

2 [s] later indicated in figure 5.15, the wave front leaving the overlapping region enables the detection of waves by the 2D sub-solver and feeds back to the Ω_{3D}. Disturbances are therefore induced in the overlapping regions. The reflected wave is first generated with the smallest OL Distance and transfer backwards. These reflected waves can be considerably fluctuating with large OL Distances, *e.g.* OL Distance 3 and OL Distance 7. It is difficult to summarize that a linear connection exists between the larger overlapping distances and the model's ability. From the observation, the smaller OL Distances of 0.5 and 1 [m] have smaller disturbances.

The final state of surge travelling in our simulations at T = 20 [s] is exhibited in figure 5.16. No conspicuous trend can be concluded regarding the relationships between the OL Distances and the surge front locations. The smallest OL Distance 0.5 produces the highest front height, however, the largest OL Distance 7 engenders the second highest front height. As a result, observations of the surge front height do not give a clue on how the OL Distances impact on the surge front heights. Regarding the free surge downstream the surge front, smaller OL Distances of 0.5 and 1 [m] are smoother than others.

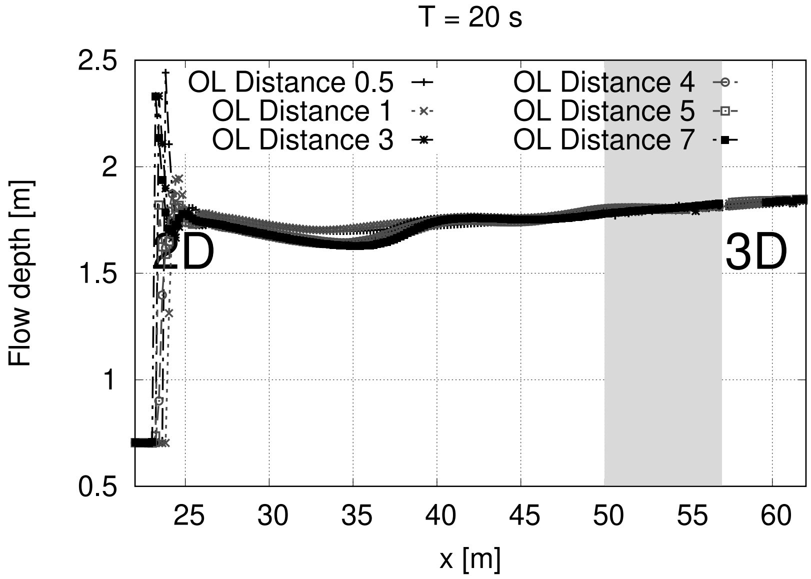


Figure 5.16: Surge travelling with different overlapping distances at 20 [s]

Approaching Froude number

A hydraulic jump can be only initiated when the Froude number Fr of the incoming flow is larger than 1. This Fr is also known as the prejump Fr before the hydraulic jump. Table 5.2 classifies roughly the water jump into five types based on the incoming flow Froude number. An insignificant disturbance of the water surface is originated by the prejump Fr ranges from 1 to 1.7. A standing or undulating wave is thus developed. When the $Fr > 1.7$, the hydraulic jump is triggered. A weak jump is characteristics by a series of small rollers and no free surface undulation with an increasing incoming Fr. When the Fr keeps increasing and is greater than 2.5, an unstable oscillating jump is formed. If the incoming flow has a big $Fr > 4.9$, a steady hydraulic jump is constructed with no displacement. This section focuses on the upstream moving surge therefore the prejump Fr between 1 and 4.5 will be covered.

Prejump Fr	Descriptive characteristics of jump
1.0 - 1.7	Standing or undulating wave
1.7 - 2.5	Weak jump (series of small rollers)
2.5 - 4.5	Oscillating jump
4.5 - 9.0	Stable clearly defined well-balanced jump
> 9.0	Clearly defined, turbulent, strong jump

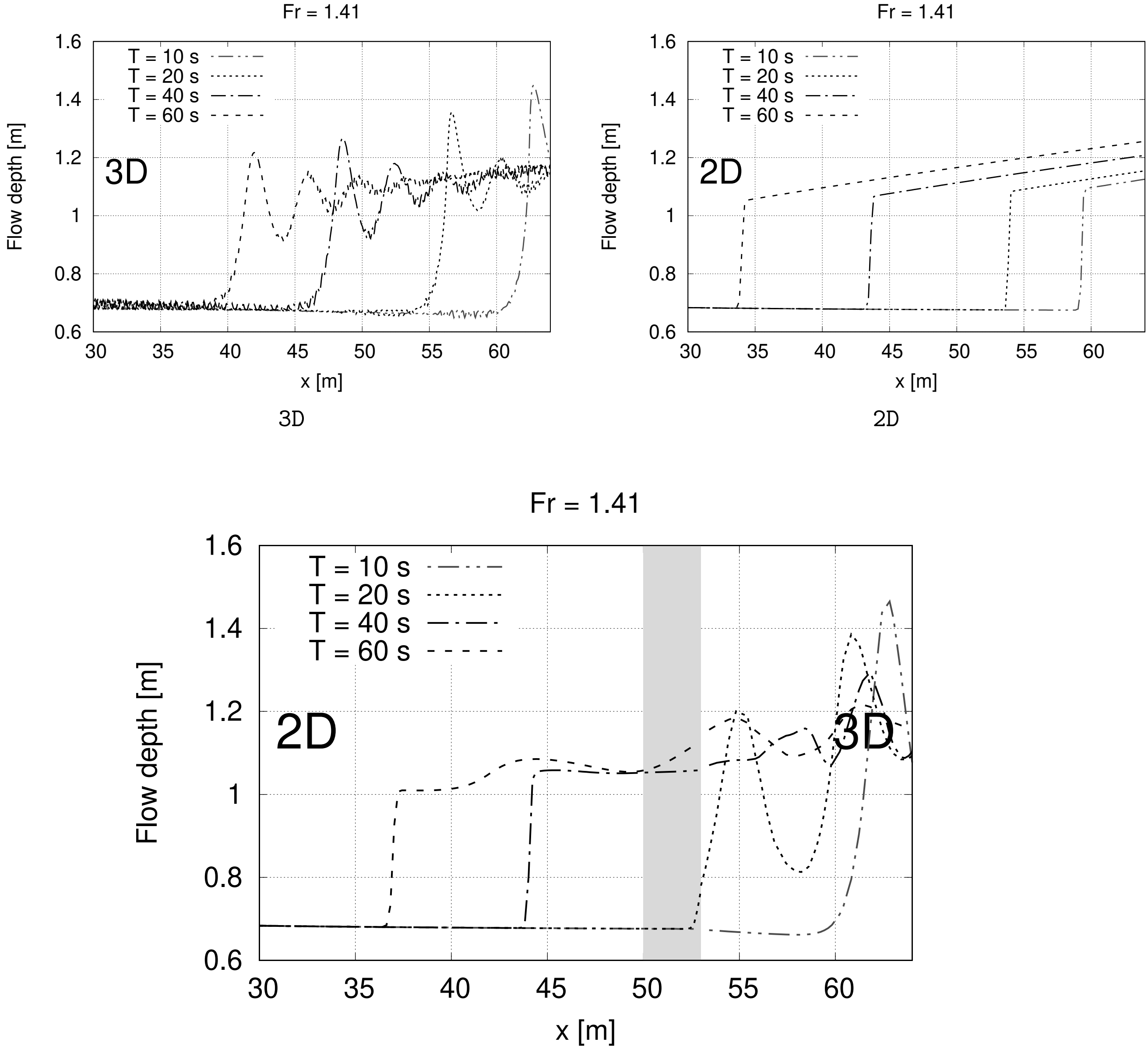
Table 5.2: Hydraulic jump characteristics (Goris (2006))

The prejump Fr expresses the Froude number right before the upstream jump toe. Under the circumstances of a moving jump, the relative prejump Fr should be replaced as the criterion, assuming the observer is moving with the jump. In following sub-tests, an instant relative Fr_{rel} is recomputed using flow information at T = 20 [s]. In this way, the original set-up case introduced previously in this section has a 1.98 prejump Fr_{rel} followed a weak jump.

Three sub-cases are conducted here with relative Froude numbers of 1.41, 1.98 and 3.24 in each instance. This sensitive study focus on a general moving jump behaviors, the 2D sub-solver results denoted as `OL 2D3D-2D` are displayed in the overlapping region as a simplification.

When replacing the fully closed wall at the outlet by a partially opened sluice gate, upstream-travelling undulating waves ($Fr = 1.41$) generated by three models are shown in figure 5.17. The first wave crest at T = 10 [s] in `3D` result is as twice as the incoming depth and sequential waves have smaller wave height. The wave amplitude decreases as it propagates upstream. Such wave is also called the wave jump (Chen (2015a)). Such bell shape wave with is not formed by the `2D` model, instead, a flattened wave surface is produced.

The `OL 2D3D-2D` model just represents the undular wave in a coupled manner that bell shape wave is formed in $\Omega_{3\mathrm{D}}$ and it tends to flatten as it transfer towards $\Omega_{2\mathrm{D}}$. Obvious unexpected disturbances can be hardly observed for the undular hydraulic jump at the overlapping region. The OL 2D3D-2D wave transfer through the overlapping region is quite smooth. However, the OL 2D3D-2D sequential wave crest has a larger amplitude than the leading crest, which differs from the 3D observations.



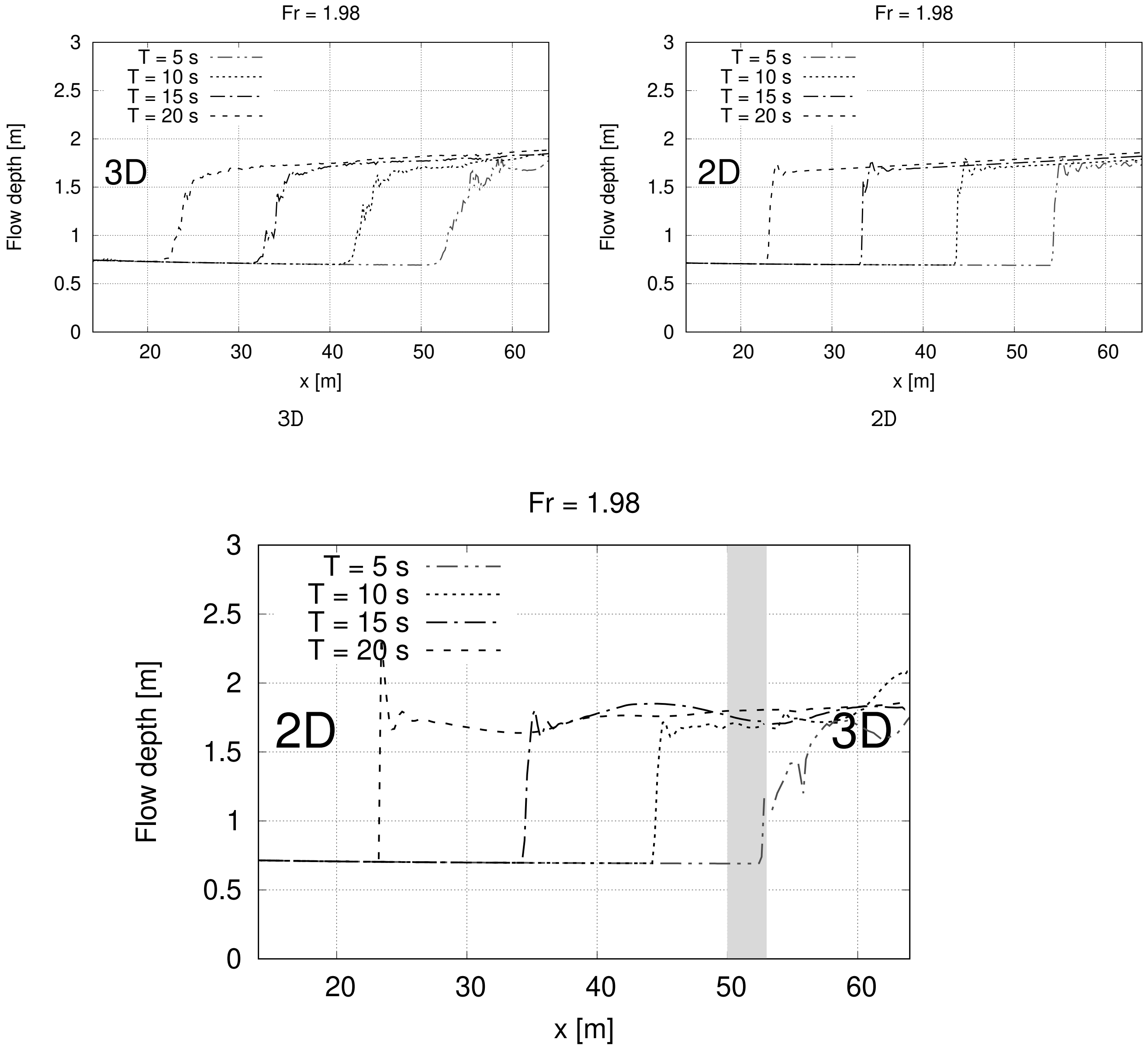
3D

2D

OL 2D3D-2D

Figure 5.17: Undular hydraulic jump time-history with $Fr_{\mathrm{rel}} = 1.04$

Moving hydraulic jumps with higher prejump Fr_{rel}, for instance 1.98, are exhibited in the following figure 5.18. This case was the one introduced previously as the surge transfer test. A larger ratio between the flow depth before and after the jump can be noticed with a larger Fr_{rel}. 3D and 2D results are in good agreement. Small disturbances are observed in the OL 2D3D-2D model.



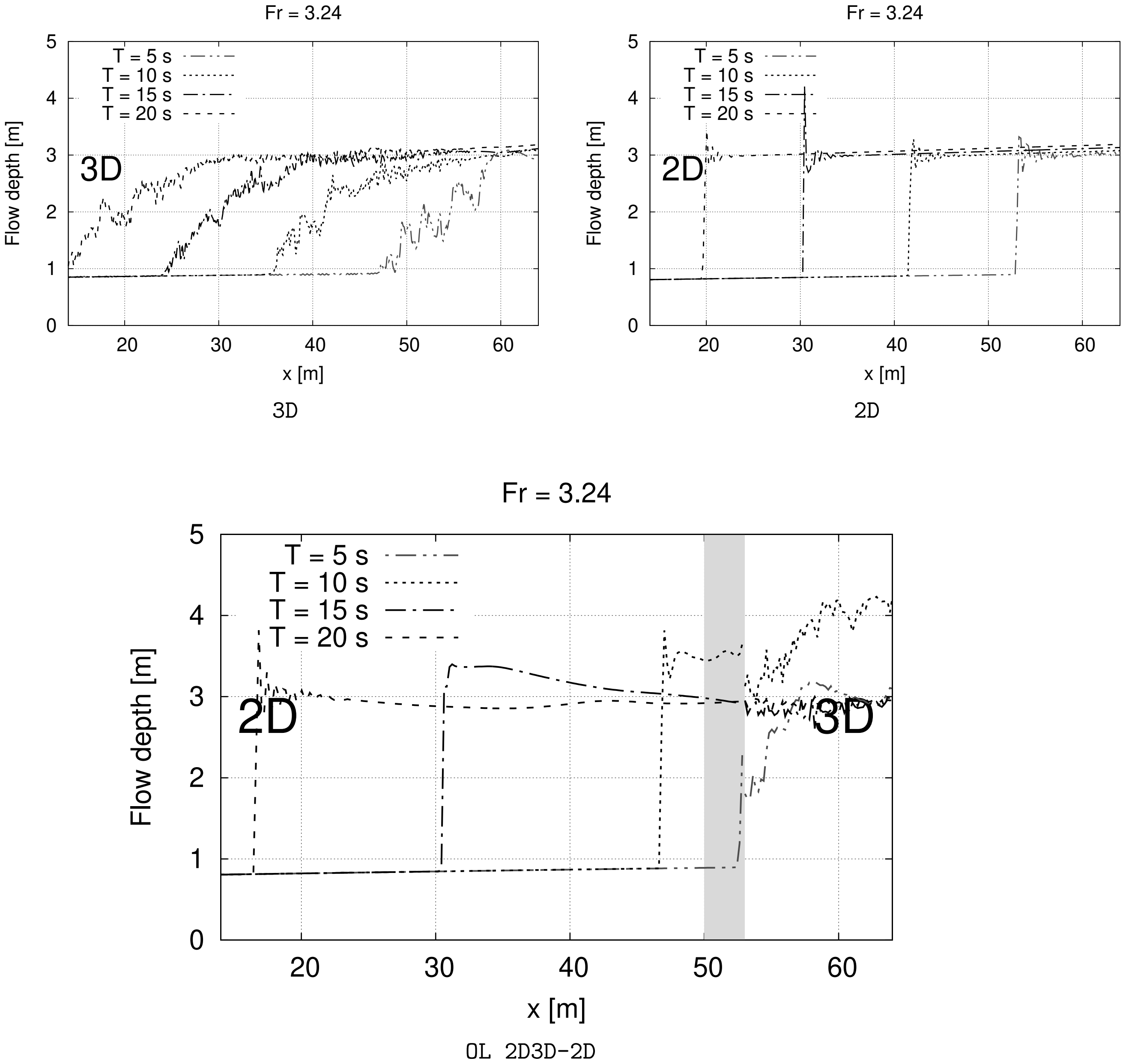
3D

2D

OL 2D3D-2D

Figure 5.18: Weak hydraulic jump time-history with $Fr_{\mathrm{rel}} = 1.98$

When further increases the Fr_{rel}, an oscillating hydraulic jump is triggered. Figure 5.19 reproduces oscillating jumps transfer ($Fr_{\mathrm{rel}} = 3.24$). A larger amplitude disturbance occurs by the `OL 2D3D` at T = 10 [s] leading to an overestimated wave height and a heavily delayed wave front in the $\Omega_{2\mathrm{D}}$. Such undesirable wavy disturbances gradually damp out and intend to achieve the same wave level as the `3D` and the `2D` outputs.



3D

2D

OL 2D3D-2D

Figure 5.19: Oscillating hydraulic jump time-history

A series of moving hydraulic jump tests based on different Fr_{rel} has been simulated here. With respect to these tests, all upstream-travelling hydraulic jumps are able to pass the coupling region and to transfer upstream. Higher Fr_{rel} produces higher hydraulic jump depth ratio and provokes stronger reflected waves. The wave front arriving time during T = 5 - 10 [s] retards more severely under the circumstance of a higher Fr_{rel}. These reflected waves and arriving time delays are diminished as the surges travel upstream. In a word, the back wave transfer over the overlapping region is sensitive to the Fr_{rel}.

Maximum Courant number

Sensitive analysis on the maximum Courant number in the $\Omega_{3\mathrm{D}}$, denoted as CFL in the figures, is studied. Figure 5.20 compares the surge representation of the coupled simulations at different time's with different CFL numbers. It concludes that the $\Omega_{3\mathrm{D}}$ is more sensitive with the CFL than the $\Omega_{2\mathrm{D}}$. However, there is no evident influence with respect to the entire surge transfer phase.

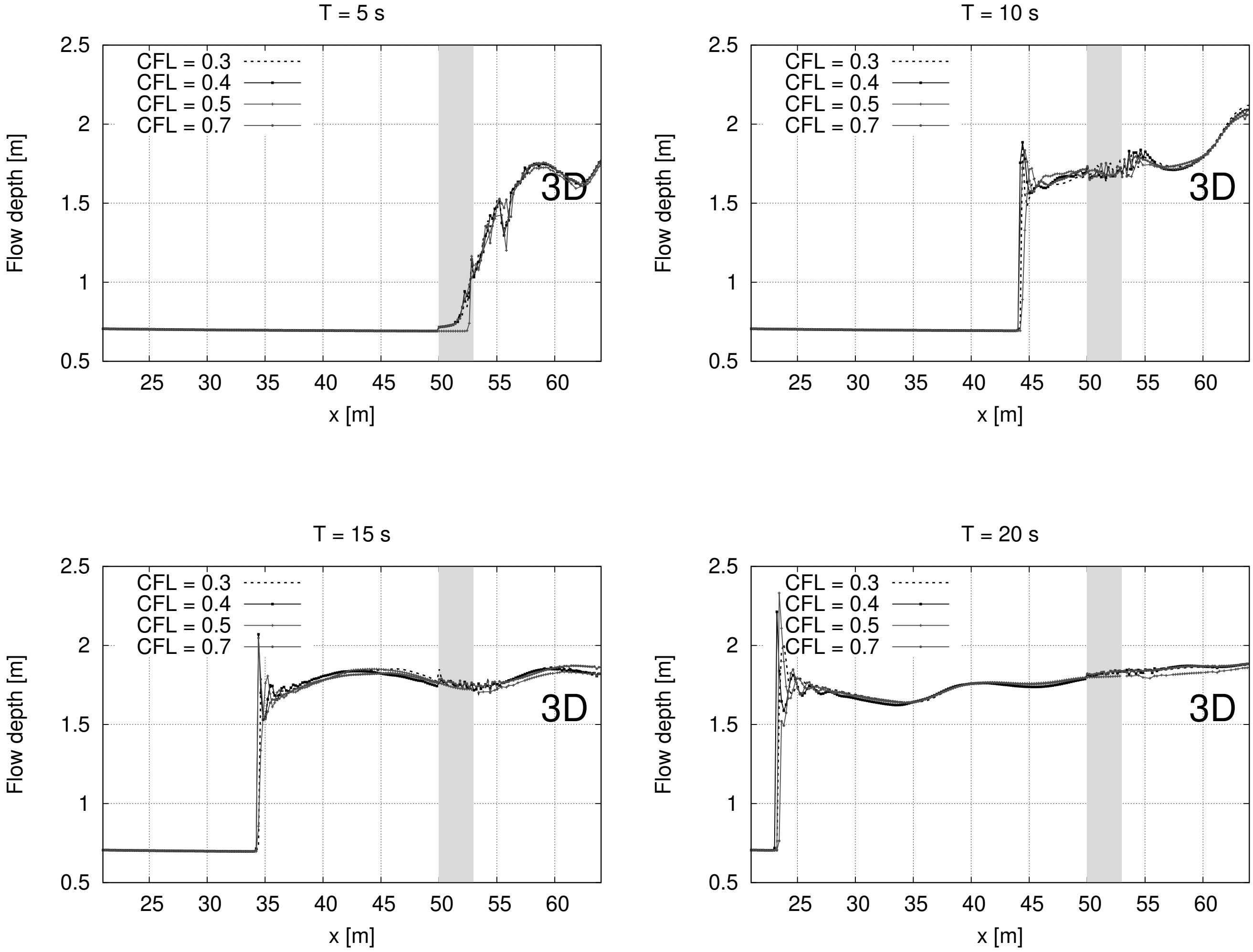


Figure 5.20: Surge travelling at 5 - 20 [s] with different CFL numbers of coupled simulations

5.2.5. Surge transfer from $\Omega_{2\mathrm{D}}$ to $\Omega_{3\mathrm{D}}$

The other configuration is built up for testing the surge transfer from the $\Omega_{2\mathrm{D}}$ to the $\Omega_{3\mathrm{D}}$, which means the surge is generated in the $\Omega_{2\mathrm{D}}$ and moves towards the $\Omega_{3\mathrm{D}}$, leading to the 3D->2D coupling connection. Under this circumstance, the flow depth and the velocity are prescribed at the $\Omega_{3\mathrm{D}}$ upstream boundary as the inlet inputs and a free outlet is set on the $\Omega_{2\mathrm{D}}$ downstream boundary to develop the steady state flow. A zero discharge is imposed to the the Ω_{2D} downstream boundary after reaching the steady state, the surge is therefore generated.

The comparison focusing on the wave front shape at T = 20 [s] between the three models is exhibited in figure 5.21. The wave front shape between the OL 2D3D and the 3D are fairly matched and the wave front positions among the three models are also in good agreement.

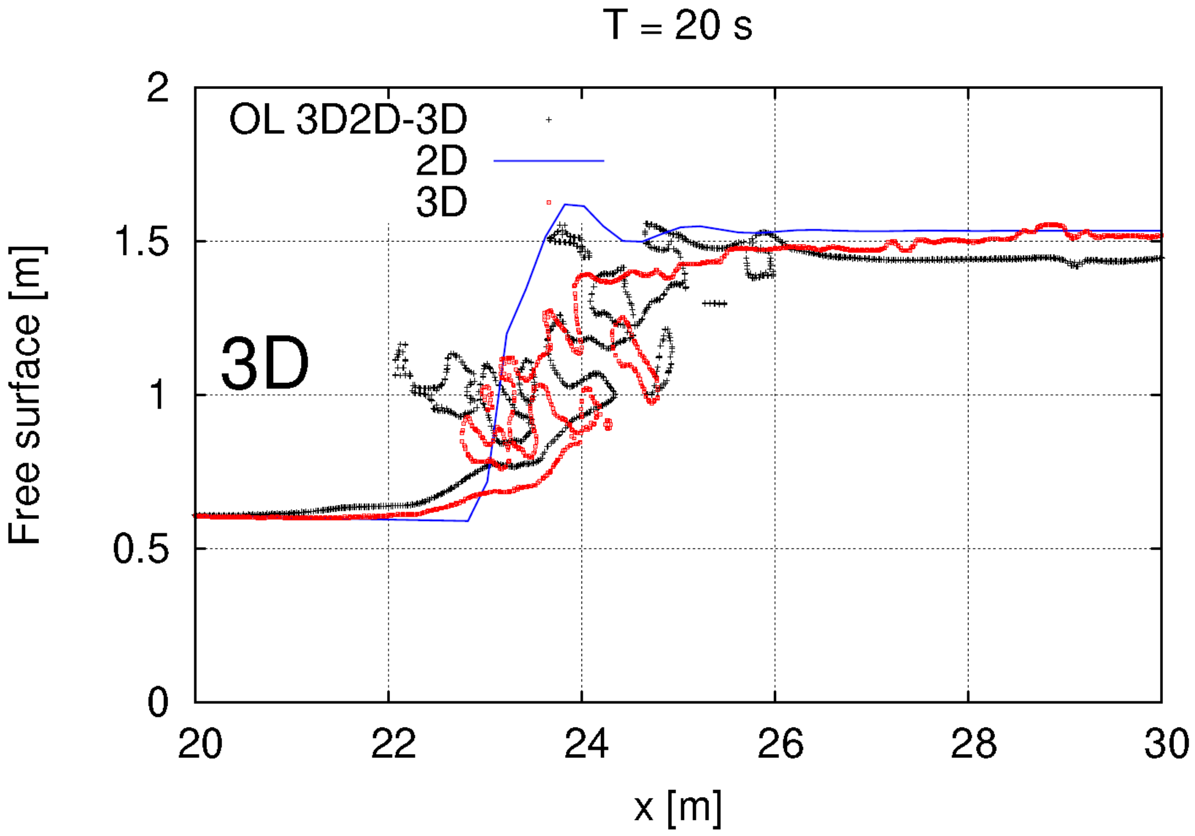


Figure 5.21: Surge representation at 20 [s]

As illustrated in figure 5.22, similar to the 2D->3D coupling connection, the OL 2D3D results behave in an analogous manner compared to the 2D and the 3D in the corresponding subregions. At T = 6 - 8 [s], the Ω_{3D} detects a subcritical flow condition and the back surge is moving in the overlapping region. The reflected waves are also observed at the right interface ans transfer upstream. As soon as the surge transfers through the overlapping region, a milder wave front is developed by the 3D sub-solver that a wave damping effect due to the skewed wave front is imposed from the Ω_{3D} to the Ω_{2D} via the overlapping region. In the mean time the surge propagates upstream, the OL 2D3D wave front fits the 3D wave front better and discrepancies between the OL 2D3D and the 2D free surface fade away.

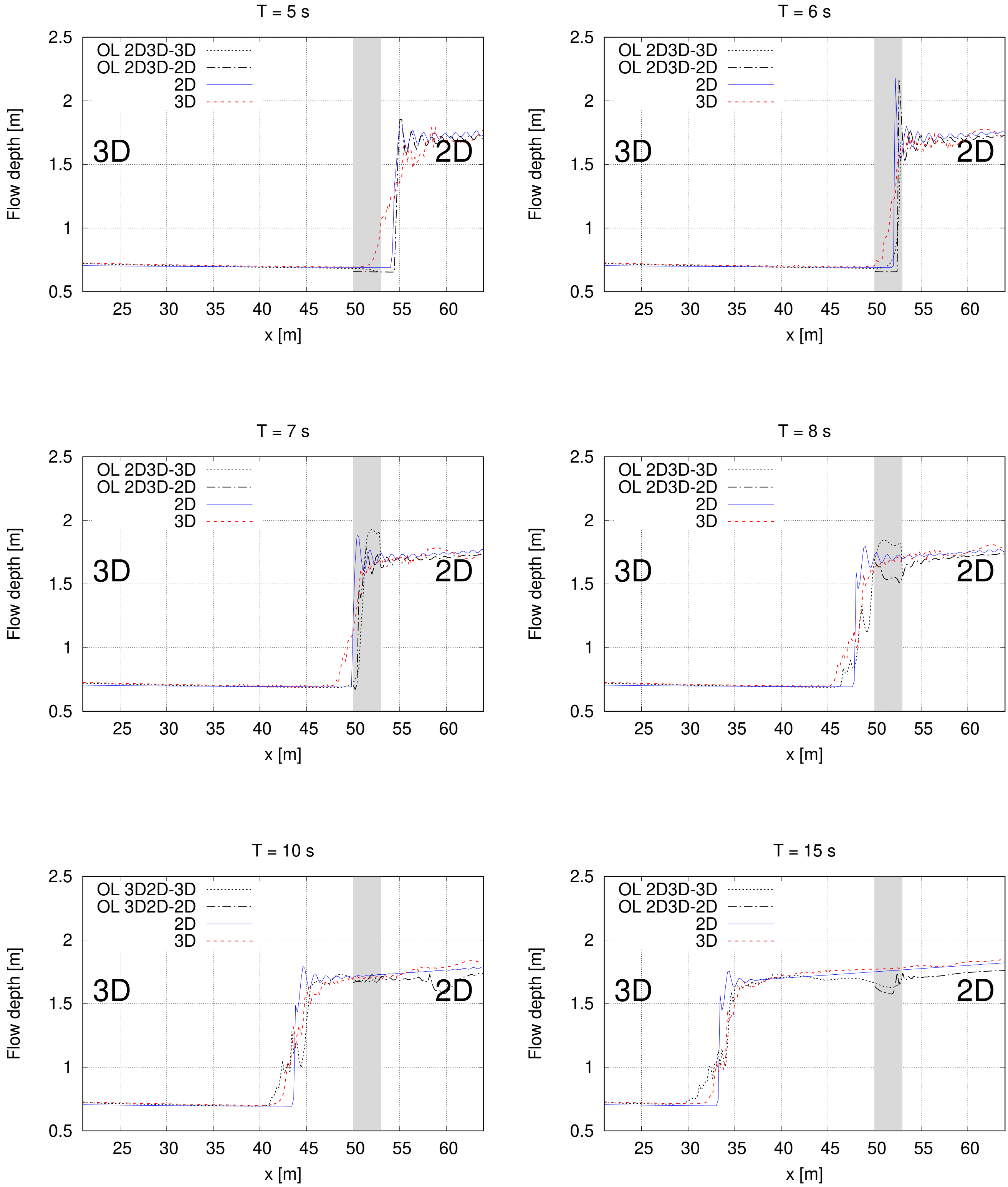


Figure 5.22: Surge travelling from 5 - 15 [s]

5.2.6. Summary

This section mainly proves the ability of the proposed coupling method in upstream-travelling surge simulations including sensitive studies on three uncertainties. This modelling was identified as an unsettled issue in Mintgen (2017) under the `2D->3D` coupling connection, where the back surge is blocked at the coupling interface in the Ω_{3D} and the water level in the Ω_{3D} keeps ascending. With the overlapping coupling algorithm, this back surge can cross the coupling common region and propagate upstream under both `2D->3D` and `3D->2D` coupling connections.

However, in the `2D->3D` coupling connection sub-case, reflected waves and heavy disturbances are concocted which severely makes the `OL 2D3D-3D` results less reliable during the period when surges transfer through the overlapping region. This undesired oscillation decades as the prejump Fr_{rel} decreases. Viewed from another perspective, the `OL 2D3D-2D` results stay smooth which could be used as the outcome on behalf of the overlapping region.

For practical purposes, a coupling model is usually applied where the Ω_{2D} covers large scale domains by showing a general flow behavior and relative significant properties of the local flow should be embedded in the Ω_{3D}. With respect to this, the transient phase of a surge travelling through the coupling area is less interesting as long as the coupling results match well in a global sense with the non-coupling models results.

5.3. Downstream-travelling wave over a supercritical flow

As mentioned in the beginning of this chapter, shallow wave transport will be analysed in this and the next sections by superposing a solitary wave on top of the steady state flow. For both two scenarios, a flat bottom channel with 100 [m] length is established and the overlapping distance of 3 [m] is chosen which lies between X = 50 - 53 [m], and the mesh resolution is 0.1 [m]. Along with the solitary wave transport, the wave shape remains unchanged, whereas the initial wave peak amplitude decreases and two waves with smaller peaks are generated. According to the solitary wave properties, the wave travels at a constant speed and the taller the wave the faster it travels. This wave is therefore ideal for validating the solver performance. The solitary wave is generated using a Gaussian function, equation (5.3.1)

$$h_{\mathrm{w}} = Ae^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{5.3.1}$$

where h_{w} [m] is the wave height, A [m] is the wave peak, μ [m] is the position of the wave center of the peak and the standard deviation term σ controls the wave length.

This section mainly tests the `OL 2D3D` performance for a wave travelling downstream in a supcritical flow. The geometry is sketched in figure 5.23. Figure 5.23 outlines the simulations set-up: the steady state supercritical flow is developed at the first place and the solitary wave is then superposed upon the flow free surface. The undisturbed flow has a specific discharge of 3.886 $[\mathrm{m}^2 \cdot \mathrm{s}^{-1}]$ and a flow depth of 0.72 [m] resulting in a supercritical flow with $Fr = 2$. Within the present set-up, the initial wave center is located at X = 40 [m], has the wave peak of 0.2 [m] and $\sigma = 2.5$. The initial Gaussian wave stands just right upstream the overlapping region in order to capture early wave movements through the overlapping region. Both `2D->3D` and `3D->2D` coupling connections are conducted.

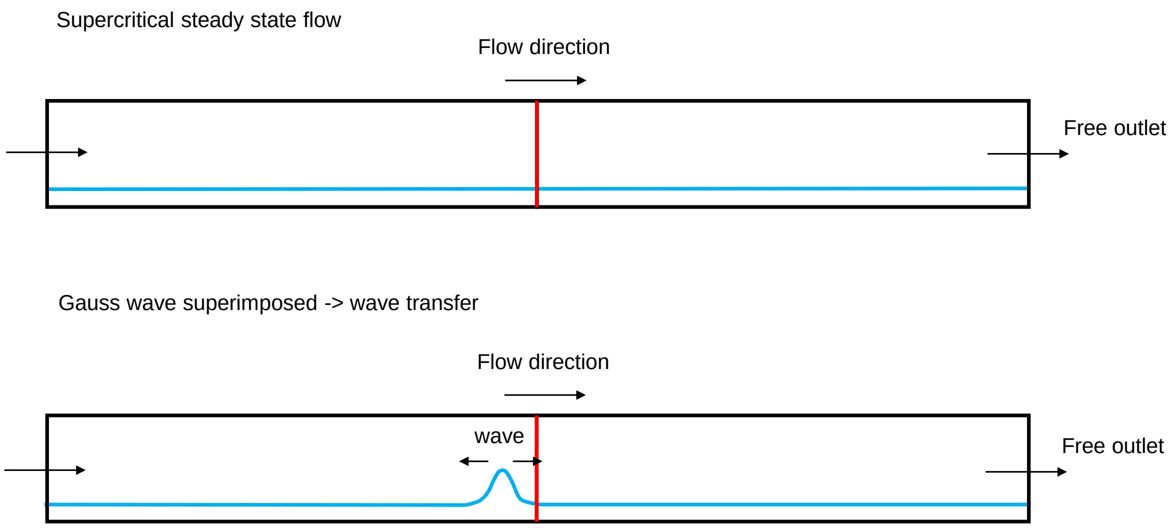

Supercritical steady state flow
Flow direction
Free outlet
Gauss wave superimposed -> wave transfer
Flow direction
wave
Free outlet

Figure 5.23: Downstream-travelling wave simulations set-up

The superposing shallow wave breaks into two smaller amplitude waves of which locally one propagates upstream and the other one propagates downstream, named the receding wave and the advancing wave respectively. As a matter of fact, the receding wave is overwhelmed by the incoming supercritical flow resulting in globally downstream-travelling waves. In general, the advancing wave has a higher wave peak and travels faster than the receding wave.

5.3.1. Results of `2D->3D` connection

Wave transfer process from the Ω_{2D} to the Ω_{3D} within the `2D->3D` connection is shown in the following figures together with the respective results of the `2D` and the `3D` simulations. The initial wave stands in upstream of the overlapping area and intends to pass the coupling region. From observations, the advancing wave travels at 8 $[\mathrm{m} \cdot \mathrm{s}^{-1}]$, while the speed of the receding wave is 2 $[\mathrm{m} \cdot \mathrm{s}^{-1}]$. Due to their different wave speeds, distance between their wave center keeps increasing and a longer time is needed for the receding wave to transverse the overlapping region.

As in figure 5.24, the `OL 2D3D` advancing wave passes the overlapping region smoothly and lies in between the `2D` and the `3D` advancing waves. As the advancing wave moves away from the receding wave, a vibrating wave tail grows in the Ω_{3D} driven by a non-shallow wave effect. The original ratio of the flow depth to the wave length is $0.92 \ / \ 20 = 0.046$, which is less than 0.05. Based on the shallow water wave theory, this wave is classified into the shallow water wave group. Shallow water wave speed only depends on the flow depth, therefore it is a non-dispersive wave. As the original wave bifurcates, the ratio of the advancing wave is recalculated by using its individual flow depth and wave length giving $0.88 \ / \ 15 = 0.057$ at $T = 2.5$ [s], which is out of the shallow wave region. Frequency dispersion phenomenon occurs at the advancing wave tail. This effect is not represented by the 2D shallow water equations solver since no distortion shows up at the `2D` advancing wave tail but appears in the `OL 2D3D-3D` and the `3D` advancing wave tails.

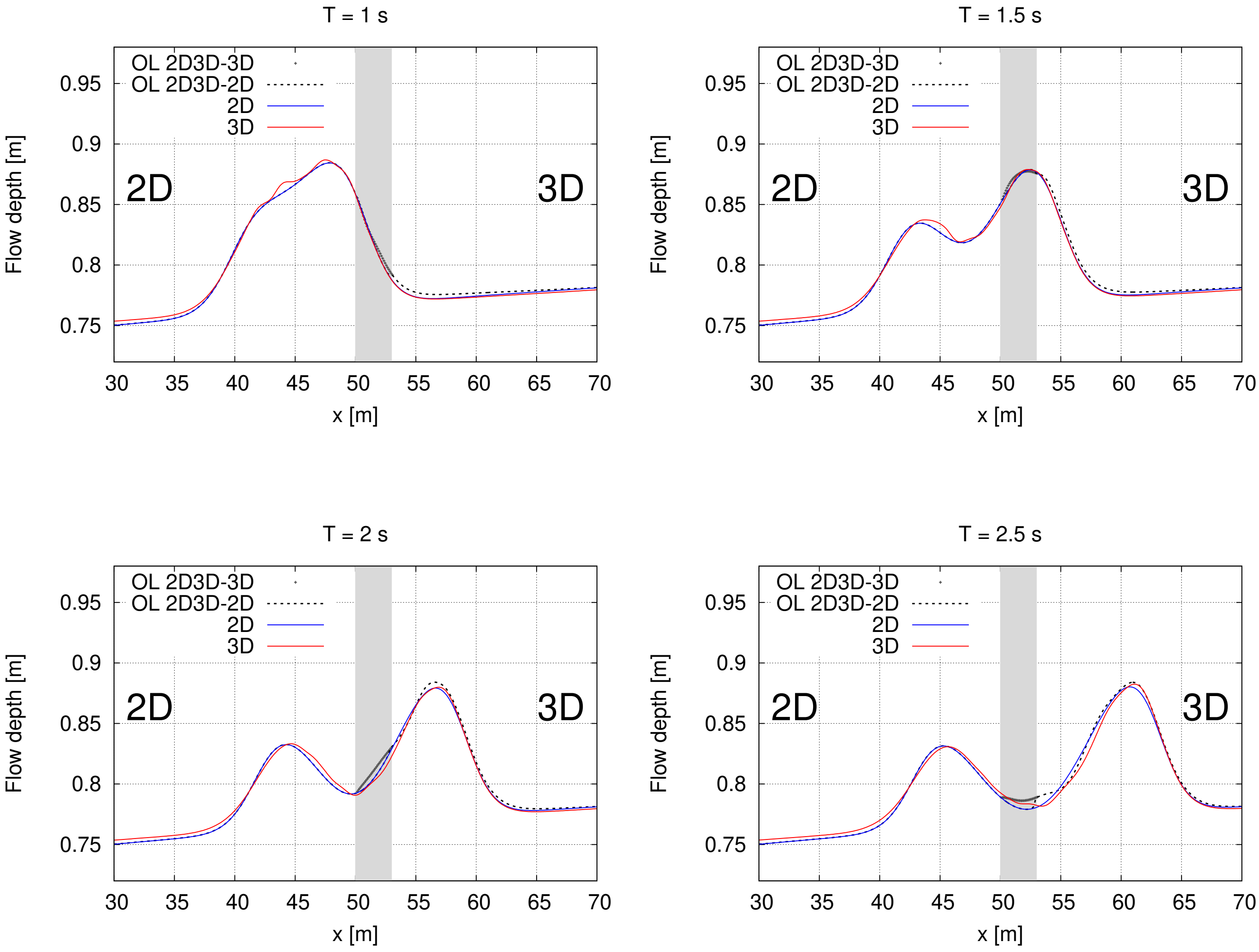


Figure 5.24: Downstream wave travelling at T = 1 - 2.5 [s]

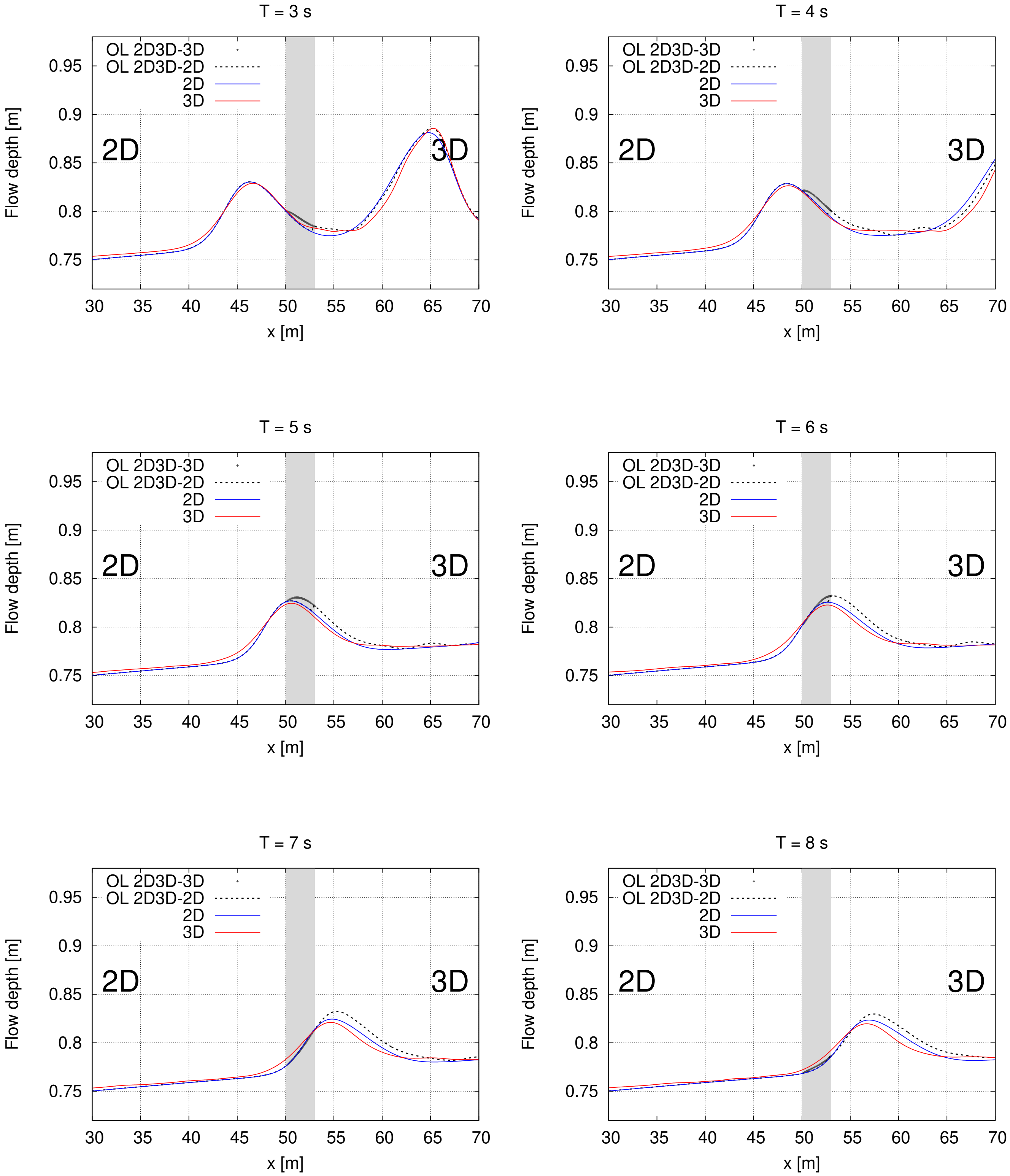


Figure 5.25: Downstream wave travelling at T = 3 - 8 [s]

The frequency dispersion is observed at the first time at T = 2.5 [s], where is located in the overlapping region. Due to the dispersion, a smaller amplitude of trough is produced by the OL 2D3D-3D and the 3D than the 2D in the overlapping region. This higher flow depth at X = 53 [m] in the OL 2D3D-3D is assigned to the OL 2D3D-2D as the Ω_{2D} outlet boundary condition. Consequently, a higher wave tail at the right side of the receding wave is further inherited from the dispersive wave effect at T = 3 [s] demonstrated in figure 5.25. An overestimated wave peak is hence generated at T = 6 [s]. However, the left wave surface of the OL 2D3D and the 2D are in good agreement.

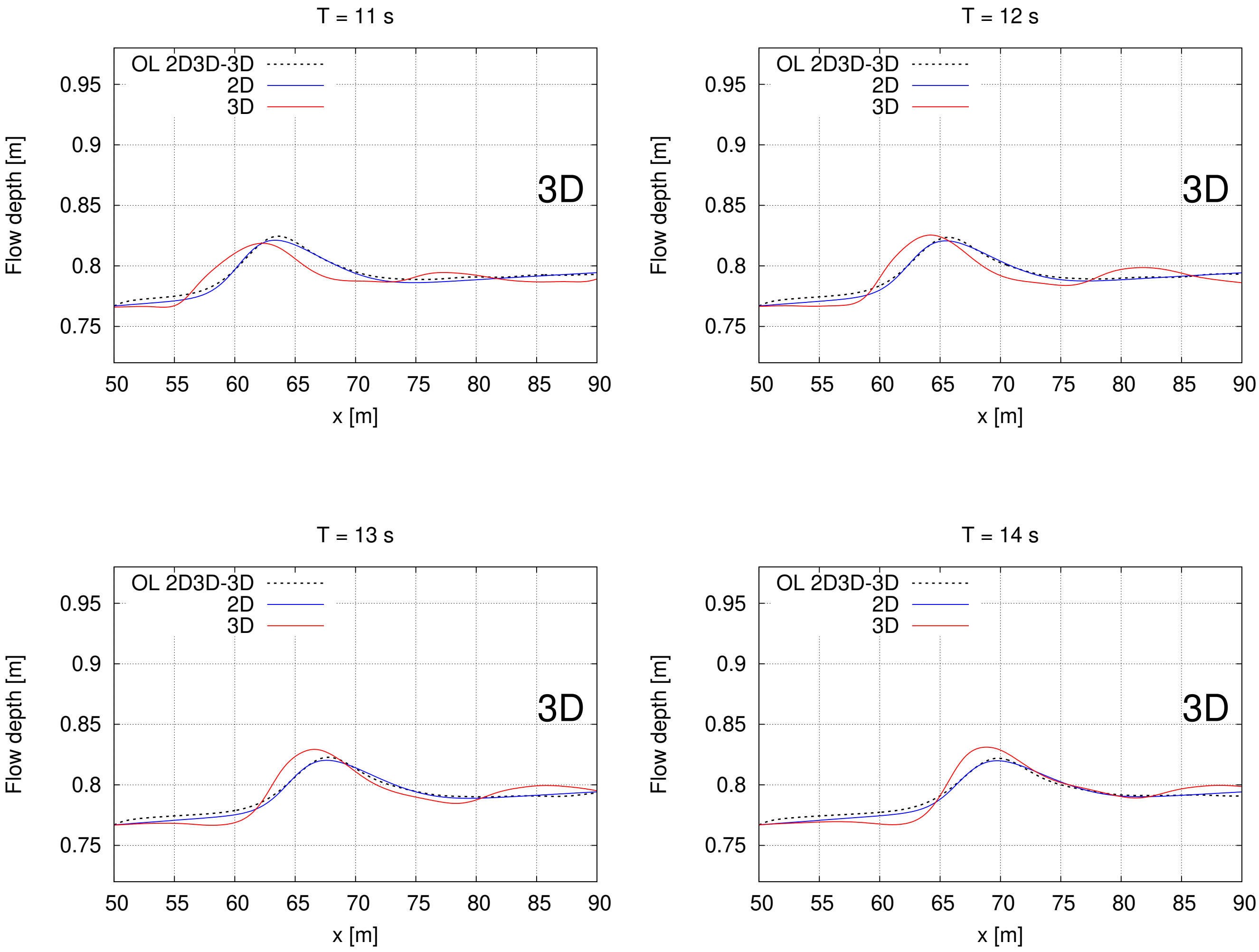


Figure 5.26: Wave travelling at T = 11 - 14 [s]

Figure 5.26 expresses the wave transfer further downstream in the Ω_{3D}. Although from T = 11 - 14 [s] the OL 2D3D wave entirely locates in the Ω_{3D}, a better fit between the OL 2D3D and the 2D wave representations is obtained than the OL 2D3D and the 3D. Wave peaks and two sides wave fronts are fairly overlapped between the OL 2D3D and the 2D and their ratio of the flow depth to the wave length becomes 0.825 / 25=0.033 indicating a shallow water wave. The 3D is still effected from the frequency dispersion resulting in a steeper wave face against the stream-wise direction and a higher wave peak.

5.3.2. Results of `3D->2D` connection

The `3D->2D` coupling connection results are briefly given in figure 5.27 at T = 1 - 14 [s]. Generally results from these models show similar waves representation as the `2D->3D` coupling connection results, the `OL 2D3D` wave is well matched with the `2D` wave along the whole simulation. Concerning the `OL 2D3D`, the advancing wave travels from the Ω_{3D} to the Ω_{2D}, no dispersion is generated by the 2D sub-solver, thus no dispersion effect impacts on the `OL 2D3D` receding wave.

5.3.3. Summary

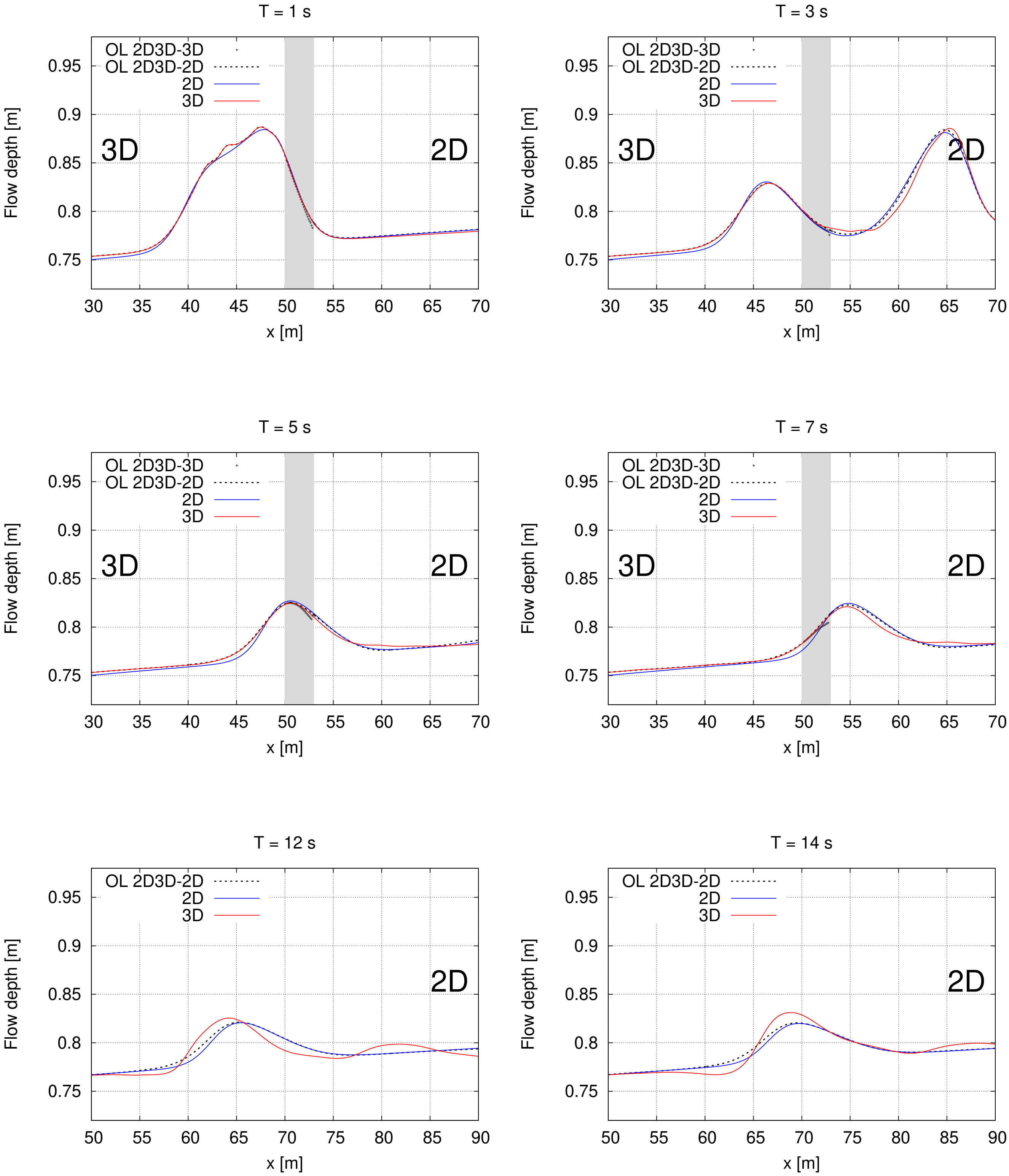
This case investigates the wave transfer downstream in supercritical flows over the overlapping region for both types of coupling connections: `2D->3D` and `3D->2D`. The two coupling conditions (`2D->3D` and `3D->2D`) perform similarly for the advancing wave and differently for the receding wave.

With both connection configurations, behaviors of the `OL 2D3D` advancing waves agree well with results from the `2D` and the `3D` models even during the overlapping region transport period. Frequency dispersion effects are observed at the advancing wave tails by the `3D` results and further impacts on the receding wave. When the wave is not fully within the shallow regime, dispersion rises and the desired wave signal can be distorted. According to the coupling algorithm, this oscillatory tail should be developed as well in the 3D sub-solver.

With the `2D->3D` connection, the oscillatory behavior resulted from the frequency dispersion occurs downstream of the overlapping region in the Ω_{3D}. The Ω_{3D} inlet inputs are adopted from the 2D sub-solver with the shallow water wave behavoir. Combining the shallow water wave information at the inlet patch of the Ω_{3D} and the dispersion effect inside the Ω_{3D}, an inflated receding wave peak is thus formed (*e.g.* T = 5 - 8 [s] in figure 5.25). However, when the Ω_{3D} locates upstream of the the Ω_{2D} (*i.e.* the `3D->2D` connection), the distorted wave information should occur downstream and cannot be generated by the 2D sub-solver in the Ω_{2D}, so that the receding wave passes the overlapping region without deformation.

After two waves cross the overlapping region, waves of the `2D` and the `OL 2D3D` models match again regardless of the connection type. Due to the supercritical characteristics, downstream-travelling flow information is always controlled by upstream. With the `2D->3D` connection, no oscillatory surface is computed by the 2D sub-solver, a shallow water wave property flow is therefore set as the input to the 3D sub-solver. As a result, the overestimated wave peak in figure 5.25 is adjusted and eventually follows a shallow water wave behavior. With the `3D->2D` connection, no disturbance is generated anyway and the downstream region is dominated by the 2D sub-solver.

In conclusion, behaviors of shallow water wave transfer in supercritical flows are dominated by the upstream input values. In general, the `OL 2D3D` complies more with the `2D` than the



T = 1 s

T = 3 s

T = 5 s

T = 7 s

T = 12 s

T = 14 s

Figure 5.27: Wave travelling from T = 1 - 14 [s]

3D . The frequency dispersion generated by the 3D solver is vanished after a while in the OL 2D3D.

5.4. Upstream-travelling linear wave transfer over a subcritical flow

Shallow water wave transfer in subcritical flows possesses distinct characteristic other than in supercritical flows. The wave of origin will still generate two small waves and the advancing wave travels downstream, however, the receding wave travels upstream. The upstream-travelling wave is mainly dominated by downstream. An upstream-travelling wave transfer over a subcritical flow is practised to further validate the coupling method.

In subcritical flows set-up usually requires the discharge at upstream and the flow depth at downstream as boundary conditions. Nevertheless, to reach the steady state subcritical flow over a flat bottom channel by the 3D model, the VOF indicator α_1 should be prescribed at the outlet, which leads to numerical instabilities as mentioned in chapter 4. The same case set-up made by Mintgen (2017) is thus adopted by adding an overfall is added at the end of the channel. In such way, numerical instabilities and disturbances caused by downstream boundary conditions can be avoided. The geometry and set-up of this case is therefore represented in figure 5.28: the entire channel is 80 [m] including a 5 [m] overfall, the channel height is 3 [m] as well as the overfall depth, and the bottom slope is 0.001. The overlapping region is allocated at X = 20 - 23 [m].

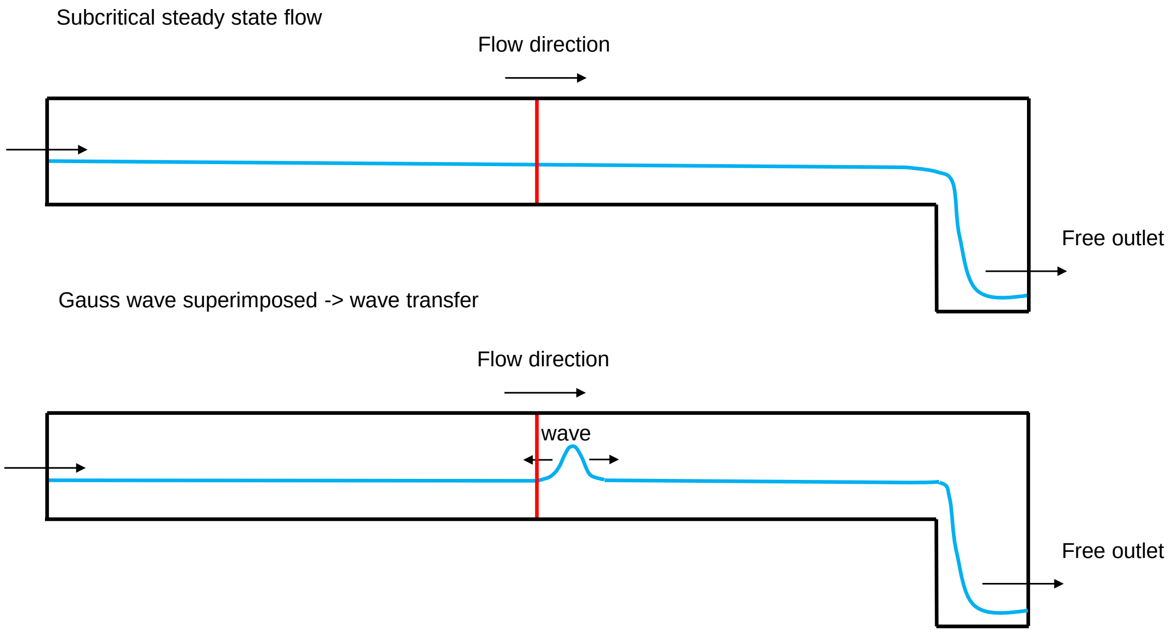

Subcritical steady state flow
Flow direction
Free outlet
Gauss wave superimposed -> wave transfer
Flow direction
wave
Free outlet

Figure 5.28: Upstream-travelling wave simulations set-up

With the help of the applying an overfall at the downstream of the channel, the set-up for prescribed flow depths at outlet boundary are not demanded any more. In the 2D simulation, a specific discharge of 2.49 [m^3 $\cdot$ s^{-1}] is given at the inlet and the Strickler coefficient of 52.3 is used in the whole domain, resulting in the inlet flow depth of 1.1 [m]. The relevant velocity distribution and the VOF indicator α_1 are therefore determined for the inlet boundary conditions specification in the 3D simulation.

Parameters for wave generation are A = 0.4 [m], $\mu = 30$ [m] and $\sigma = 2.5$ in this case, using equation (5.3.1). Higher flow depth is generated in subcritical flows resulting in higher ratio 1.4 / 20 = 0.07, which is not in the fully shallow regime. Frequency dispersion is expected in this case.

5.4.1. Results of 2D->3D connection

The transport of the superposing wave under the 2D->3D connection during the first 4 [s] is demonstrated in figures 5.29.

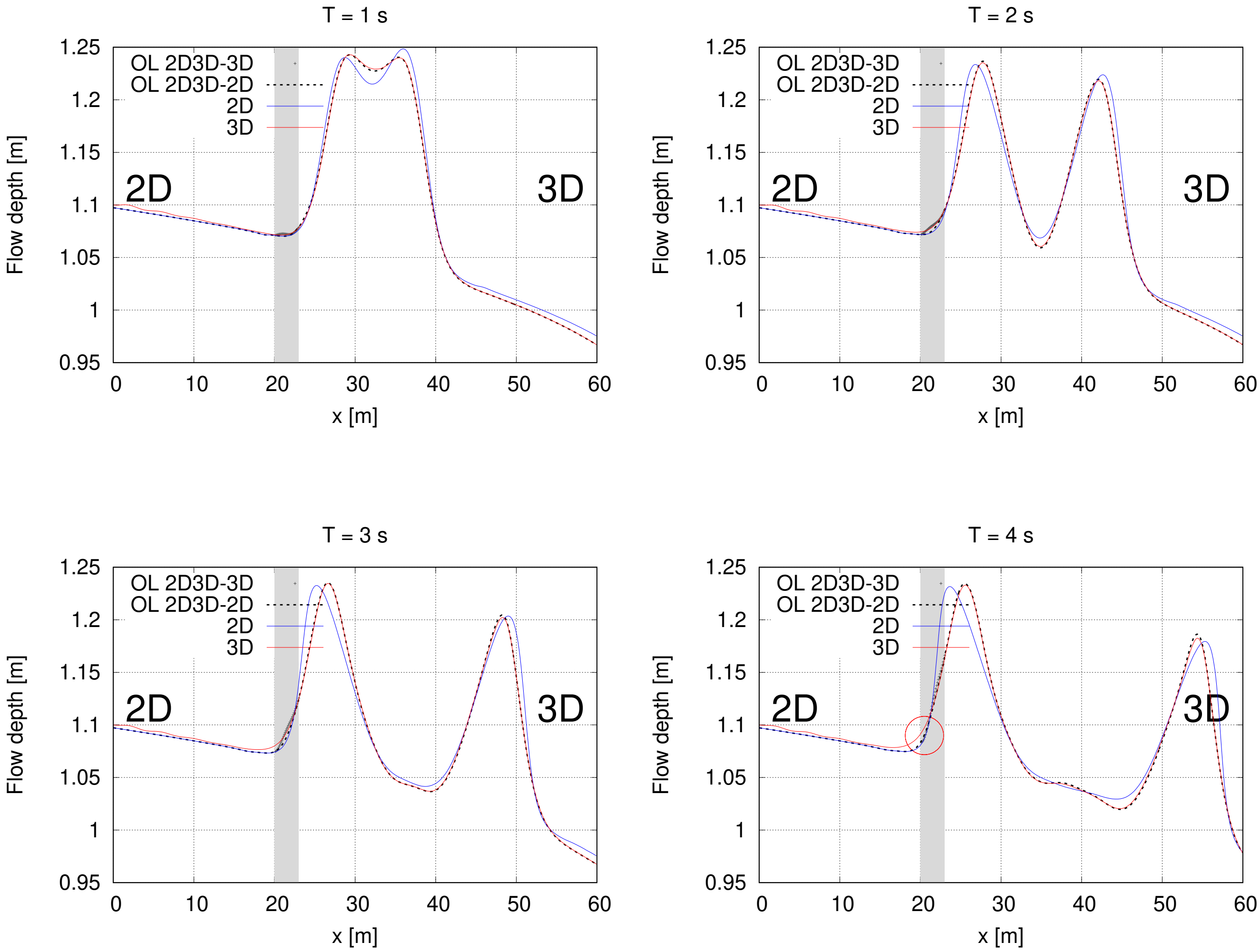


Figure 5.29: OL 2D3D depth averaged velocity at T = 1 - 4 [s]

The resulting advancing and the receding waves travel in an opposite direction with the identical wave shape and their distance keeps increasing. Two waves of `OL 2D3D` are fully located in the Ω_{3D} and a very good agreement with the `3D` waves, whereas wave fronts steepen by two `2D` waves. It's worth mentioning that the `OL 2D3D` receding wave front toe overlaps the corresponding `2D` result marked in the red circle at T = 4 [s], which means the `OL 2D3D` receding wave front partially steepens at this moment. This effect further impacts on the `OL 2D3D` receding wave travelling exhibited in figures 5.30 and 5.32.

At T = 5 - 8 [s] in figure 5.30, the three waves were travelling through the overlapping region, where the `OL 2D3D-2D` and the `OL 2D3D-3D` perform fairly different in the overlapping region. The `OL 2D3D-2D` wave remains the bell shape and is situated between the `2D` and the `3D` waves. However, an reflected wave is generated in the `OL 2D3D-3D` and is moving downstream, which is due to the mismatching of flow quantities (flow depth and velocity) at the 2D-3D interface.

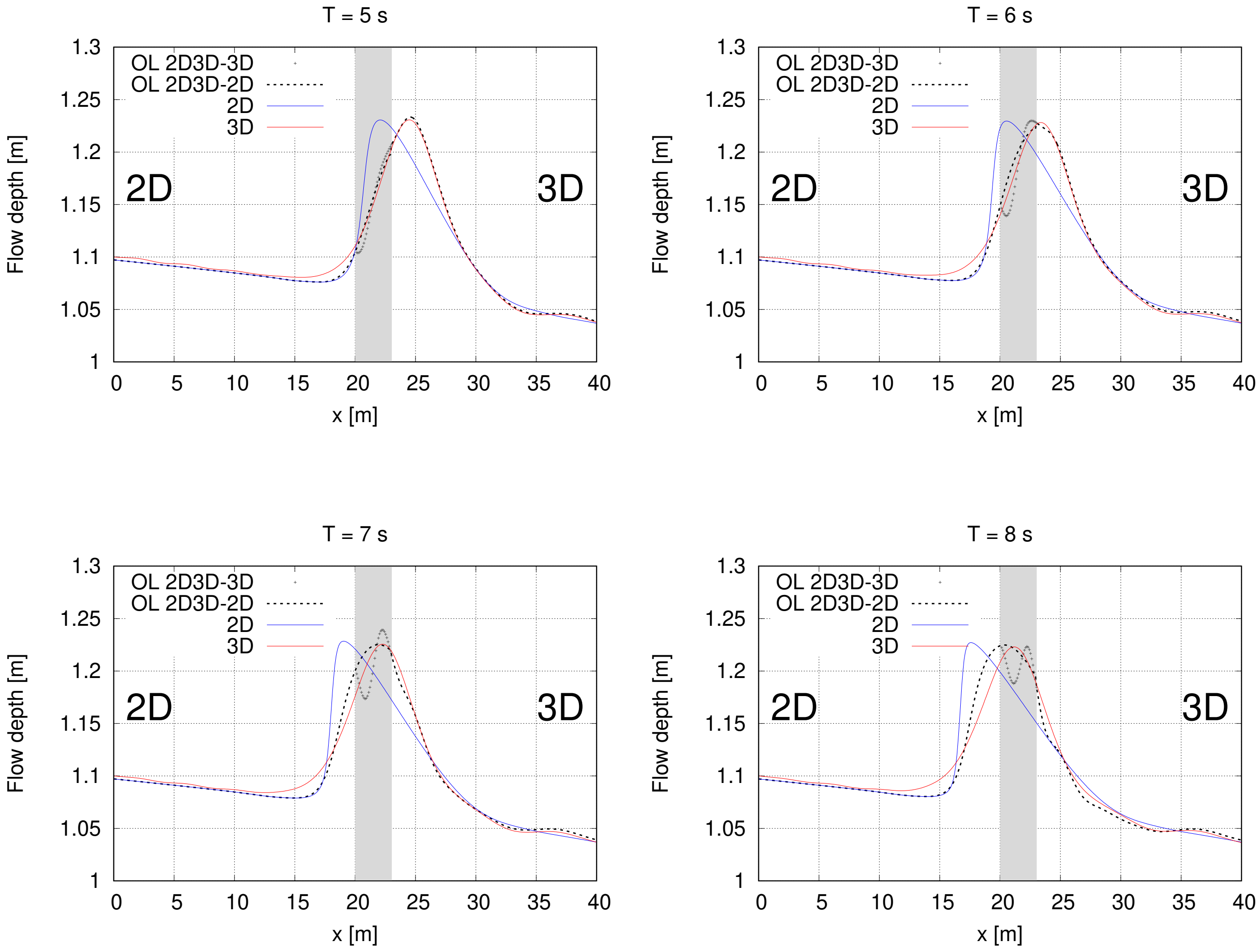


Figure 5.30: Wave travelling at T = 5 - 8 [s]

At T = 5 [s] as exhibited in figure 5.30, as a result of the steepen wave front feature computed by the 2D solver, the 3D receding wave peak locates downstream the overlapping region and the 2D receding wave peak locates inside the overlapping region. The 3D wave height in the overlapping region is thus lower than the 2D. In the meantime, a smaller stream-wise discharge is represented by the 2D in the overlapping region referring to figure 5.31. The higher flow depth and the smaller discharge of the 2D wave in the overlapping region imply a slower transport velocity of the 2D receding wave front. In the coupled model, the slower velocities from the 2D sub-solver results pass to the left overlapping interface and act on the OL 2D3D-3D. An reflected wave in OL 2D3D-3D is formed as a consequence of the wave velocity differences.

Discrepancies of flow quantities between the 2D and the 3D enlarge from the wave front toe to the wave peak and then drop along the wave back. The discrepancy crest occurs at T = 6 - 7 [s], where the 2D wave peak is leaving the overlapping region while the 3D wave peak is entering the overlapping region. The amplitude of reflected wave is hence mitigating after T = 7 [s].

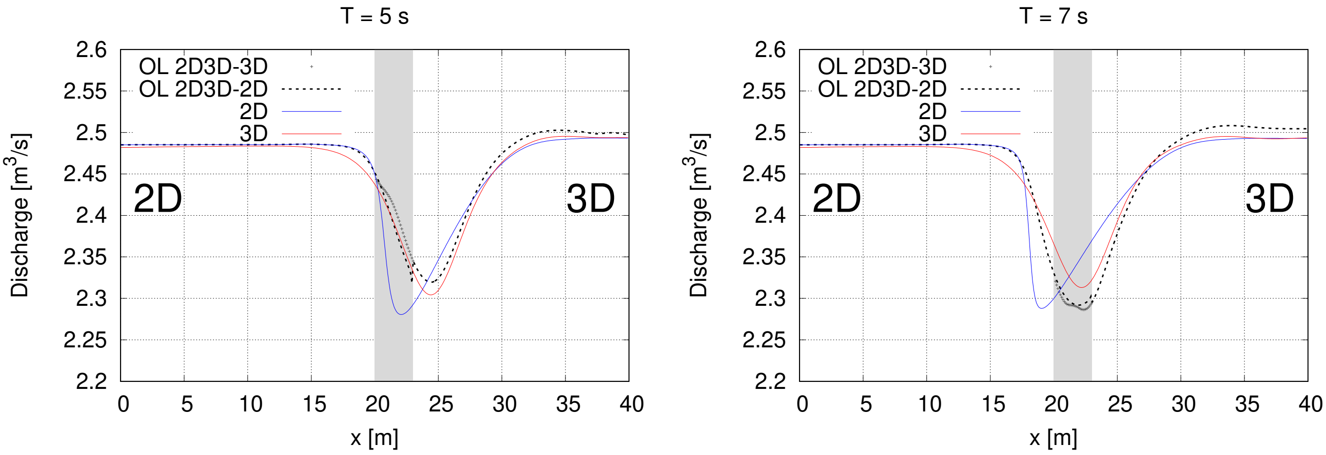


Figure 5.31: Discharge at T = 5 - 7 [s]

At a later time illustrated in figure 5.32, the OL 2D3D wave back positions in the middle of the 2D and the 3D waves and the OL 2D3D wave front steepens as it is governed by the 2D sub-solver. The reflected wave created by the OL 2D3D-3D still remains and moves towards the outlet slowly. The oscillation amplitude of the reflected wave is not more pronounced over time.

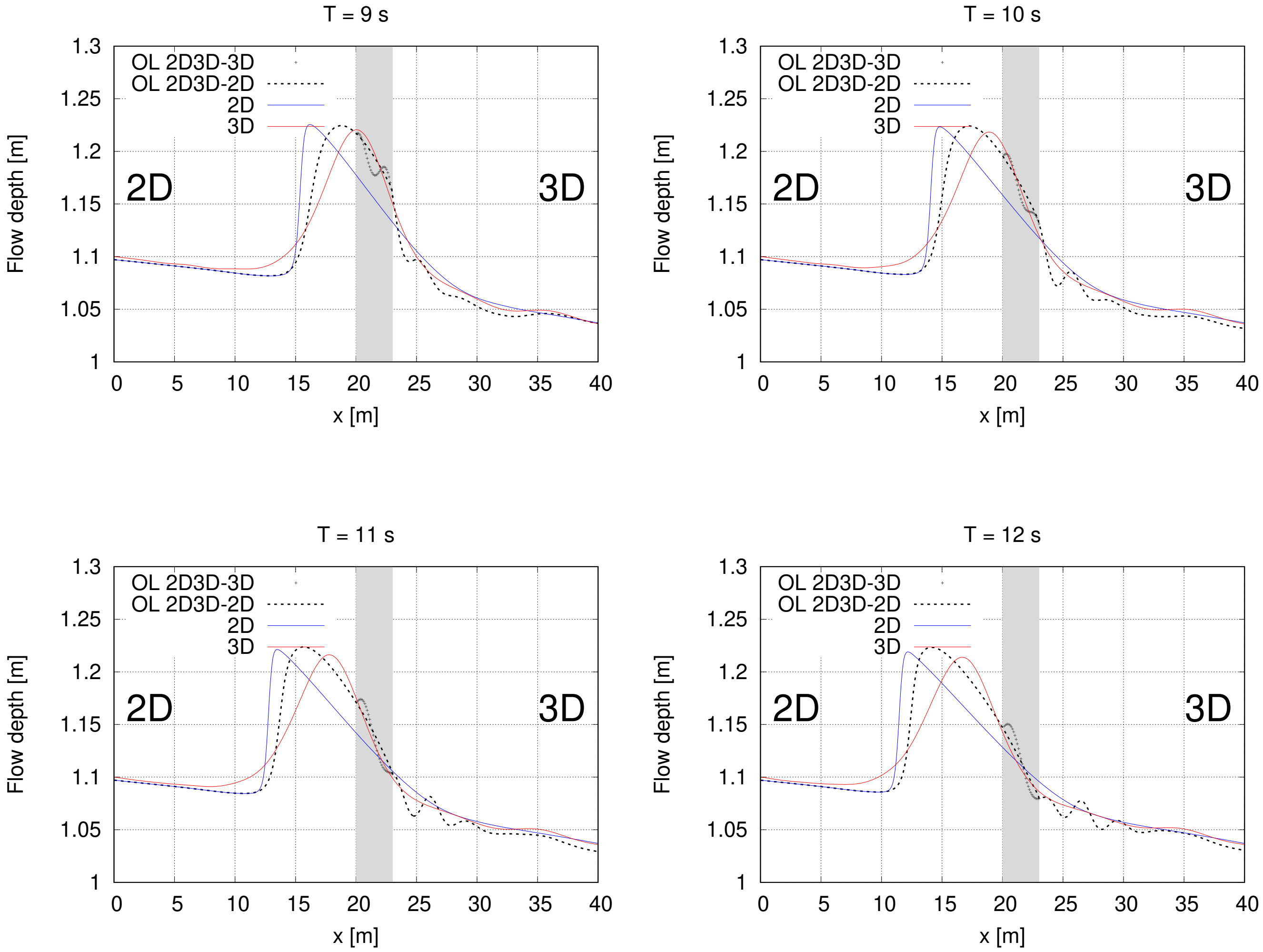


Figure 5.32: Wave travelling at T = 9 - 12 [s]

5.4.2. Results of 3D->2D connection

Again the 3D->2D coupling connection is conducted. Like already introduced in chapter 4, a slightly different boundary condition is implemented with the 3D->2D coupling connection in subcrtical flows. On the Ω_{3D} interface (Γ_{3D}), the Neumann condition is employed to the VOF indicator α_1 whereas the Dirichlet condition is specified for the pressure term p_d by applying the hydrostatic pressure. The water level thus adjusts spontaneously to the corresponding OL 2D3D-2D water level.

The Dirichlet condition implemented in the overlapping coupling algorithm not only fixes the desired value at the boundary (ϕ_Γ) but also to the inner cells adjacent to the boundary (ϕ_{Ω_U}) (referring to figures 4.16 and 4.18), where ϕ stands for the physical quantity. However, when the Neumann condition is applied, the inner cell value is not necessarily consistent with the boundary value. In other words, the 3D velocity at the Γ_{3D} ($u^{3D}_{\Gamma_{3D}}$) definitely equals to the 3D velocity at the inner cell $\Omega_{\cup 3\mathrm{D}}$ ($u^{3\mathrm{D}}_{\Omega_{\cup 3\mathrm{D}}}$), while the VOF indicator α_1 not. More specifically in this case, $u^{3\mathrm{D}}_{\Omega_{\cup 3\mathrm{D}}} = u^{3\mathrm{D}}_{\Gamma_{3\mathrm{D}}} = \frac{hu^{2\mathrm{D}}_{\Omega_{\cup 3\mathrm{D}}}}{h^{2\mathrm{D}}_{\Omega_{\cup 3\mathrm{D}}}}$ and the prescribed 3D pressure term fixes the water level at the $\Gamma_{3\mathrm{D}}$ so that $h^{3\mathrm{D}}_{\Gamma_{3\mathrm{D}}} = h^{2\mathrm{D}}_{\Omega_{\cup 3\mathrm{D}}}$. Due to the fact that $h^{3\mathrm{D}}_{\Omega_{\cup 3\mathrm{D}}} \neq h^{3\mathrm{D}}_{\Gamma_{3\mathrm{D}}}$, the mass conservation is only guaranteed at the $\Gamma_{3\mathrm{D}}$ but not the $\Omega_{\cup 3\mathrm{D}}$. As a result of the subcritical flow condition, the flow depth information transfers upstream and the non-conserved mass will be eventually corrected after a certain time steps.

As shown in figure 5.33, the three receding wave fronts get in the overlapping region at T = 4 [s]. Until this moment, the OL 2D3D-2D waves stand in the $\Omega_{2\mathrm{D}}$ and behave exactly in 2D manners that the black dashed line overlaps the blue line. The OL 2D3D-3D wave front is lightly distorted when it enters the overlapping region and stays in between the 2D and 3D wave fronts. No frequency dispersion appears as long as the coupled waves are dominated by the 2D sub-solver.

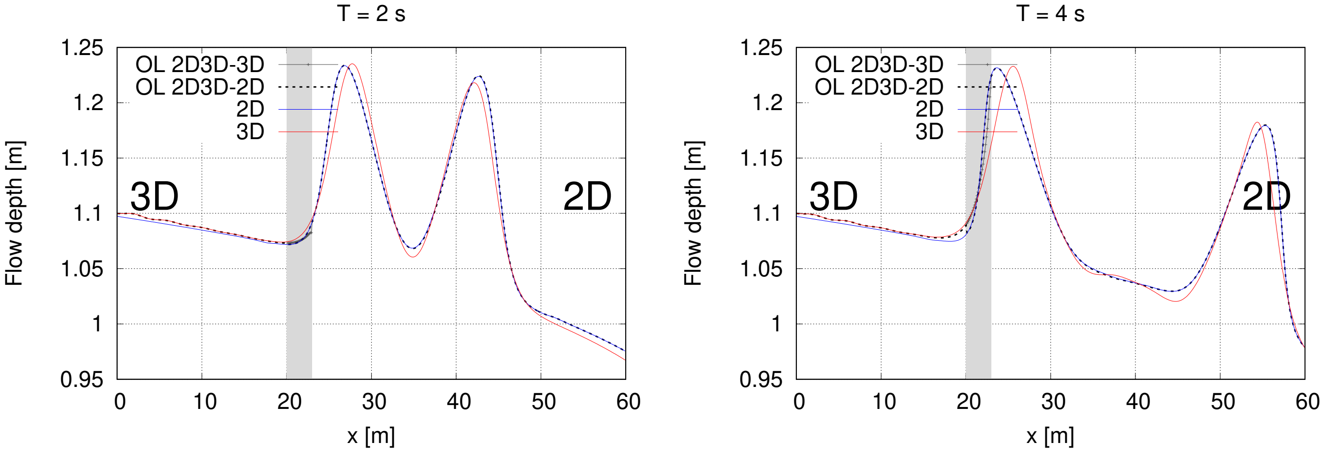


Figure 5.33: Wave travelling at T = 2 - 4 [s]

From T = 6 [s] onwards, a straight drop or lift of the flow depth can be observed by the OL 2D3D-3D at the right overlapping interface implied in figure 5.34, which is caused by a combination of the Neumann condition for the flow depth term and the Dirichlet condition for the velocity term in the 3D sub-solver. Another discontinuity occurs at the left overlapping interface ($\Gamma_{2\mathrm{D}}$), where the OL 2D3D-2D receives the 3D sub-solver's results. These discontinuities are diminished over time. The shape of the OL 2D3D turns to the bell shape as it locates in the $\Omega_{3\mathrm{D}}$. There is no oscillatory wave tail of the OL 2D3D in the $\Omega_{2\mathrm{D}}$.

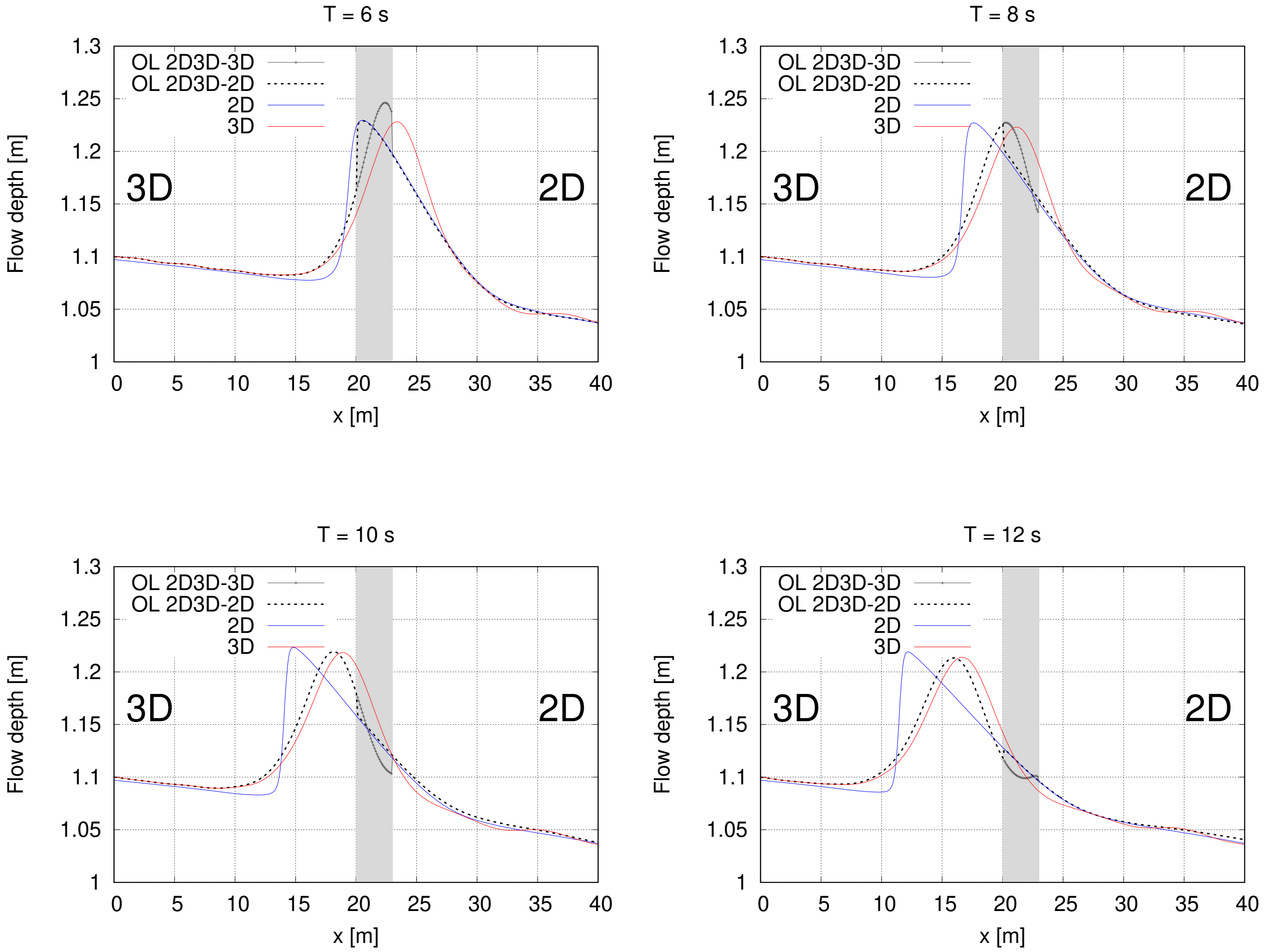


Figure 5.34: Wave travelling at T = 6 - 12 [s]

5.4.3. Summary

The upstream-travelling wave performance passing through the overlapping region in the subcritical flow is introduced by this case. Comparing to the previous test case, two main differences are presented that one is in the geometry and the other one is in the coupled algorithm. A challenge of this case is to establish consistent steady state of subcritical flows by the three models. The first modification is made by adding an overfall at the end of the channel in order to prevent numerical instabilities at the downstream boundary. The second change takes place at the Γ_{3D} using the Neumann condition for α_1 and the Dirichlet condition for p_d. This special treatment shall be employed in the subcritical flow with `3D->2D` connection as explained in section 4.4.

Under both coupling connections, the coupled waves always position in between the two non-coupled solutions. Approximately the same as the previous test case, the `OL 2D3D` advancing waves perfectly match the corresponding pure solver's performance.

With the 2D->3D connection, the frequency dispersion occurs in the Ω_{3D} at T = 4 [s] but not heavily affects the receding wave since the receding wave enters the overlapping region and is governed by the 2D sub-solver from now on. The oscillatory tail illustrated in figure 5.32 is mainly caused by the flow quantities mismatching at two overlapping interfaces during the wave transient period. This occasioned oscillatory tail is more pronounced than the frequency dispersion effect of the 3D simulation. After the receding wave passes through the overlapping region, the wave crest rises to the level of the 2D wave crest.

With the 3D->2D connection, no frequency dispersion effect is detected in the Ω_{2D}, therefore a smooth wave tail is represented downstream of the receding wave. When the wave enters the overlapping region, a strong overshoot appears in the OL 2D3D-3D and a considerable undershoot occurs in the OL 2D3D-2D. These discontinuities are occasioned by the change in flow dynamics (Mintgen (2017)).

5.5. Conclusions

This chapter conducts three cases with respect to demonstrate the overlapping coupling algorithm properties. With exchanging flow quantities between cells value not the faces', a stronger coupling of two systems is conducted. This strong explicit coupling algorithm forces every information to transfer from one region to the other. The first case, the upstream-travelling surge, is also applied by other coupling methods described in chapter 4.

The three flow scenarios simulate three wave propagation phenomena in plane channels, where a steeper channel slope of 0.005 is employed in supercritical flows development and a milder channel slope of 0.001 is used in subcritical flows development. The overlapping region locating X = 50 - 53 [m] is taken in the three cases as a standard set-up. Surge transfer with higher flow velocities requires higher mesh resolution, hence the moving surge case utilise a finer grid spacing of 0.05 [m] and the wave transportation cases build a coarser mesh with the grid spacing of 0.1 [m]. Both 2D->3D and 3D->2D coupling connections are constructed.

These cases mainly test the information transfer in different directions of a sudden or gradual change in flows. In upstream-travelling surge case, the change of flow information transfer due to a shock flow condition change can be detected by the explicit exchange flow properties in the overlapping area. The 3D simulations reproduce well the shock surge shape and the velocities, while steepened smooth wave fronts and wavy wave surfaces are presented by the 2D results. With this respect, the coupled results give an overall consistent outcome where the wave forms an accurate shape in the Ω_{3D}; and it steepens in Ω_{2D}. However, the OL 2D3D causes disturbances during the surge transfer through the overlapping region phase, this effect is further analysed by a sequence of sensitive studies. The investigation of sensitive study reports that the OL 2D3D simulations are not apparently influenced in terms of the maximum Courant number. Making the overlapping region smaller gives smaller disturbances. The most important uncertainty evidently affecting the OL 2D3D is the flow property.

The stronger variation occurs in the flow, the more violent disturbance is engendered during the overlapping region transfer phase. Nevertheless, disturbances are introduced during the wave transition through the overlapping region and are weakened as wave propagates. In Gaussian wave transportation scenarios, similar observations are obtained but the smoother transfer processes are reproduced since the change in flow properties are rather small.

In conclusion, a feasibility study of the overlapping coupling algorithm is validated by demonstrating a preserved flow behaviors in both the shape and the amplitude of the coupled wave. The present coupling gives stable and accurate results for steady cases. Disturbances are produced in some extreme flow conditions as the wave shapes and characteristics are so different between 2D and 3D. An improved simulation can be obtained by shrinking the overlap region to at least two grids in each direction. For future coupling algorithm improvement in wave transport problems, a numerical scheme that brings the 2D wave closer to the 3D would be of great interests.

6. Numerical Modelling of the Bore-Structure Interaction

From a practical perspective, hydraulic processes modelling in natural disasters, such as floods, hurricanes and tsunamis, become critical of the overlapping coupling algorithm. These natural disasters typically crop up in one regional area and dramatically influence the surrounding geometry. The present algorithm developed in this work more accurately models extreme hydrodynamic events than the pure 2D Shallow Water solver and requires less computational time with comparable accuracy to the pure 3D RANS solver. The 3D sub-solver governs the demanding vertical dimension regions and the 2D sub-solver comprises the shallow water regions.

This chapter conducts numerical simulations on an extreme hydrodynamic scenario: tsunami bore - structure interactions. The scenario was initially built up in an experimental program by the University of Ottawa and the Canadian Hydraulic Center (CHC) of the Natural Research Council (NRC) of Canada. We replicate the experiment numerically to analyze and quantify the overlapping coupling algorithm performance of tsunami-induced bore generation and extreme hydrodynamic forces on structure. Numerical results will be compared with experimental measurements.

Motivation of extreme hydrodynamic events modelling will be shortly introduced in section 6.1. Experimental and numerical set-up will be detailed in sections 6.2 and 6.3. Results will be demonstrated and discussed in sections 6.4 and 6.5. Summary will be given in section 6.6.

6.1. Motivation

Many cities are situated on shorelines and are thus vulnerable to flooding, especially in inundation high risky areas. Tsunami inundation and river flooding have physical processes in common. A tsunami is one of the deadliest and most economically damaging natural disasters in the world. The last decade witnessed several devastating tsunamis, particularly in the Pacific Ocean whose bordering seas are the most severely affected. For example, the 2004 Indian Ocean tsunami caused 230,000 deaths and significant economic losses, while in 2011 a tsunami killed nearly 20,000 people and caused 360 bn. US dollars in total damage in northeastern Japan. Due to such events, more and more research is being devoted to tsunamis.

A tsunami is a series of waves triggered by the displacement of a large mass of water (*e.g.* earthquakes, volcanic eruptions and other underwater explosions). Such waves are less harmful when they are far from the shoreline. They travel in the form of long waves characterized by fairly low wave heights and long wave lengths, which make them difficult to detect offshore. When they approach inland, sea floor elevation increases and water depth decreases leading to a growing wave height of the tsunami waves. In the wave shoaling process, the incoming tsunami waves are transformed into the tsunami-induced bores, also known as hydraulic bores.

Destructive effects of tsunami-induced bore can be divided into three segments: (1) the onshore run-up behavoir of tsunami bore inundates coastal communities with high speed; (2) bore-structure interaction forces destroy objects in bores' path; (3) debris from damaged infrastructures transported by the bore further cause considerable deaths and injuries. Accordingly, damages can be attributed to: (1) flooding; (2) Structural destroy, scour and slope / foundation failure; (3) second structural damage, fire spread by floating debris and combustible liquids. Each of these effects, alone or in combination brings sever damage.

To mitigate tsunami impact, building tsunami-resilient structures, with intention to trip over the bore and make it break, is the main protection against tsunami. The complexity of the bore-structure interaction arouses awareness of coastal engineers. Investigations on tsunami-like hydraulic bores impacting structures will significantly help in tsunami-resilient structures design.

6.2. Case description

To identify tsunami-induced effects on structures, the University of Ottawa and the Canadian Hydraulic Center (CHC) of the Natural Research Council (NRC) of Canada have conducted comprehensive experiments to determine the forces structures are exposed to by turbulent bores (Palermo et al. (2012)). Their experimental program will be compared with numerical simulated results from the new coupling algorithm, the overlapping coupling method.

The experiment reproduced tsunami-induced bores by subjecting a solid structure to dam break flows in a flat rectangular open channel flume. Two square glass windows inserted on the flume side enable the visualisation of the bore-structure interaction. Such flows can produce the desired tsunami bores (Chanson (2005)). The dam break phenomenon resulting from the instantaneous collapse of a dam, is generated by a sudden opening of the swinging gate which releases the stored water. A square hollow structural cylinder made by acrylic sheets established downstream from the swinging gate triggers the hydraulic bore. Thirteen wave gauges distributed in the flume measure water depths over time, five of which were mounted on the structural element. These gauges were used to record the bore depth-time histories along the flume and around the cylinder. In addition, a six degree-of-free (6 DOF) high-frequency dynamometer (MC6 Series) bolted to the flume floor to record hydraulic-induced forces imposed on the structure.

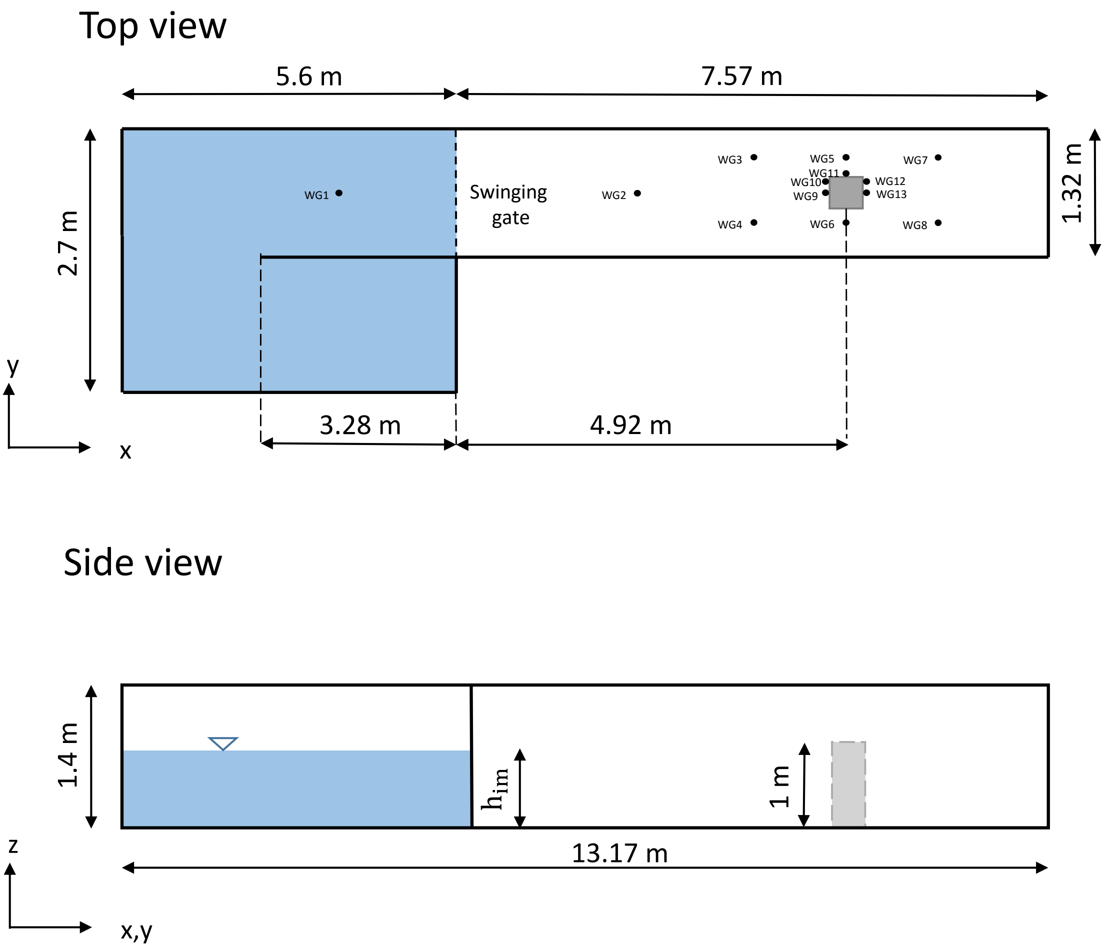


Figure 6.1: Computational domain with 13 wave gauges location

The physical experiment flume was reconstructed based on a U-turn shape high discharge flume at the CHC by adding a swinging hinged gate 5.6 [m] downstream of the upstream end. The impounding water is kept in the U-turn zone, which is divided by a 3.28 [m] long stainless steel-constructed flume, to generate the dam break flow. A sluice gate was placed at the downstream end of the flume and combined with a vertical drain, which functioned as the water level controller to regulate the water level during the re-zeroing of the water level gauges. The structural element was then installed 4.92 [m] further downstream of the swinging gate, where the tsunami bore is fully developed and the glass windows are located. To simulate a tsunami, this set-up was also applied to the numerical model in figure 6.1 as well as the wave gauges. Apart from the sluice gate, a free outlet boundary condition is incorporated in the numerical simulations. The entire flume is 13.17 [m] long and 1.4 [m] high. Due to the U-turn shape reservoir, the upstream boundary of the flume is about twice as wide the downstream boundary; consequently, the inlet patch width is 2.7 [m] and the outlet patch width is 1.32 [m]. The downstream structure has a 0.305 × 0.305 [m^2] cross section and a height of 1 [m]. Three impounding water depths (h_{im}) of 550 [mm], 850 [mm] and 1150 [mm] were tested to provide a better understanding of the impact of hydraulic bores on the square structure. For convenience, the location of the model center is denoted as WG 0. As illustrated in figure 6.1, one wave gauge was placed in the reservoir, seven wave gauges distributed along the dry area and the remaining five were installed on the face of the structure perpendicular to the bottom. Wave gauge location coordinates are given in table 6.1.

	WG 0	WG 1	WG 2	WG 3	WG 4	WG 5	WG 6
X [m]	10.52	4.35	8.37	9.52	9.52	10.52	10.52
Y [m]	0.66	0.66	0.66	0.88	0.44	0.88	0.44
	WG 7	WG 8	WG 9	WG 10	WG 11	WG 12	WG 13
X [m]	11.52	11.52	10.3675	10.3675	10.52	10.6725	10.6725
Y [m]	0.88	0.44	0.66	0.805	0.815	0.805	0.66

Table 6.1: Wave gauge location coordinates

6.3. Numerical set-up

Detailed simulation of the forward and backward wave fronts requires 3D models taking into account vertical fluid accelerations. In order to examine the impact of hydraulic bores on structures, the fully 3D Navier-Stokes solutions are demanded near the square structure. In the coupling simulation the Ω_{3D} ranges from 7 to 13 [m] in the stream-wise direction is selected and thus splits the computational domain into three sub-regions, leading to a `2D->3D->2D` coupling connection. The structure is therefore completely embedded in the Ω_{3D}. The overlapping distance is 1 [m] long on both sides of the Ω_{3D} as shown in figure 6.2. Again, the present model is simulated using the pure `2D` solver - `shallowFoam`, the pure `3D` solver - `interFoam` and the `OL 2D3D` solver - `shallowInterFoamOL`.

We determined the mesh resolution in line with Douglas and Nistor (2014), who reproduced numerical investigations on experiments mentioned before using the 3D OpenFOAM solver - `interFoam`. They conducted a sensitive study on mesh resolution and concluded that sufficient details (*e.g.* the splash at initial impact) in the column region can be produced when the single cell dimension is smaller than $2 \times 2 \times 2$ [cm^3]. Applying such a fine mesh resolution requires a huge computational cost in the pure `3D` and the `OL 2D3D` models. Unfortunately, the coupled model has not been yet parallel computing functioned, so a coarser mesh resolution of 4×4 [cm^2] in horizontal directions was selected. For the 3D mesh, the vertical grid spacing remained as 2 [cm]. The purpose of this case therefore is mainly to compare the `OL 2D3D` and the `3D` behavors in the structure region.

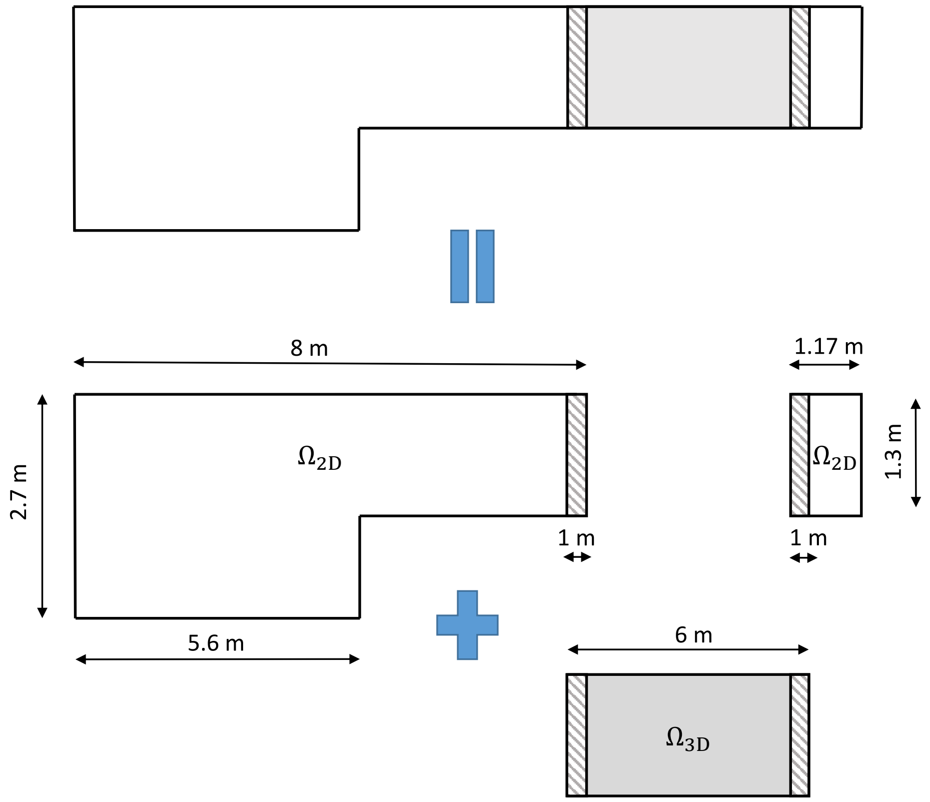


Figure 6.2: Domain decomposition in the coupling simulations

Initial impounding water h_{im} is prescribed in the upstream reservoir region and no other water source is supplied. In the remaining channel section a dry bed condition is applied. The initial status of the simulation is given in figure 6.3.

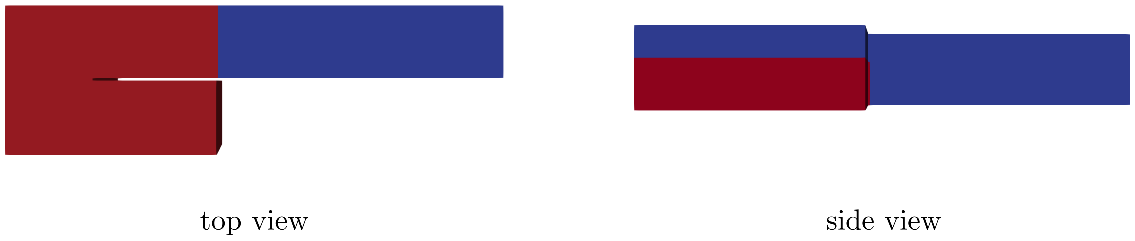


Figure 6.3: Initial status of the water in numerical set-up

All parameters used in computations are given in table 6.2. The time step Δt is automatically adjusted according to a prescribed maximum local Courant number CFL_{max} 0.5. The Strickler coefficient for the 2D domain is 25 $[\text{m}^{1/3} \cdot \text{s}^{-1}]$ and the sand roughness for the 3D domain is 0.001 [m]. The dam break problem is characterized by three-dimensional aspects. Based on 2D Shallow Water equations assumptions, vertical velocities and accelerations are negligible. No driven force in the vertical direction is presented in the 2D model to commence the first motion immediately after the dam break. If we initialize the channel with a zero discharge in the 2D model, the flow will remain in the reservoir and not flow into the channel. The 2D model is therefore initialised with a small discharge of 0.01 [$\text{m}^2 \cdot \text{s}^{-1}$] over the channel.

Parameters	Values
$\Delta X = \Delta Y$ [m]	0.04
$\Delta Z^{3\text{D}}$ [m]	0.02
Δt [s]	$\text{CFL}_{\text{max}} = 0.5$
h_{im} [m]	0.55 / 0.85 / 1.15
I_{s}	0
k_{st} [$\text{m}^{1/3} \cdot \text{s}^{-1}$]	25
k_{s} [m]	0.001
$hu^{2\text{D}}$ [$\text{m}^2 \cdot \text{s}^{-1}$]	0.01

Table 6.2: Numerical parameters

6.4. h_{im} = 850 [mm]

The numerical model was firstly validated using $h_{\text{im}} = 850$ [mm] by a flow profile comparison with the experimentally monitored results in the absence of the structural model. A dam break induced hydraulic bore was developed. After that, the model was introduced to generate a tidal bore moving upstream. The flow characteristics of the bore depth time-history, flow profile were then analysed.

6.4.1. With no square structure

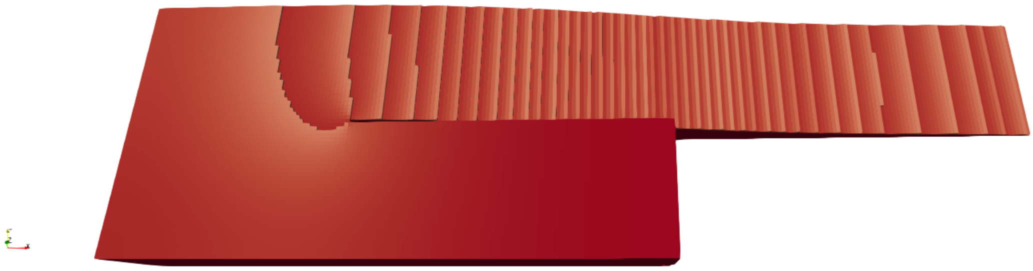


Figure 6.4: Free surface representation at T = 1 [s] with h_{im} = 850 [mm]

A screen shot of the free surface at T = 1 [s] by 3D is reproduced in figure 6.4. The impounding water is released by opening the swinging gate and rushes downstream. Free surfaces recorded by four wave gauges (see figure 6.5) placed along the flume to represent the dam break flow propagation process. WG 1 was placed in the reservoir to adjust the initial impounding water height, WG 2 was about 3 [m] behind the swinging gate to capture the near dam break flow profile, WG 3 was around 1 [m] further downstream from the WG 2 to measure the flow feature before the structural location and WG 0 was situated at the structural model central. Accordingly, for the coupling simulation three wave gauges are in Ω_{3D}, whereas the WG 1 is located in the upstream Ω_{2D}.

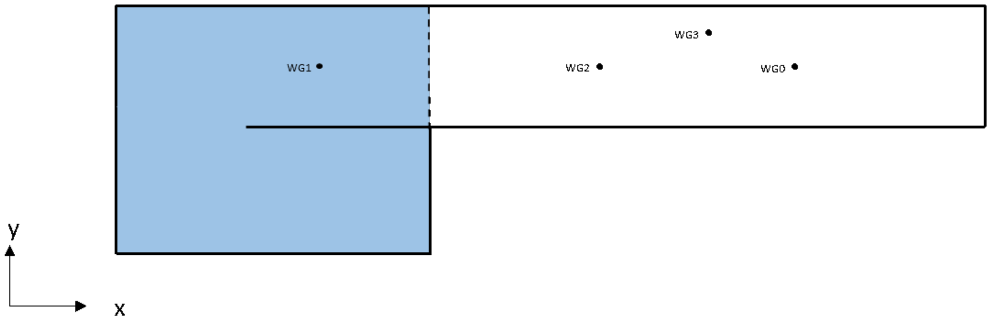


Figure 6.5: Set-up without structural model in the flume with locations of four waves gauges

At an early stage of the simulation, the impounding water front begins to collapse due to gravity and propagates downstream as displayed in figure 6.6. Comparison of the initial water block deformation, under gravity at T = 0.1 [s], with three numerical models is pictured in the left figure 6.6. The 3D water front propagates with advancing toe and declining top, while the 2D result remains unchanged. 2D solver has difficulties in accurately modelling the bore immediately after the gate opening since high free surface gradients are present in the dam failure, while the vertical acceleration is ignored by the 2D solver. At 1 [s], the figure on the right, the 2D front runs to X = 9 [m] and the 3D front position is close to X = 10 [m], indicating a greater wave celerity computed by the 3D solver than the 2D solution.

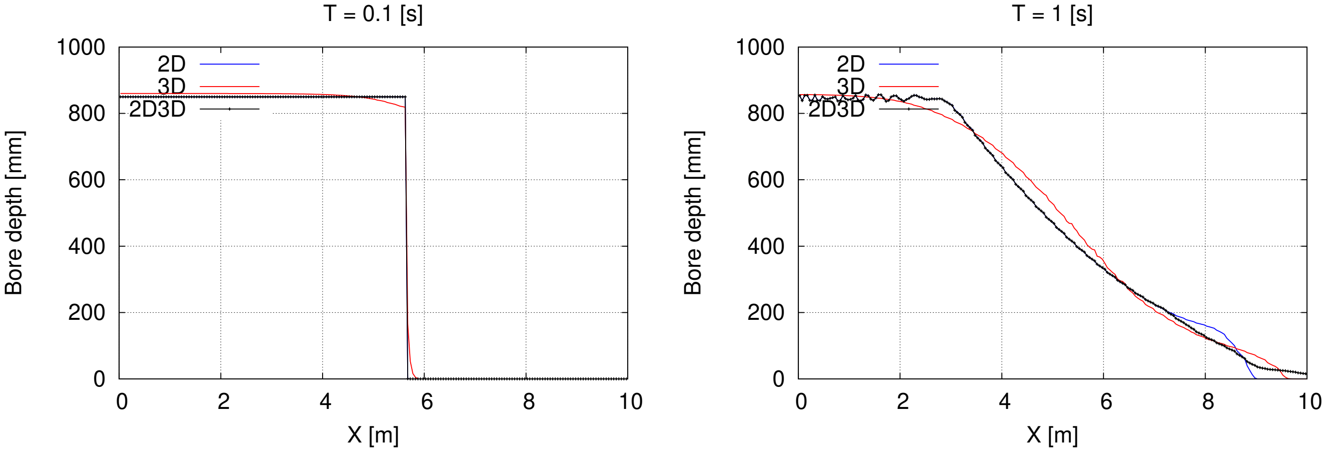


Figure 6.6: Free surface along the flume at T = 0.1 and 1 [s] with $h_{\text{im}} = 850$ [mm]

During the dam collapse, the OL 2D3D profile completely overlaps the 2D outcome in Ω_{2D}

(*i.e.* X = 0 - 7 [m]) and converges to the `3D` in Ω_{3D} (*i.e.* X = 7 - 10 [m]). A steeper flow front (*i.e.* X = 3 - 6 [m]) forces a higher flow velocity of the `OL 2D3D` as an input to Ω_{3D}, thus the `OL 2D3D` runs further than the `3D` results.

A free surface comparison at different times after the dam failure between experimental (refer to Al-Faesly et al. (2012)) and numerical results for several wave gauges with no square structure in the channel is shown in figure 6.7. WG 1 was placed upstream the swinging gate in order to record the depth change in the reservoir. During the first half second of the motion, water surface levels observed with the RANS model at WG 1 are lower than those predicted by the shallow water equations. As mentioned above, these discrepancies are mainly due to different initial accelerations, resulting in a time lag in the `2D` flow. The `3D` model is in general agreement with the experimental data reproducing real behaviors. The `3D` flows rush out of the swinging gate faster than the measurement, which can be attributed to the fact that gate opening in real time is not instantaneous.

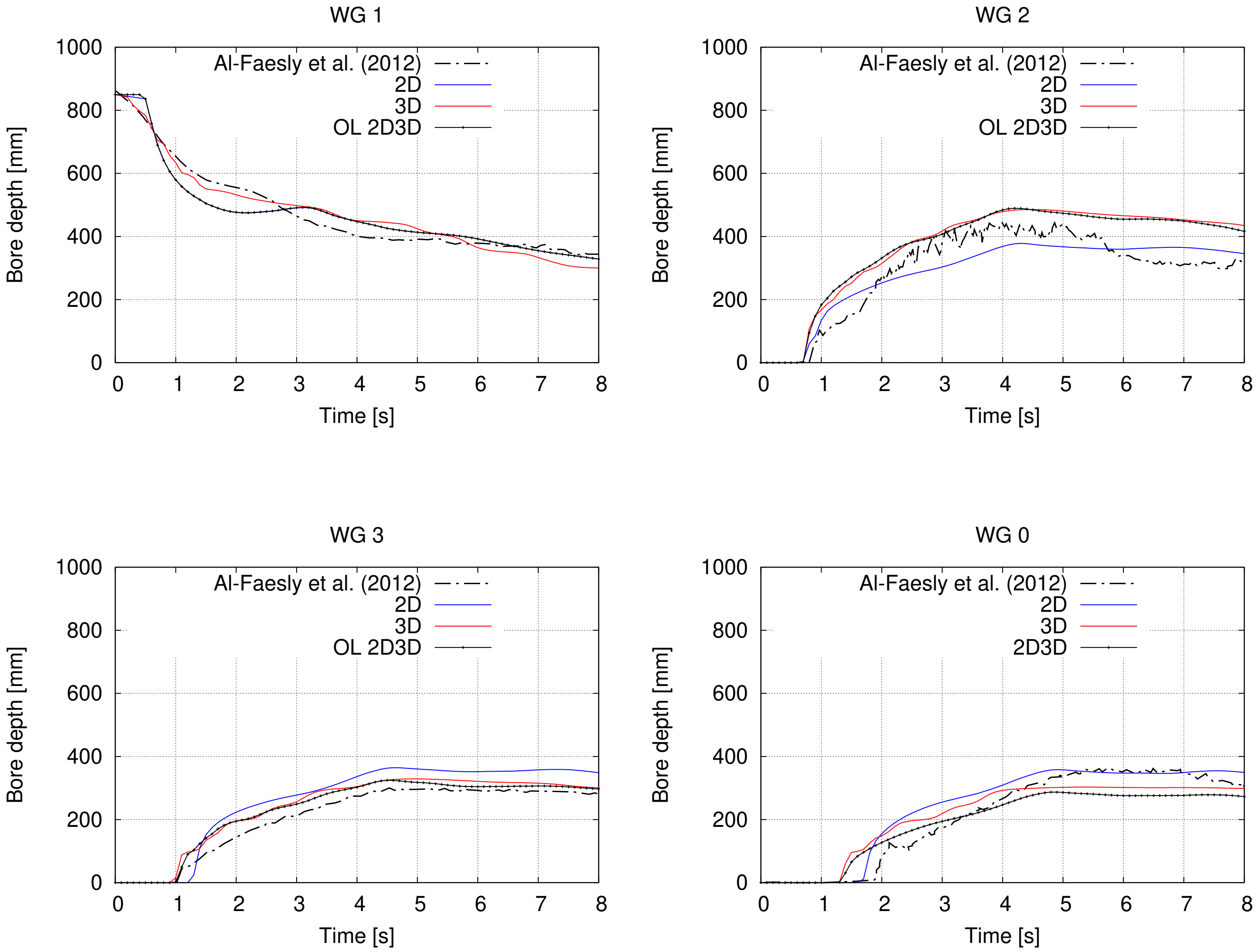


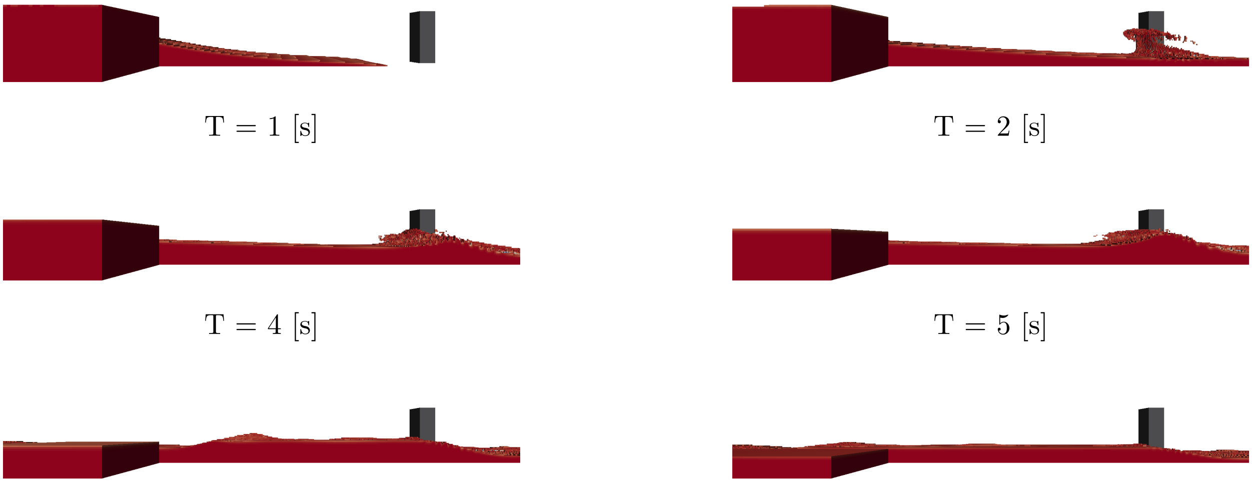
Figure 6.7: Bore depth time series at different WG with $h_{\mathrm{im}} = 850$ [mm]

WG 2, WG 3 and WG 0 are situated downstream of the dam break site in Ω_{3D}, where similar behaviors are represented by the OL 2D3D and 3D models. The bore depth time-history profile, at the structure location (WG0) for instance, starts with a sharp steep slope and then the slope moderates as the bore depth increases until it reaches the maximum, which is 30 - 40% of the impounding water depth. The steady state is attained after a slight reduction of the maximum bore depth. The aforementioned slight delay in the measurement further affects subsequent wave gauges (*e.g.* WG 1, WG 2, WG 0). As the flow propagates downstream, the time lag between the experimental and numerical, in terms of the front arriving time, becomes greater. Steady state flow depths computed by 3D models (*i.e.* OL 2D3D and 3D) are marginally overestimated at WG 2, well matched at WG 3 and underestimated at WG 0, compared to the experimental results. A rather stable bore depth of 350 [mm] is reached at each wave gauge by the 2D solutions.

A general agreement appears to exist between the experimental and the numerical bore depth time-history profile. 3D model reproduces more accurate bore depth representations than a 2D simulation if we take the measurement as evidence. The performance of the OL 2D3D model coincides with the fact that free surfaces at WG 1 fit well with the 2D model and the approximately identical depth profile with the 3D model is computed at the rest four wave gauges. In conclusion, the present numerical model set-up performs steady and accurately.

6.4.2. With the square structure

After calibrating the model set-up, we reran the three models, introducing the square structure at the designated position, where the structure center is situated at X = 10.52 [m] in the span-wise middle of the flume. Non-slip conditions were prescribed at the body surfaces.



T = 1 [s]

T = 2 [s]

T = 4 [s]

T = 5 [s]

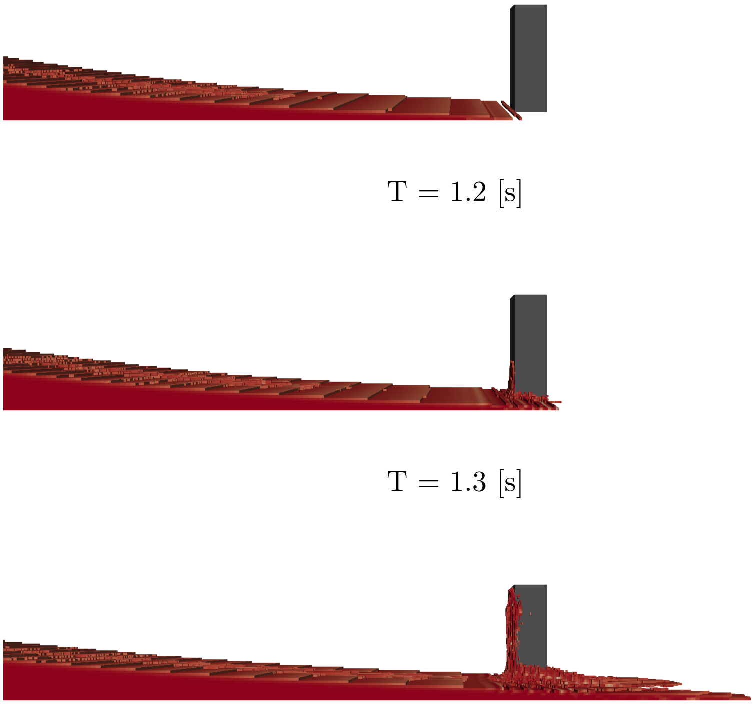
T = 10 [s]

T = 15 [s]

Figure 6.8: Global view of free surface screen shot with $h_{\mathrm{im}} = 850$ [mm] by the 3D model

Figure 6.8 represents a global view of free surfaces modelled by the 3D at different time stamps. At T = 1 [s], a dam break flow profile is generated, and its front approaches the square structure. The maximum water level in the reservoir stays the same. One second later, the flow front hit the structure creating a strong turbulent jump as well as a mass of splashed water drops around the model. The resulting turbulent jump at the front face of the structure runs back against the incoming flow at T = 4 and 5 [s]. At a later point in time, the back wave transfers continuously upstream, while its energy and amplitude gradually decay. At T = 15 [s], the water level in the reservoir apparently falls off and the flow condition tends to become steady where the upstream flow of the structure is subcritical, the flow at the model is critical and the flow in downstream of the model is supercritical.

The flow behavior around the structural model during the hydraulic bore impact is detailed in figures 6.9 and 6.10. Before the bore hits the obstacle, the stream-wise velocity dominates. The incoming flow front touches the model at 1.2 [s] and climbs up the model at T = 1.3 [s]. At T = 1.6 s, the initial run-up tongue reaches its peak elevation and begins to collapse onto the incoming flow due to the gravity at T = 1.8 [s]. The bypassing model flow runs downstream with small turbulent side waves. From T = 2.1 - 2.4 [s], a huge wave is generated and the surface roller forms a steep front. The cylinder is high enough that there is no vertical flow across the top of the cylinder.

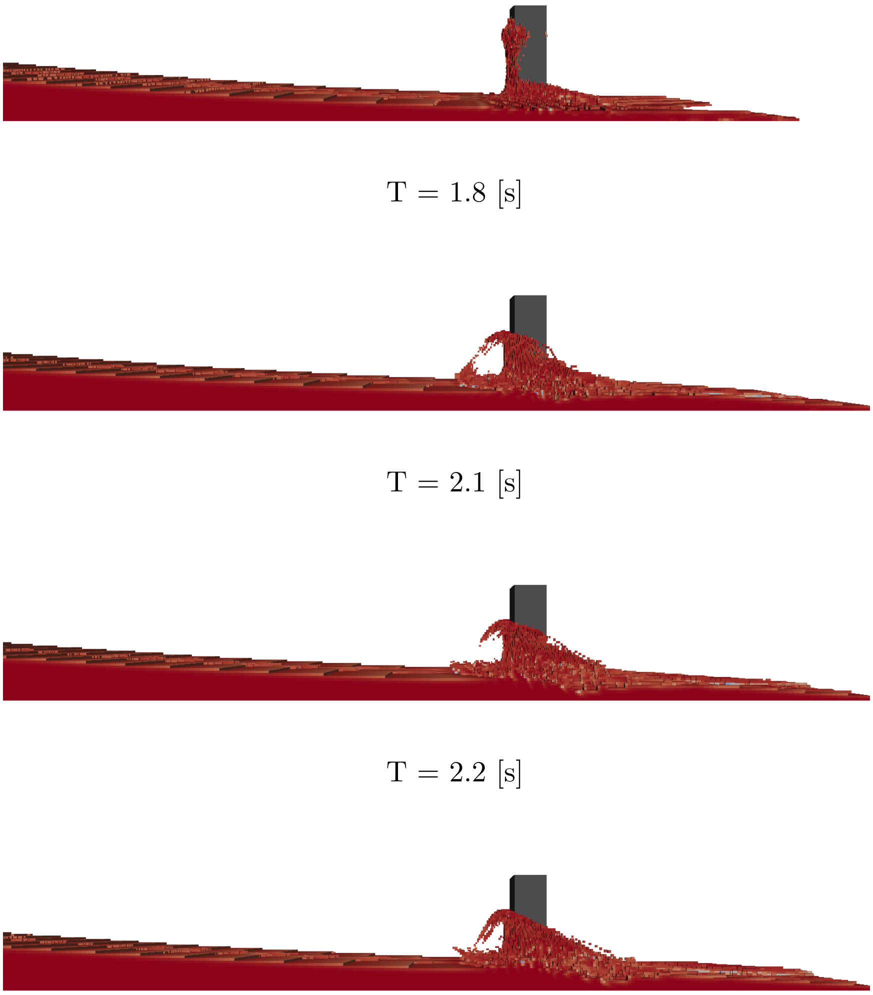


T = 1.2 [s]

T = 1.3 [s]

T = 1.6 [s]

Figure 6.9: Local view of free surface screen shot at T = 1.2 - 1.6 [s] with h_{im} = 850 [mm] by the 3D model



T = 1.8 [s]

T = 2.1 [s]

T = 2.2 [s]

T = 2.4 [s]

Figure 6.10: Local view of free surface screen shot at T = 1.8 - 2.4 [s] with h_{im} = 850 [mm] by the 3D model

Free surface screen shots above are obtained from the pure 3D model. We compare the bore travelling process together by the three models. The process of dam break induced bores transferring through the overlapping region is displayed in figure 6.11. The overlapping region is marked in grey (X = 7 - 8 [m]), and results from both 2D and 3D sub-solvers in the coupled model are illustrated in the overlapping region. Similar behaviors can be observed as the results from non-structure presence simulations (see figure 6.6). The coupled bore front always stays between the two non-coupled bore fronts and it passes the overlapping region smoothly.

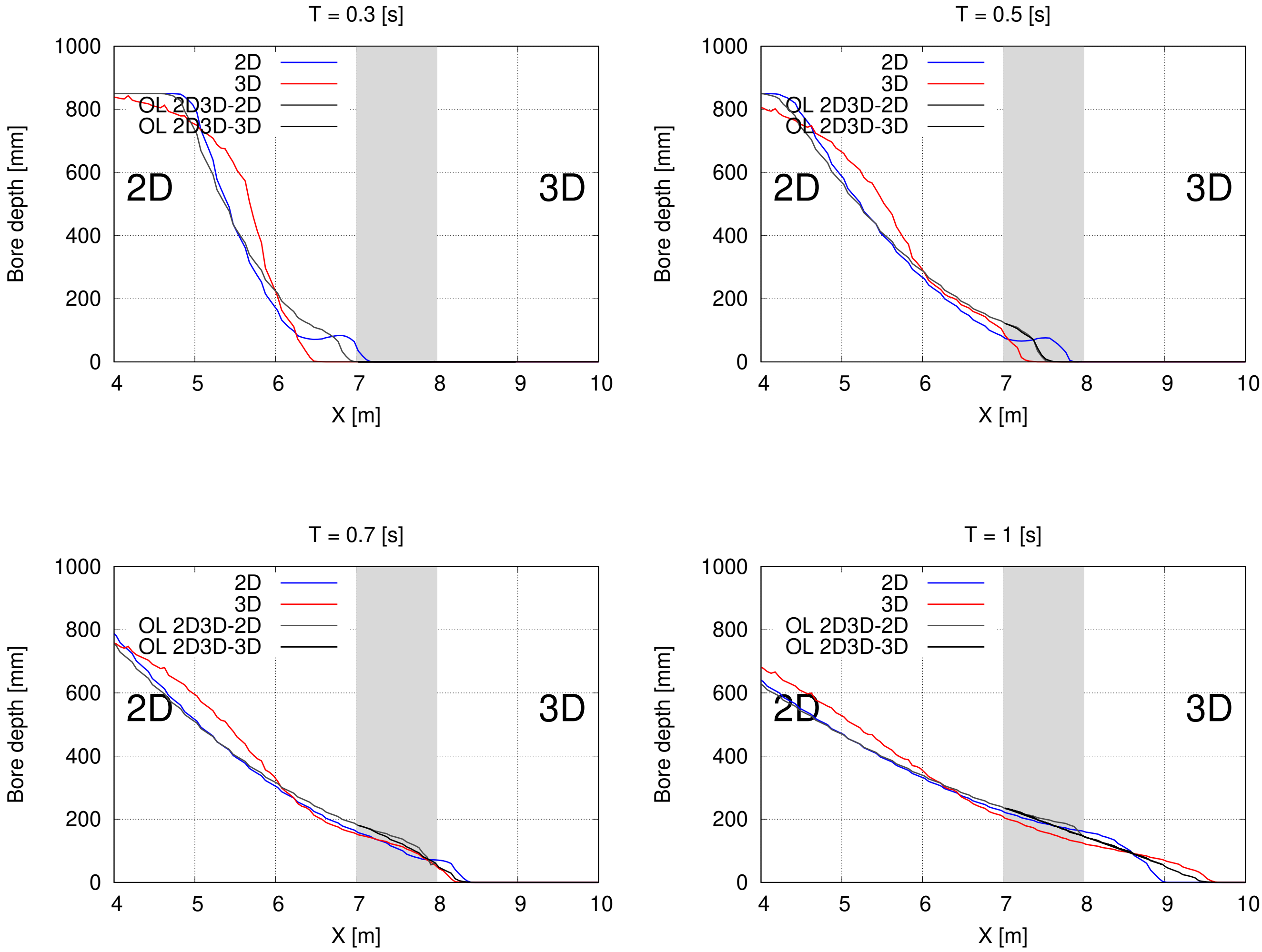


Figure 6.11: Water depths during passage of the dam break flow over the overlapping region with $h_{\text{im}} = 850$ [mm]

The upstream-travelling wave is engendered by the bore-structure interaction. From figure 6.12, the wave amplitude decreases and the wave front becomes smooth as it moves upstream. The 2D wave is far advanced than the coupled and the 3D model in upstream direction. The coupled wave shape gets transformed from a 3D-like wave into a 2D-like wave during the period shown in figure 6.12. In the meanwhile, the coupled wave front proceeds faster when it enters the 2D sub-region. Disturbances occur in the coupled model at T = 9.5 - 10 [s] and diminish at T = 10.5 - 11 [s].

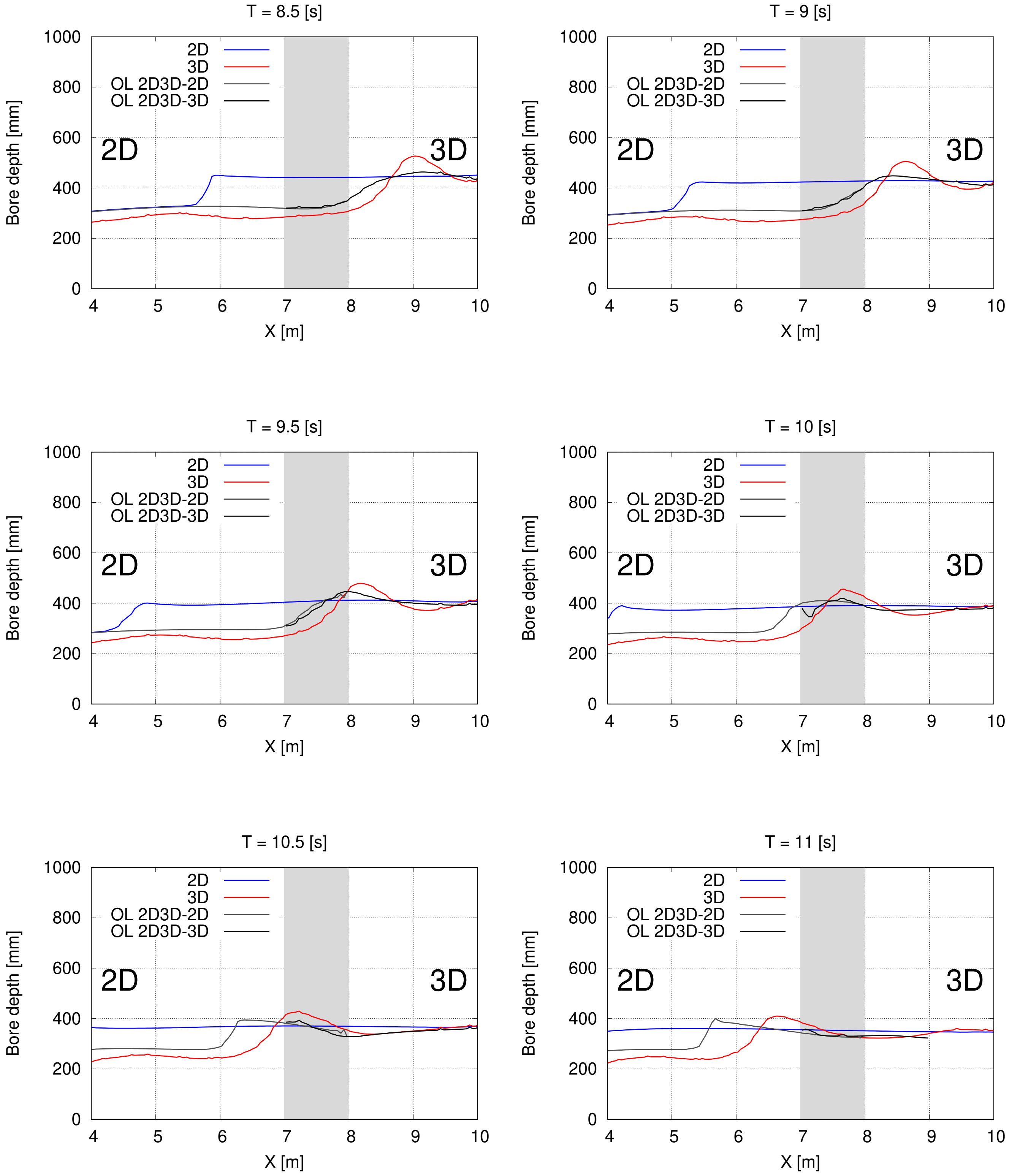


Figure 6.12: Bore-structure interaction induced jump transfer through the overlapping region at T = 8.5 - 11 [s] with h_{im} = 850 [mm]

Comparison between the experiments and the simulations are made around the model. Five peculiar wave gauges are installed along the model surfaces, of which two gauges are placed at the front faces, another two are attached on the back faces, and one gauges is established at the left side face of the model. As illustrated in figure 6.13, the face center of each three faces is equipped with one gauge, moreover, the edge of the front and the back faces is additionally settled one gauges. In this area, the coupling simulation is governed by the 3D sub-solver.

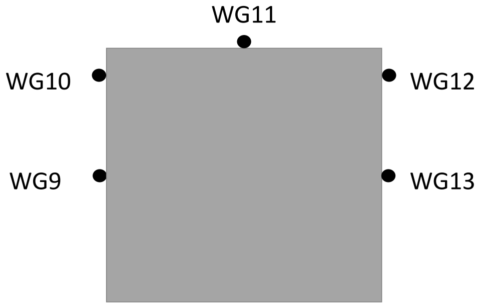


Figure 6.13: Locations of wave gauges at the structural model

The time-history bore depths among these methods at WG 9 and 10 in figure 6.14 imply an overall good accordance. The maximum bore depth occurs at WG 9 placed on the front face center of the model and a tiny bit lower bore peak is observed at its neighbouring gauge WG 10. The hydraulic bore peak recoded at WG 9 from the measurement, the OL 2D3D and the 3D are 870, 840 and 910 [mm] in each instance, where the OL 2D3D slightly underestimates the peak while the 3D overestimates it with the same order of magnitude. The 2D solver however has difficulties in capturing the abrupt increase in bore depth resulting in a rather smooth wave peak. Comparatively good match of the hydraulic bore peaks between the measurement data and the two simulated results (*i.e.* the OL 2D3D and the 3D) revealing a better shock wave representation in the three dimensional solver. The bore peak arriving time of the measurement and the simulations are in a consistency manner as demonstrated in figure 6.7 at WG 0 in the no model present case, where the OL 2D3D and the 3D bores are ahead of the 2D and the measurement. Due to this time-lead in the OL 2D3D and the 3D, their upstream moving waves transfer further from the cylinder and more drained flow are generated, therefore a shallower bore depth than the measurement is represented in the time-history series after 8 [s]. Similar behaviors are observed at WG 10, whereas the 2D reproduces even severe underestimated bore depth during the bore impact phase.

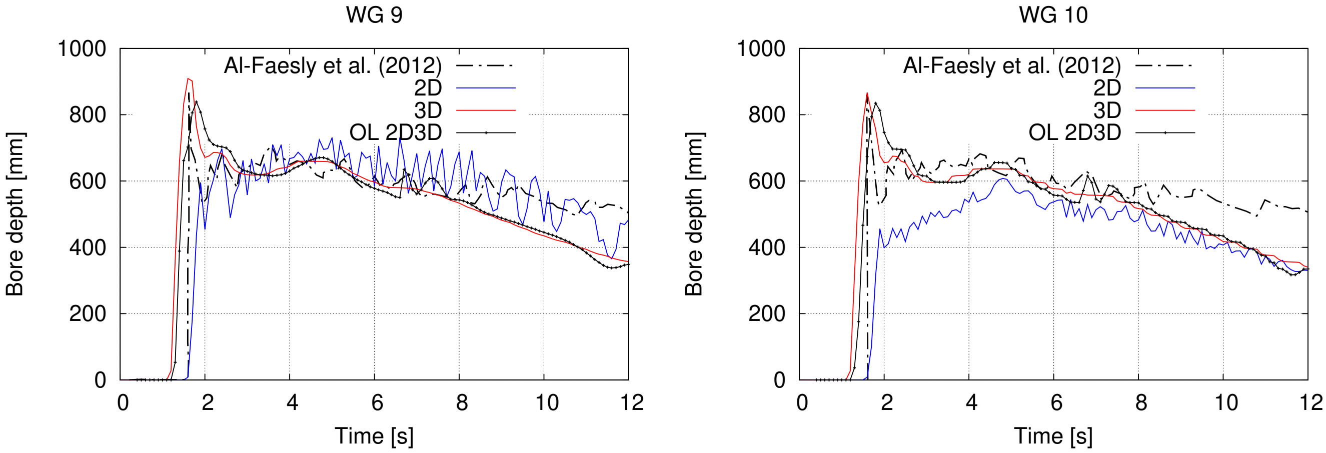


Figure 6.14: Bore depth time series at WG 9 and 10 with $h_{\text{im}} = 850$ [mm]

Right after the impact of the bore front on the obstacle, the vertical velocity becomes more pronounced and the water level springs up around the model front. When the potential energy of the flow reaches the peak, the wave peak breaks that partial flows travel upstream and the other part passes around the cylinder. When estimating the bore depth on the side face, recorded at WG 11, a considerable overestimated flows are observed by numerical solutions exhibited in figure 6.15.

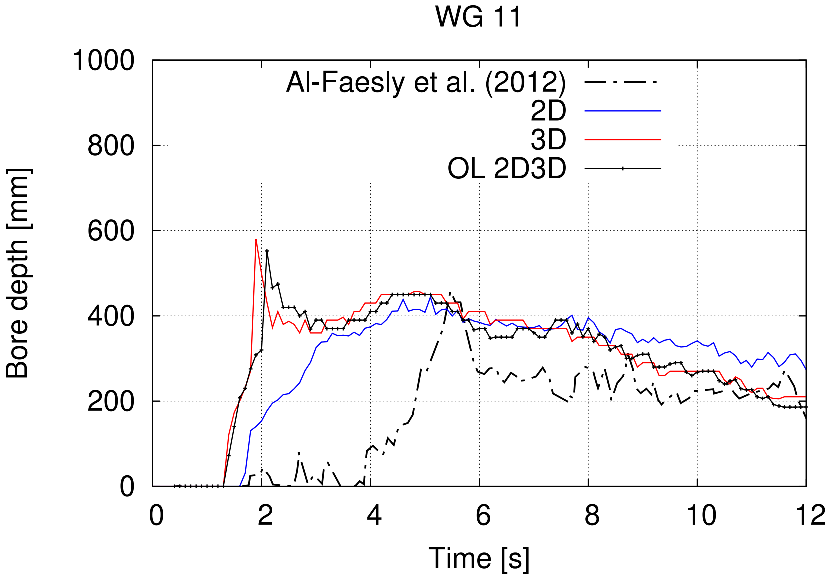
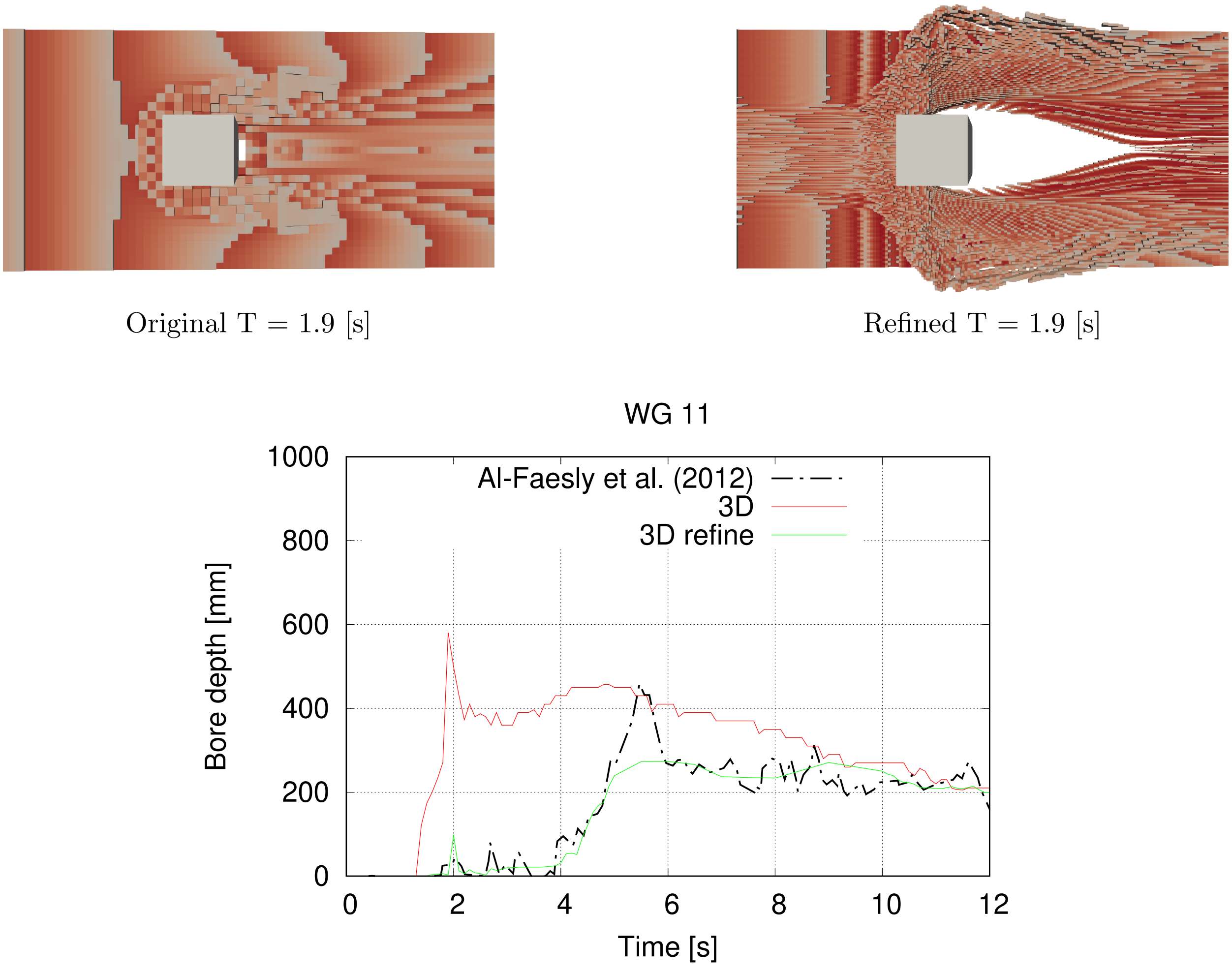


Figure 6.15: Bore depth time series at WG 11 with $h_{\text{im}} = 850$ [mm]

This can be explained by the inadequate mesh resolution in span-wise direction that the flow separation at the model's side is not represented in the simulation. To accurately produce the splash at initial impact, the aforementioned necessary minimum grid spacing is 2 [cm].

It was recommended by Douglas and Nistor (2014) that a substantial improvement of the results can be obtained when further refining the mesh around the column region. With this perspective, a mesh refined 3D model is conducted following the Douglas and Nistor (2014) mesh structure, where the grid spacing varies from 1 - 2 [cm]. Screen shots of the VOF indicator α_1 representing the flow depth in figure 6.16 at T = 1.9 [s] indicate notable difference flow behaviors in span-wise. In the refined mesh case, the span-wise velocity plays a prominent role in flow separation at the trailing edges of the cylinder. In the original case, the flow remains attached to cylinder side faces and only separate at the back of the cylinder. The refined 3D results fit the physical model well except the capture of the bore peak at 5.5 [s]. The mesh resolution is still not fine enough on this regards.

Regardless of the bad match between the measurement and the coarse meshing simulations at T = 2 - 5.5 [s], the OL 2D3D outcome stays in between the 2D and the original 3D and more close to the 3D.



Original T = 1.9 [s]

Refined T = 1.9 [s]

T = 1.6 [s]

Figure 6.16: Local view of free surface screen shot with $h_{\mathrm{im}} = 850$ [mm]

The influence from the aforementioned insufficient fine mesh emerge also to wave gauges at back side of the column. WG 12 locates at the edge of the back face where is close to the side face, an obvious run-up ahead of the experimentally controlled one is observed by the coarse grid simulations and a better fit is obtained from the refined 3D model. Comparing to the WG 12, the WG 13 lies in the center of the back face which stays further to the side face than the WG 12, the overvalued volume of the early arrived bore from the coarse grid models reduces. From 4 [s] onward, the OL 2D3D and the 3D fit the measurement well. Strong oscillations are shown by the 2D.

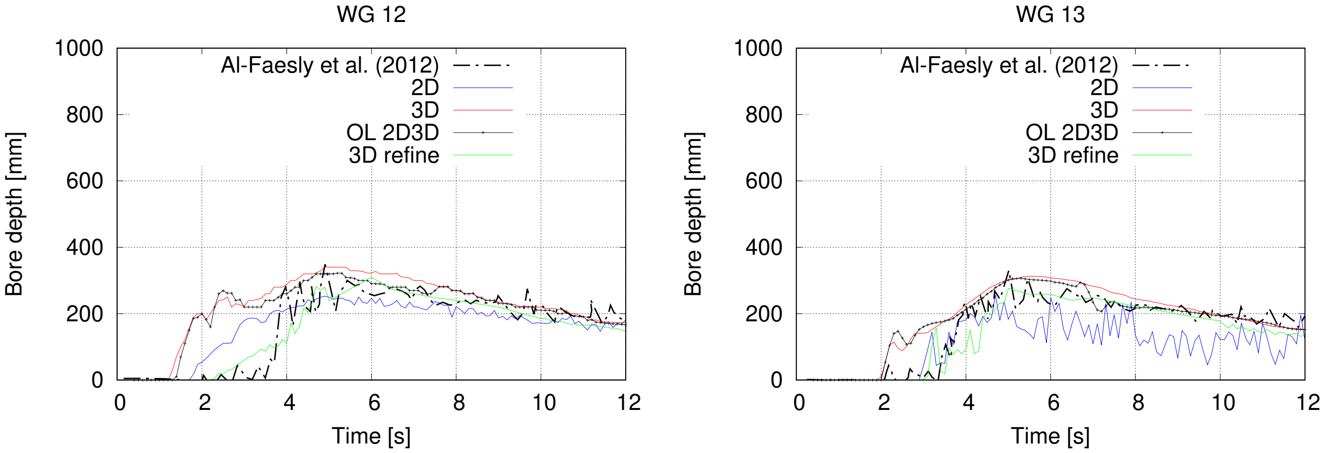


Figure 6.17: Bore depth time series at WG 12 and 13 with $h_{\mathrm{im}} = 850$ [mm]

6.4.3. Force

Tsunami-induced forces on structure contribute crucial effects of tsunamis. The numerical and experimental base shear force time histories induced on the structure is discussed here. To estimate tsunami-induced forces in tsunami-resilient structures design, formulations published in FEMA P-646, the Guidelines for Design of Structures for Vertical Evacuation from Tsunamis by Federal Emergency Management Agency (2019), are referred in this study.

Forces exerted on a structure resulting from the interaction of a hydraulic bore are associated with three stages in bore depth during the bore-structure interactions respectively: (1) the first splash-up of the incoming bore onto the front face of the structure, (2) the initial run-up flow begins to rebound onto the incoming flow, and (3) the break bore flows around the structure. A typical time-history of such forces is thus characterised by three force components: (1) the impulsive force, (2) the run-up force and (3) the hydrodynamic force.

- The impulsive or impact force occurs when the leading edge of the bore reaches the solid structure as the initial impingement, followed by the splash-up in the water level. Such force behaves different in dry- and wet-bed conditions. In laboratory scale bore-structure interactions, the impact of impulsive force in dry-bed conditions is less significant than in wet-bed conditions. The maximum ”overshoot” in force observed in wet-bed conditions is as much as 1.5 times the hydrodynamic force (Federal Emergency Management Agency (2019)). As a result, such force may lead to sever damage to structures.

- Another increase in force formulates the run-up or transient hydrodynamic force. Due to the bore flow accumulation in front of the structure, the force magnitude keeps growing with a relatively smooth gradient. The run-up force duration is longer than the impulsive force's.
- The hydrodynamic force is acted at the centroid of the wetted surface, which means the force is applied to the entire structure if the bore flows around the structural body. This force functions as the drag force combining the lateral force and the friction force.

Calculation of hydrodynamic forces on near-shore structures using equation (6.4.1) proposed by Yeh (2006):

$$F_\mathrm{d} = \frac{1}{2}\rho_\mathrm{s} C_\mathrm{D} B (hu^2)_\mathrm{max} \tag{6.4.1}$$

where F_d [N] is the hydrodynamic force, ρ_s [$\mathrm{kg}\cdot\mathrm{m}^{-3}$] is the fluid density, C_D is the drag coefficient, B [m] is the width of the structure in the plane perpendicular to the flow direction, h [m] is the flow depth, and u [$\mathrm{m}\cdot\mathrm{s}^{-1}$] is the flow velocity at the location of the structure.

The maximum momentum flux term $(hu^2)_\mathrm{max}$ is not necessarily a function of the maximum flow velocity and the maximum bore depth ($(hu^2)_\mathrm{max} \neq h_\mathrm{max}u^2_\mathrm{max}$). Estimation of this term can be performed using a simplified formula given by FEMA P-646 based on Yeh (2007):

$$(hu^2)_\mathrm{max} = gR^2(0.125 - 0.235\frac{z}{R} + 0.11(\frac{z}{R})^2) \tag{6.4.2}$$

where g [$\mathrm{m}^2\cdot\mathrm{s}^{-1}$] is the gravity acceleration, R [m] is the maximum inundation depth at the site and z [m] is the bottom elevation of the structure.

To well anticipate R, uncertainty inherent of experiments in terms of the tsunami run-up level, the splash-up of tsunami bore impact and possible flow retention in the structure should be taken into account. Preliminary predicted maximum inundation depth R^* at the location of the model without the structure (*i.e.* at WG 0) is determined by physical / numerical modelling. According to FEMA P-646, R can be therefore evaluated as 30% higher than R^*, see equation (6.4.3).

$$R = 1.3R^* \tag{6.4.3}$$

In OpenFOAM, a 3D model introduces rigid objects in specifying the object surfaces as a wall boundary, while in a 2D model the objects are represented by prescribing their surface elevations, so that no extra boundary is generated.

In our bore-structure interaction numerical simulations, the structure is introduced by specifying a rigid object in the `OL 2D3D` and the `3D` models. The object surfaces are set as a wall boundary that forces imposed on the front face of the structure can be directly computed as a solution output for this boundary patch. Nevertheless, in our 2D model, no additional boundary is created and the structure is represented as a part of the channel bottom where the bottom elevation Z is prescribed to 1. In this way, forces on the structure for 2D model can not be calculated by the model itself. We use equations (6.4.1) - (6.4.3) to estimate the 2D forces. Parameters for numerical forces determination are fixed that the density of tsunami flow, a mixture of sediment and seawater, ρ_{s} sets as 1200 $[\mathrm{kg} \cdot \mathrm{m}^{-3}]$ and the drag coefficient C_{D} is 2.0 for rectangular structure (Federal Emergency Management Agency (2019)). The experimental forces were recorded by the dynamometer.

Comparison of numerical and experimental forces acted on the structure in the direction of the flow is displayed in figure 6.18. 2D model in general reproduces much oscillating and underestimated hydrodynamic forces with a rather flat tendency. These could be attributed to the incapability of a 2D model in bore-structure interaction modelling and improper force parameters estimation. 2D bahavoir in terms of the force will not be discussed here. Overall, simulated forces from the OL 2D3D and 3D models and the measured forces follow the trend of the aforementioned typical force time-series features. The 3D curve exhibits the smoothest force profile, whereas several bulges in the OL 2D3D curve are resulted from the impact of 2D model.

The first impact, the impulsive force, coincides with the sharp increase in the bore depth (see figure 6.14) as the bore front touches the structure front face. Compared to the measured data, gradients of the impulsive force in numerical models are rather gradual. The magnitude of this force (*e.g.* the maximum impulsive force is 306 [N] at T = 1.7 [s]) is accurately reproduced by the numerical model. During T = 2 - 5 [s], the run-up force dominates as the bore depth increases at the structure front face. The overall maximum force is produced at T = 4.7 [s], where the measured force reaches 630 [N] and the computed forces are 535 [N]. The OpenFOAM numerical models predict the force with an underestimation, which was also observed by Sarjamee et al. (2017). Subsequently, the force decreases due to a pressure drop on the front face of the structure since the entrapped air is expanded and compressed (Bullock et al. (2007)). The peak hydrodynamic force does not exceed the peak run-up force resulting from lots of energy being reflected upstream.

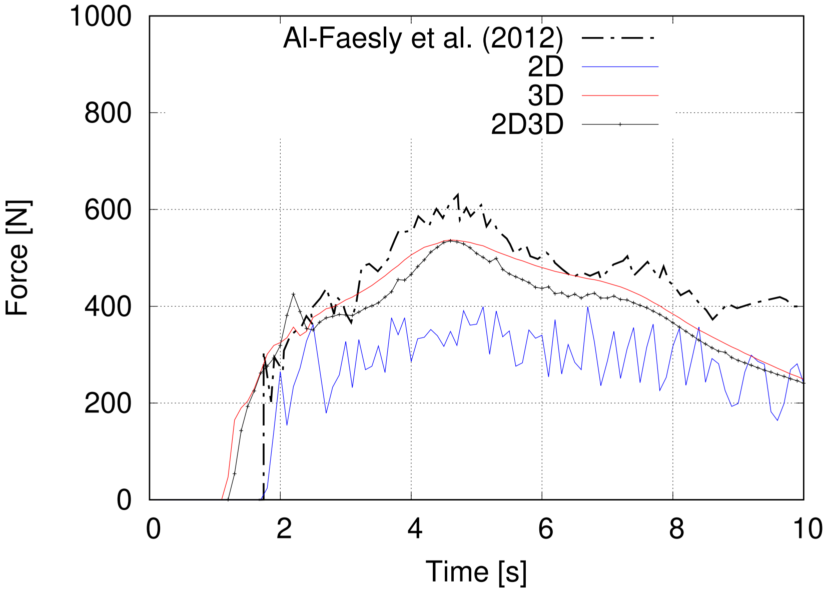


Figure 6.18: Force time series at the front surface of the structure with impounding depth of 850 [mm]

The gradual gradient in impulsive force and the underestimation in run-up force are attributed to a not fine enough mesh resolution and the approximation of the eddy viscosity and wall shear stress. The maximum run-up force accounts for the maximum force imposed on the structural element. Additionally, Al-Faesly et al. (2012) reported that parameters proposed by FEMA P-646 lead to hydrodynamic forces underestimation.

6.5. Comparison together

Same set-up with different impounding water h_{im} of 550 and 1150 [mm] are further tested in both with and without structure presence. Comparison of the `OL 2D3D` bore depth and force time-history profiles by the three initial impounding water (*i.e.* 550, 850 and 1150 [mm]) will be discuss in this section to find a common regulation.

Bore depth time series of `3D` and `OL 2D3D` numerical results are compared at two wave gauges, WG 0 and WG 9, shown in figure 6.19. On the left displays the bore depth time-history profile at the structure location with absence of the structure. Larger value of the imposed h_{im} induce higher estimation in bore depth and earlier inundation in time at WG 0. The bore depth runs up to the maximum with distinct slopes, where the highest impounding water leads to the steepest gradient and vice-versa. Consequently, the bore rises faster to the top point under a larger amount of impounding water. The bore peak of $h_{\text{im}} = 550$ [mm] case is 200 [mm] and remains constant, the bore depth of $h_{\text{im}} = 850$ [mm] case increases until nearly 300 [mm] and decrease slowly thereafter, the bore depth of $h_{\text{im}} = 1150$ [mm] case reaches its maximum of 400 [mm] and drops faster than the bore resulted from 850 [mm] impoundment. It can be concluded that the bore peak is approximately 35% of the impounding water depth.

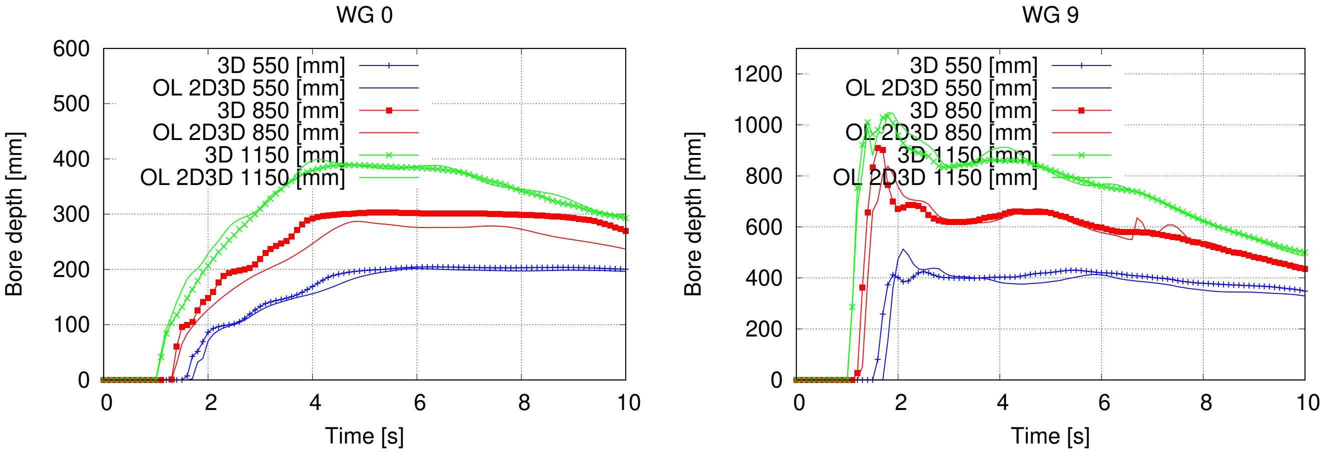


Figure 6.19: Bore depth time series at WG 0 and 9 with three h_{im}

The bore depth in front of the structure are demonstrated on the right in figure 6.19. The bore-structure induced bore peak is more or less identical to its initial impounding water depth. After the first peak, another smaller peak is produced.

Bore-induced forces by three impoundments are compared in figure 6.20. Similar tendency can be observed as in the bore depth in terms of the magnitude, the interact time and the number of peaks. Impulsive loads in $h_{\mathrm{im}} = 550$ to 1150 [mm] cases ranges from 85 - 55% of the later run-up forces. In the smaller impounding water situation, the impulsive force accounts for a larger proportion.

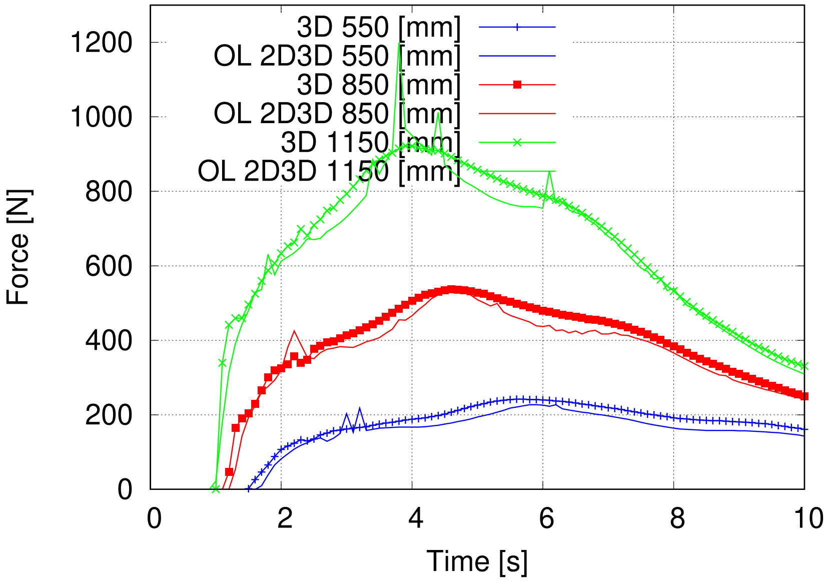


Figure 6.20: Force time series at front surface with three h_{im}

From figures 6.19 to 6.20, the `3D` and `OL 2D3D` results fit quite well in overall. Disturbances in the OL 2D3D model are observed when the structure is introduced and their magnitude increases as the impounding water depth increases.

6.6. Conclusion

This chapter performs a series of numerical investigation of the tsunami-like bores and the bore-structure interactions over dry flume bed. A dam collapse process is used to generate tsunami-like hydraulic bores and the structure with a square cross-section is introduced downstream of the flume to model the impact of bore-structure interaction. Numerical models of purely 2D, 3D and OL 2D3D are established to demonstrate capabilities of the applied models by comparing time-histories of bore depths and hydrodynamic forces exerted on the structure with the physical experiments conducted at the NRC-CHC in Ottawa, Canada. To save computational expense, a relatively coarse mesh resolution, for instance 0.04 × 0.04 × 0.02 [m^3] in x-, y- z-directions of 3D model, is applied in numerical models. Three different impounding water are prescribed to simulate dam break process, each case is tested with and without the downstream structure yielding 6 sub-cases in total.

Tsunami-like bores propagating over a dry channel bed without the downstream structure are first reproduced with three stored water depths. The initial impounding water depth contributes to bore depth developing the most. The more impounding water, the faster velocity in the bore front and the higher bore depth at the structure location. When the solid square structure is introduced, relationship between the impounding water and the bore depth remains the same with different bore behaviors. The bore is rebounded by the front face of the structure, they splash-up to their top points and collapse to flow against and around the structure.

Tsunami bore propagation and tsunami-induced forces cannot be accurately simulated by the 2D Shallow Water model in the dam failure and the structure regions, especially at the structure edge (*e.g.* WG 10 and 12). The bore depth and force time history profiles of 2D model are severely fluctuating in bore-structure simulations. Prediction by the 3D RANS model and the overlapping 2D-3D coupled model show reasonable accuracy and a general good agreement with the measurements. The 3D model presents smooth bore depth and force time history profiles and disturbances appear in the OL 2D3D model.

7. Development of a workflow for 2D-3D coupled flood simulations

With aim of building an integral multi-scale flood simulations framework, a fully developed workflow is described in this chapter. To make the most advantages of the present coupled solver in a complex flood scenario, for intense an urban flood, purely 2D simulations are recommended to compute prior to coupled simulations. With the help of the 2D results, users may get a general view of the whole problem in a short time and define problematic regions as 3D domains for the coupled simulation later on.

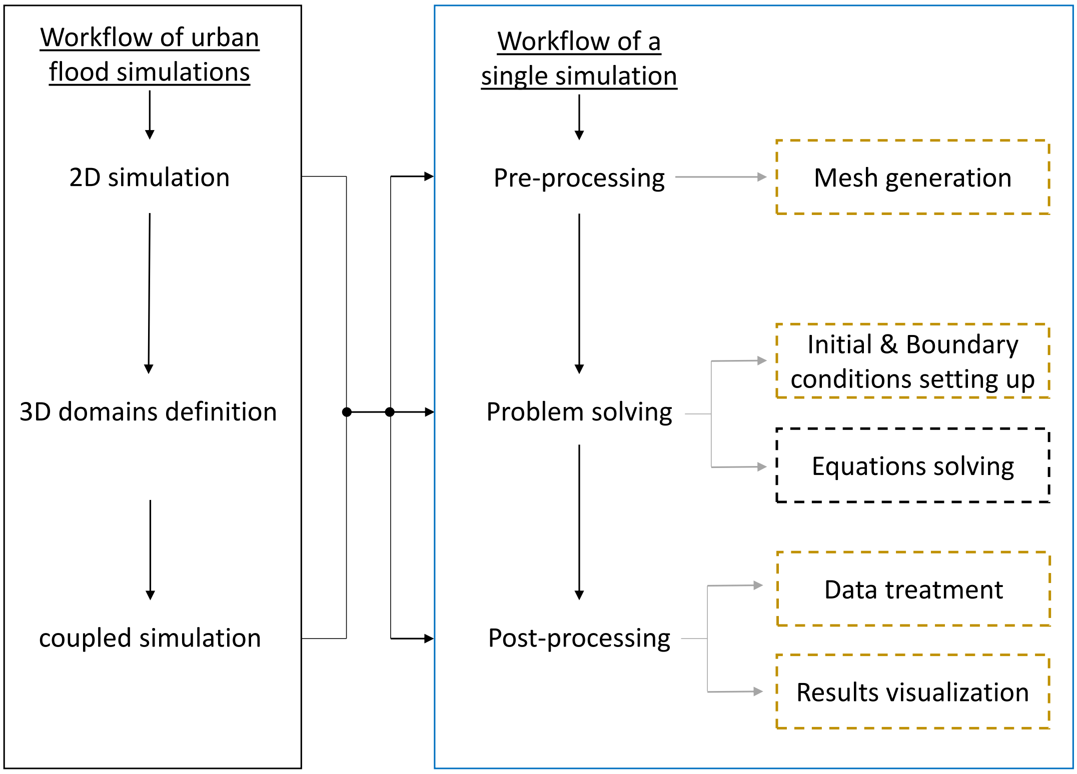


Figure 7.1: 2D-3D coupled flood simulations framework

Figure 7.1 outlines the framework of urban flood simulations where the flood simulation workflow is defined on the left side and every single CFD simulation workflow is demonstrated on the right side including necessary tasks of pre-processing, problem solving and post-processing. Chapter 4 mainly discussed the problem solving part by implementing a new coupling algorithm. Apart from equations solving, mesh generation, initial and boundary conditions setting up, data treatment and results visualization are remaining to establish. To complete the workflow, terms bounded by an orange dashed line in figure 7.1 specifically for coupled simulations will be detailed in this chapter.

7.1. Mesh generation

Mesh generation is the groundwork which plays an significant role in CFD simulations. How to build suitable mesh for practical utilization is indispensable. A complete workflow in mesh generation is introduced here along with special treatment in coupling meshes creation.

Currently, only cases with regular bottom have been tested; where coupling meshes are created by using `blockMesh`, a default mesh generation utility supplied with OpenFOAM. On the interest of realistic geometry creation, `snappyHexMesh` can be used on based of the `blockMesh` generated background mesh to create complex body-fitted 3D meshes (more details refer to appendix C). In the process of mesh generation, 3D meshes can directly obtained from `snappyHexMesh` since `snappyHexMesh` is originally designed for 3D mesh creation. Due to the special structure in 2D mesh, another mesh manipulation function, `extrudeMesh`, is applicable which extrudes the given mesh from existing patch with the specified layers. 2D meshes can therefore be achieved by executing `extrudeMesh` on the body-fitted 3D meshes from the top patch to the bottom patch with only one layer in between.

As a consequence of the coupling algorithm used in this work, the 2D and the 3D mesh must be matched at the overlapping area in horizontal directions. Fully body-fitted girds in 3D region can hardly match to the 2D region under influence of snapping mesh to the surface step in `snappyHexMesh`. For this reason, the castellated surface representation mesh for 3D regions has been chosen.

Steps in mesh generation flowchart exclusively for the coupled solver are exhibited in figure 7.2, where a circle represents the input file(s), a rectangle is the resulting mesh produced from the previous execution. Executable utilities are marked in the grey color of which texts with a underline is an OpenFOAM command. In general, `blockMesh` provides a background mesh which covers the whole geometry; by means of `snappyHexMesh`, a 3D body-fitted mesh is thereby generated as the base mesh for 2D and 3D regions creation; mesh for purely 2D simulations can be obtained by extruding the base mesh from 3D to 2D and sub-meshes for 2D region in the coupled simulation are reached by removing all 3D regions from the resulting 2D mesh; similarly, 3D sub-meshes are extracted from the base mesh in the predefined 3D regions. Simply by means of combing the 2D mesh and the castellated 3D mesh does not results the proper mesh which is compatible with the coupled solver. In order to get a proper coupling mesh for the coupled solver, two additional special care are required which are highlighted in blue dashed line boxes and are elaborated in the following subsections.

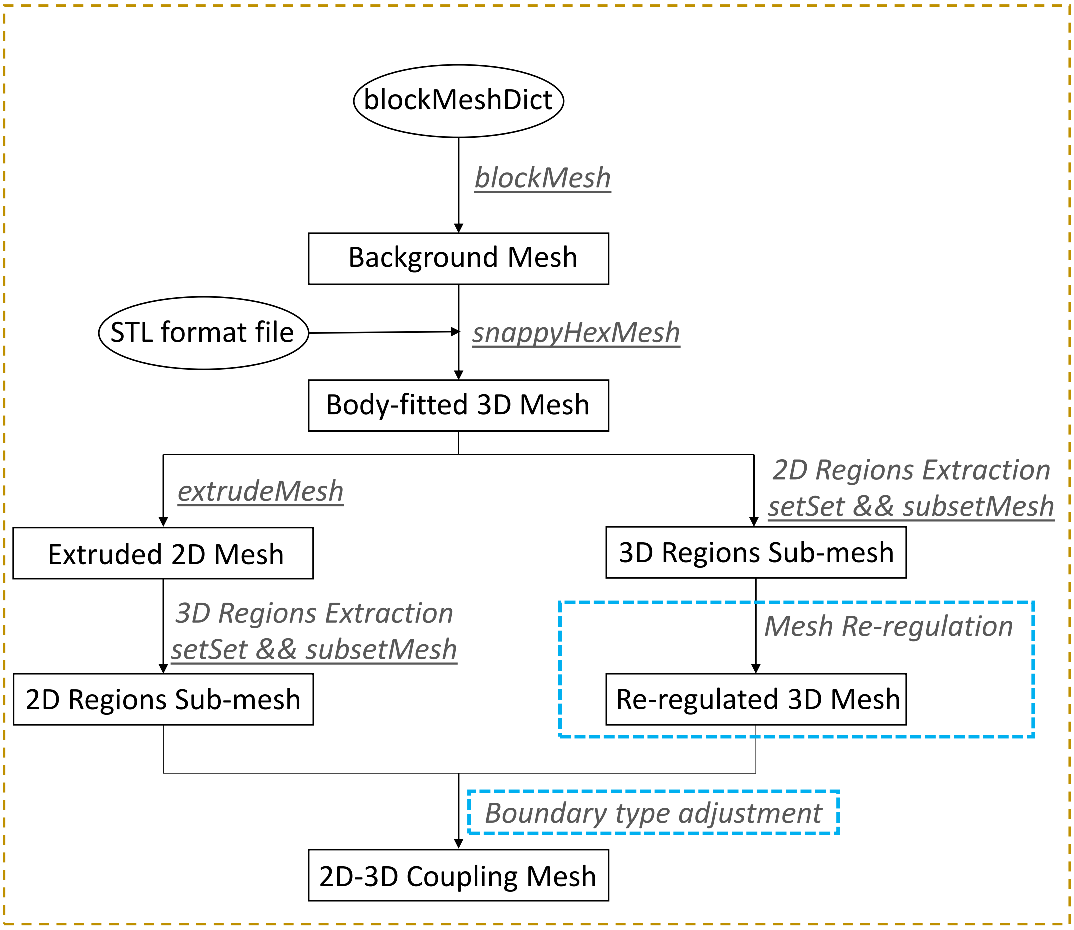


Figure 7.2: Mesh generation flowchart

7.1.1. Mesh re-regulation and adaptation for 3D mesh

When applying `snappyHexMesh` to generate body-fitted 3D mesh (only using the castellated mesh), unwanted cells are removed and new boundaries are formed. The process of 3D regions sub-mesh creation is listed as follows:

- (1) Generate the background mesh.
- (2) Introduce the geometry input data (*e.g.* the STL file).
- (3) Define unwanted cells following the surface feature.
- (4) Remove these cells.
- (5) Define 3D sub-region.
- (6) Extract 3D sub-mesh.

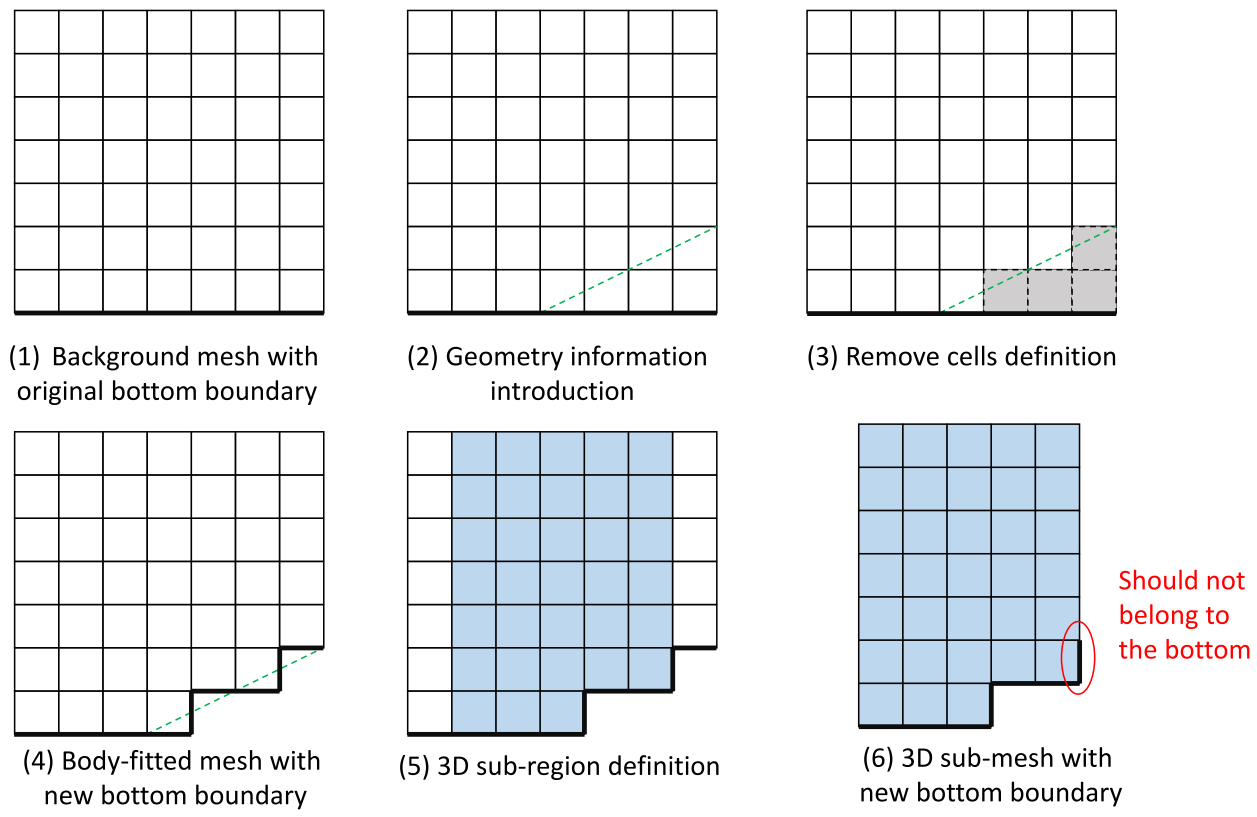

(1) Background mesh with original bottom boundary
(2) Geometry information introduction
(3) Remove cells definition
(4) Body-fitted mesh with new bottom boundary
(5) 3D sub-region definition
(6) 3D sub-mesh with new bottom boundary
Should not belong to the bottom

Figure 7.3: Process of 3D sub-mesh creation with ill-posed bottom boundary

However, this could cause ill-defined bottom boundary problems. For example we create a 3D sub-mesh with the given STL file (marked in green dotted line) shown in figure 7.3, steps (1) - (4) are general processes by the `snappyHexMesh` and the bottom boundary is marked as bold black line. After the body-fitted treatment, the new bottom boundary is no longer flat. If we define our 3D sub-region marked as blue, a face from the previous bottom boundary highlighted by the red circle becomes a lateral boundary face, which should be identified as a member of the coupled interface. This face is now considered as a bottom element. Consequently, a false wall is built in the simulation and the coupled model fails to run. In view of this, all 3D sub-meshes should be checked and re-regulated the boundary patches if necessary to match the correct interface needs.

A mesh re-regulation method is introduced to establish proper meshes for the coupled solver. The mesh re-regulation treatment is dedicated to finding out all ill-posed faces ID at the first place and regrouping them into the correct boundary patches. Accordingly, these ill-posed faces are taken out from the bottom boundary patch.

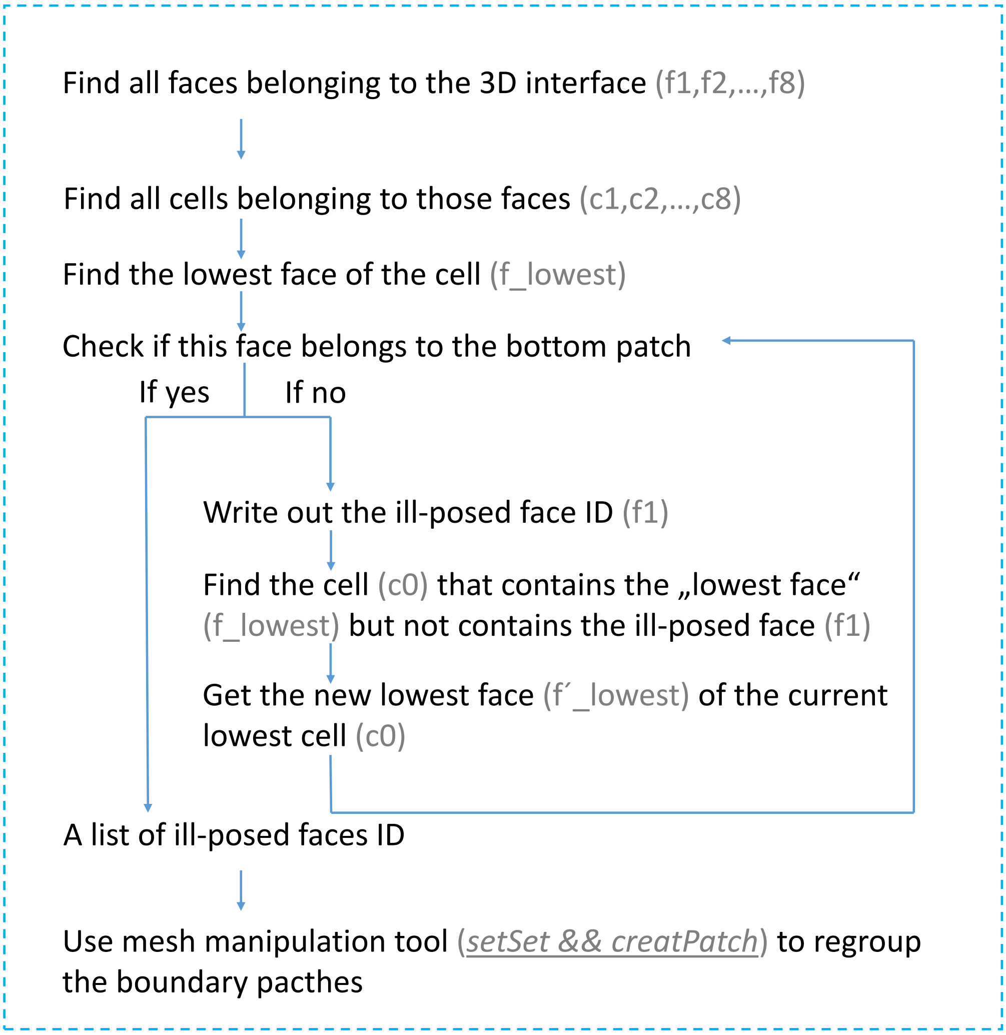

Find all faces belonging to the 3D interface (f1,f2,…,f8)
Find all cells belonging to those faces (c1,c2,…,c8)
Find the lowest face of the cell (f_lowest)
Check if this face belongs to the bottom patch
If yes
If no
Write out the ill-posed face ID (f1)
Find the cell (c0) that contains the „lowest face“ (f_lowest) but not contains the ill-posed face (f1)
Get the new lowest face (f´_lowest) of the current lowest cell (c0)
A list of ill-posed faces ID
Use mesh manipulation tool (setSet && creatPatch) to regroup the boundary pacthes

Figure 7.4: Mesh re-regulation flowchart

For better understanding the method, the flowchart in figure 7.4 is explained with labels from an elaborated example illustrated in figure 7.5. Without the mesh re-regulation, the 2D-3D interface boundary is marked as the red line and the ill-posed face is marked as the grey line in figure 7.5. The mesh re-regulation :

1. finds all faces on this interface boundary, denoted as `f1` to `f8`;
2. finds the corresponding cells which belonging to these faces, denoted as `c1` to `c8`;
3. find the lowest face of all these cells, demoted as `f_lowest`;
4. check whether the lowest face `f_lowest` belongs to the bottom patch;
5. if it is not a bottom face, find the other cell which contains the `f_lowest`, denoted as `c0`;
6. find the lowest face of `c0`, denoted as `f'_lowest`, and the lateral face on the `c0` aligned with `f1` to `f8`, denoted as `f0`, and repeat step 4;
7. if it is the lowest face, quit the if-loop and regroup the ill-posed faces ID (*e.g.* `f0`) by

using two OpenFOAM mesh manipulation tools: `setSet` and `createPatch`

where, all ill-posed faces ID are stored in `setSet` and `createPatch` modifies the patch components by deleting these ill-posed faces from the bottom boundary and add them to the Γ_{3D} boundary.

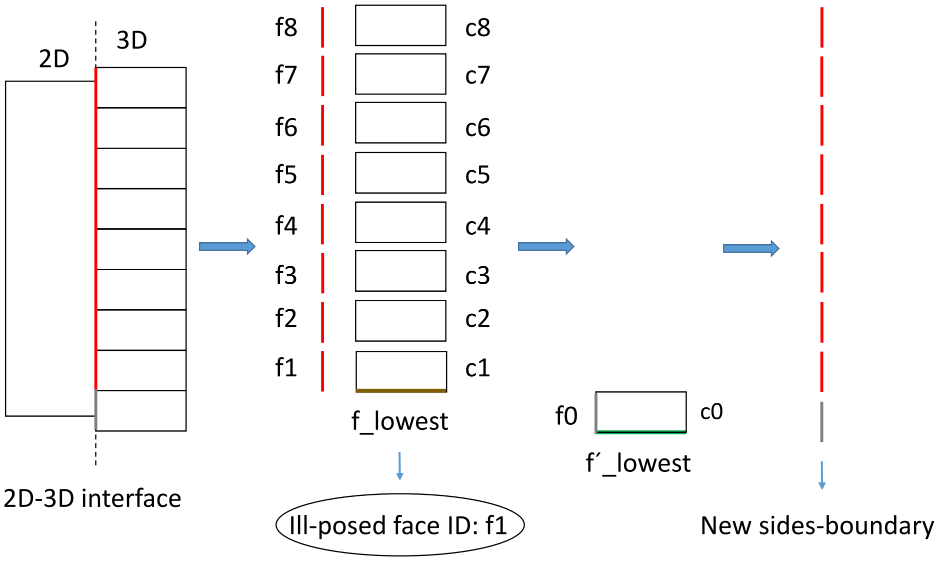


Figure 7.5: Mesh re-regulation process example

7.1.2. Boundary type adjustment

An entire mesh is represented by the defined computing domain embedded with boundary patches. Only standard boundary types that already defined in OpenFOAM can be formulated by performing OpenFOAM mesh utilities. As a developed solver, the coupled solver requires a customised boundary type allowing coupling regions to match and to communicate. As a consequence, boundary types setting up for coupling areas shall be adjusted.

A python script is implemented to update boundary types for coupling regions.

7.2. Initial and boundary conditions setting up

Initial and boundary conditions in most cases are mainly prescribed by `setFields`-utility that values for the target scalar or vector field are set without programming. Relevant `setFields`-utilities are:

- `setFields`: set values on a selected set of cells or boundary faces via a simple predefined dictionary.

- `funkySetFields`: set values for cell centers on selected internal field depending on an user defined expression.
- `funkySetBoundary`: set values for faces centers on selected boundary depending on an user defined expression.

Nevertheless, an irregular list of values, *e.g.* real terrain data, is difficult to obtain by either a easy definition or writing expressions. Bottom data extraction of a realistic terrain for 2D meshes and set them to the 2D solver are studied.

7.2.1. Surface interpolation of terrain data for 2D simulations

As introduced in appendix C, the 2D mesh in OpenFOAM is a pseudo 2D mesh. The mesh only follows the geometry in the horizontal dimensions (x- and y- directions) and is always flat with a constant thickness in the vertical dimension (z-direction). Most of the numerical simulations read the geometry information through the mesh, such as the dimension of the simulation domain, the objects inside the domain and the bottom elevation. Unfortunately, no information in z-direction is available in the OpenFOAM 2D mesh. For this reason, another parameter `Z` is introduced in `shallowFoam` to represent Z_{bi} in equation (2.2.23).

We can easily assign regular bottom level values to 2D cell centers by using mesh manipulation tools in OpenFOAM (*e.g.* `setFields`, `funkySetField` (Greenshilds and Ltd, 2015)). For cases that include river flow over a realistic terrain a problem arises since the elevation is usually not uniform and varies a lot throughout the domain. Usually, the data that describes a 3D realistic terrain is initially an unstructured set of X Y Z points which are obtained using 3D aerial laser scanning methods, *e.g.* the unstructured set of points is firstly used to construct a bottom geometry in a STL format using triangulation methods. Besides using them for geometry creation, they are useful for creating interpolation surfaces, which could be used to describe the completely realistic terrain geometry.

As exhibited in figure 7.2, a 2D mesh was created from available triangulated data (STL) using the OpenFOAM meshing utility `snappyHexMesh`. Since bottom elevation data is required to solve the SWE it was necessary to store it in the cell centres. This procedure was successfully completed using surface interpolation with radial basis functions.

Surface interpolation method

An interpolated surface could be made based on a set of unstructured points that describe a real terrain. For this case we use radial basis functions to approximate the surface of a real terrain. Creating an interpolated surface allows us to extract the elevation level for every X and Y coordinate of the mesh cell centre. Radial basis functions were used to obtain elevation in realistic surfaces. Radial basis interpolation in equation 7.2.1 describes a simple variant of the radial basis interpolation method.

$$z_{\mathrm{e}} = \sum_{k=1}^{n} \lambda_{\mathrm{k}} \phi(r_{\mathrm{k}}) + b \tag{7.2.1}$$

where z_{e} [m] is the estimated bottom elevation for the surface grid point, r_{k} [m] is the distance from the unstructured grid point to the input data, λ_{k} is the mapping coefficient and b [m] is the bias value. λ_{k} and b are estimated from the unstructured data points.

The function of distance or the radial basis function could be defined as a wide variety of different spline functions and in this case, the radial basis function $\phi(r)$ is defined as a thin plate spline (a special polyharmonic spline) function 7.2.2. The thin plate spline function minimizes the bending energy which balances the tradeoff between the accuracy to the data and the smoothness to the interpolated values.

$$\phi(r) = r^2 log(r) \tag{7.2.2}$$

To apply the method of surface interpolation with radial basis functions, a set of unordered X Y Z points that describe a geometry need to be used to create a surface interpolation function. In order to use the created surface interpolation function that is based on the original geometry, the extracted cell centres X_{c} and Y_{c} of the 2D mesh need to be evaluated. The output of this process is the corresponding elevation Z_{c} for every X_{c} and Y_{c}.

Cell center points extraction method is tested by the 2D model on a flooding simulation case that has a realistic terrain. The case simulated the Malpasset river dam break which was experimentally and numerically studied in (Alcrudo and Gil, 1999), (Valiani et al., 2002) and (Hervouet and Petitjean, 1999). Our numerical analysis showed good agreement with provided experimental data. More details of the test case can be found in Zeng et al. (2018).

7.3. Conclusions

A complete workflow for the coupled model includes (1) mesh generation, (2) initial and boundary conditions setting up, (3) equation solving, (4) data treatment and (5) results visualization. This chapter mainly tackle the challenges in realistic geometry simulations for (1) mesh generation and (2) initial and boundary conditions setting up steps. Step (3) equation solving has been elaborated in chapter 4. Steps (4) data treatment and (5) results visualization are strongly dependent on user's need. The most common and easiest way is to use post-processing functions or tools provided by OpenFOAM.

8. Conclusion and Outlook

The main interests in improving an 2D-3D coupling algorithm for free surface flows have been investigated in this research. This coupled model could be used as an efficient numerical tool in hydrodynamic events assessment, which provides insight in fluid flow phenomena and guidelines for civil and structural protection.

The coupling algorithm improvement in this work was based on a characteristics-based coupling algorithm proposed by (Mintgen, 2017), which combined the 2D Shallow Water solver `shallowFoam` and the 3D RANS solver `interFoam` in the OpenFOAM framework. A transfer of information between 2D and 3D solvers has been implemented via the exchange of boundary conditions at the interface between 2D and 3D sub-meshes. Within the coupling algorithm implementation, a non-overlapped mesh where the 2D and the 3D sub-meshes perfectly matched in horizontal directions is necessary. The direction of information transfer from one sub-region to another depends on the flow characteristics: the flow condition and the flow direction. In this characteristics-based coupling algorithm, the interface receives information by means of the Neumann condition and transfer the information by Dirichlet condition. However, a shortcoming of this coupled solver was that the coupling algorithm cannot detect an upstream-travelling hydraulic jump in the supercritical region, which is caused by a zero-gradient condition of flow depth at the interface of this region. This blocking of hydraulic jump result in non-physical simulation results in some particular cases *e.g.* upstream-travelling hydraulic jump over the incoming supercritical flow.

To overcome the hydraulic jump blocking problem, several ad-hoc modifications of the characteristics-based coupling algorithm have been tested in order to transfer the downstream hydraulic jump information correctly over the interface: (1) apply the Dirichlet condition for all flow features on the interface from both sides regardless the flow conditions; (2) add a discontinuous wave corrector (hydraulic jumps can induce discontinuous waves and the wave celerity is no longer equals $\sqrt{gh}$) in Fr calculation to redetermine the flow condition; (3) set up a hydraulic jump detector to sense whether an upstream-travelling hydraulic jump is approaching to the interface, if an approaching hydraulic jump is detected, then we apply the Dirichlet condition on the upstream interface and the Neumann condition on the downstream interface. Upstream-travelling surge test cases were simulated by the above coupling algorithms. With the ad-hoc modifications (1) and (2), the resulting surge was still blocked at the interface. The third ad-hoc modification of introducing a hydraulic jump detector was able to transfer the upstream-travelling surge through the interface, however, the surge behaved unrealistic after passing through the interface.

Therefore, the improved 2D-3D coupling algorithm has been developed to solve the drawback mentioned above. A simplified overlapping method has been implemented using just one Schwarz iteration and cell-to cell overlapping. Instead of using common faces, data mapping between 2D and 3D sub-regions is applied by means of the overlapped cells. Within every time step of simulation, the following procedures are performed sequentially: (1) the Ω_{2D} is solved, (2) from the cells located in the overlapping regions and adjoin to the Γ_{3D}, the 2D results are mapped to the Γ_{3D}, (3) the Γ_{3D} passes the received values into the adjacent 3D cells, (4) the Ω_{3D} is solved with the updated information from the Ω_{2D}, (5) the 3D results then update the Ω_{2D} in an analogous manner. Grid sizes of 2D and 3D sum-meshes in overlapping regions shall be also matched in horizontal directions. Data transfer between connected regions is achieved via a set of cells with identical horizontal dimension, thus no interpolation is required.

The overlapping method has proved to work in the same upstream-travelling surge test where the upward transfer surge passes the coupling region with an overall good accuracy compared to the pure 3D model, but disturbances appear in the overlapping region in the process of surge transferring the overlapping region. Sensitive studies of this case indicates that when we shrink the overlapping region, the disturbances become smaller accordingly. Solitary waves transport in supercritical and subcritical flows have further demonstrated the ability of the new coupling algorithm. Wave transfer in these cases are smooth and no flow condition change occurs. As a consequence, smoother transfer processes over the overlapping region are reproduced in contrast to the surge transfer case. The bore-structure interaction has been comprehensively studied and compared with the experiments. Good agreement with the measurement data and the 3D simulations around the structure located area have been obtained by the new method.

Finally, a workflow for running the new coupled model has been developed which is applicable in realistic applications. The workflow consists of mesh generation, simulation set-up, call the solver and the visualization.

Open points for future work are subdivided into the four major points and are proposed as follows:

Improvement of 2D solver Disturbances can be observed when the wave shapes and characteristics differ extensively between 2D and 3D models. A better fit could be reached by improving the 2D model that brings the 2D wave closer to the 3D.

High performance computations Unfortunately, the new coupling algorithm is not able to run in parallel yet. An efficient prediction of flood events requires the use of High-Performance Computing (HPC). Enabling the ability of the new solver for parallel running is the next task beyond doubt. Moreover, the speed-up will not be linearly increased after the number of processor reaches the critical point. The major influencing factor on speed-up is the communication and synchronization overhead between processors. More precisely, improvement of effectiveness of communication needs to be tackled.

User interface development The current workflow was written in the bash and python scripts. To make the whole process more user friendly and conveniently, a nice user interface shall be developed that users can import the geometry data and set relevant coefficients in mesh creation, solving and post-processing calculations. More specifically, the automatic mesh generator could be a worthy point that provides an interactive mesh refinement where complex flow features are likely to occur.

Real world applications In this research, real world applications are not tested. It would be very interesting to see the overlapping coupled solver performance for realistic simulations.

Bibliography

A. G. Adeogun, M. O. Daramola, and A. Pathirana. Coupled 1d-2d hydrodynamic inundation model for sewer overflow: Influence of modeling parameters. *Water Science*, 29(2): 146–155, 2015.

R. Adrian. Analysis and interpretation of instantaneous turbulent velocity fields. *Experiments in Fluids*, 29, 2000.

R. Ahmadian, R. A. Falconer, and J. Wicks. Benchmarking of flood inundation extent using various dynamically linked one- and two-dimensional approaches. *Journal of Flood Risk Management*, pages 1–15, 2015.

T. Al-Faesly, D. Palermo, I. Nistor, and A. Cornett. Experimental modeling of extreme hydrodynamic forces on structural models. *International Journal of Protective Structures*, 3:477–505, 12 2012. doi: 10.1260/2041-4196.3.4.477.

F. Alcrudo and E. Gil. The malpasset dam break case study. In *Proceedings of the 4th Concerted Action on Dambreak Modelling Workshop*, pages pp. 95–109, 1999.

D.A. Anderson, J.C. Tannehill, and R.H. Pletcher. *Computational fluid mechanics and heat transfer*. 1984. OSTI ID: 5012735.

J.D. Anderson. *Computational Fluid Dynamics: The Basics with Applications*. McGraw Hill, New York, 1995.

A. Apostolatos. *Isogeometric Analysis of Thin-Walled Structures on Multipatch Surfaces in Fluid-Structure Interaction*. PhD thesis, Technische Universität München, 2019.

J. Roenby B. E. Larsen, D. R. Fuhrman. Performance of interfoam on the simulation of progressive waves. *Coastal Engineering Journal*, 61(3), 2019. ISSN 380-400.

J A Benek, J L Steger, F C Dougherty, and P G Buning. Chimera. a grid-embedding technique. Technical report, DTIC Document, April 1986.

Edin Berberović, Nils Van Hinsberg, Suad Jakirlić, Ilia Roisman, and Cameron Tropea. Drop impact onto a liquid layer of finite thickness: Dynamics of the cavity evolution. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 79, 04 2009. doi: 10.1103/PhysRevE.79.036306.

E. Blade, M. Gomez-Valentn, J. Dolz, J.L. Aragon-Hernandez, G. Corestein, and M. Sanchez-Juny. Integration of 1d and 2d
nite volume schemes for computations of water flow in natural channels. advances in water resources. *Journal of Hydraulic Engineering*, 42:17–29, 2012.

J. Boussinesq. Théorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, en communiquant au liquide contenu dans ce canal des vitesses sensiblement pareilles de la surface au fond. *Journal de Mathématiques Pures et Appliquées*, pages 55–108, 1872. URL `http://eudml.org/doc/234248`.

J. Boussinesq. Essai sur la théorie des eaux courantes, 1877. Mémoires présentès par divers savants à l'Académie des Sciences.

S. Bradford and B. Sanders. Finite-volume model for shallow-water flooding of arbitrary topography. *Hydraulic Engineering*, 128(3):289–298, 2002.

M. Brocchini, D. H. Peregrine, and Philip L-F Liu. *Advances In Coastal And Ocean Engineering - Interaction of Strong Turbulence with Free Surfaces*, volume 8. WORLD SCIENTIFIC, April 2002. doi: 10.1142/4964.

G. Bullock, C. Obhrai, D.H. Peregrine, and H. Bredmose. Violent breaking wave impacts. part 1: Results from large-scale regular wave tests on vertical and sloping walls. *Coastal Engineering*, 54:602–617, 08 2007. doi: 10.1016/j.coastaleng.2006.12.002.

H. Bungartz, B. Gatzhammer, F. Lindner, M. Mehl, K. Scheufele, A. Shukaev, and B. Uekermann. precice - a fully parallel library for multi-physics surface coupling. *Computers and Fluids*, 141:250–258, 2016.

H. Bungartz, G. Chourdakis, D. Risseeuw, A. Rusch, and B. Uekermann. Scalable coupled simulations with openfoam and other solvers, 2018.

Luis Cea, Jerónimo Puertas, and María-Elena Vázquez-Cendón. Depth averaged modelling of turbulent shallow water flow with wet-dry fronts. *Archives of Computational Methods in Engineering*, 14(3):303–341, Sep 2007. ISSN 1886-1784. doi: 10.1007/s11831-007-9009-3. URL `https://doi.org/10.1007/s11831-007-9009-3`.

H. Chanson. Analytical solution of dam break wave with flow resistance. In *Application to tsunami. 31st International Association for Hydro-Environment Engineering and Research (IAHR) Congress*, pages 3341–3353, 2005.

S.H. Chen. *Hydraulic Structures.* Springer-Verlag Berlin and Heidelberg GmbH & Co. KG, Berlin, Germany, 06 2015a.

S.H. Chen. *Hydraulic Structures.* Springer, Wuhan University, August 2015b. ISBN: 978-3-66-247330-6.

Y. Chen, Z. Wang, Z. Liu, and D. Zhu. 1d-2d coupled numerical model for shallow-water flows. *Journal of Hydraulic Engineering*, 138 (2):122–132, 2012.

COM. Flood risk management - flood prevention, protection and mitigation, 2004.

R. Courant, K. Friedrichs, and H. Lewy. Uber die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen (in German)*, 100(1):32–74, 1928.

L. D'Alpaos and A. Dena. Mathematical modeling of tidal hydrodynamics in shallow lagoons: A review of open issues and applications to the venice lagoon. *Computers & Geosciences*, 33(4):476–496, 2007.

Riegel dan Beeba. Hydraulics course material, section 29.5. in thandaveswara, (2017), January 1917.

A. B. de Saint-Venant. Mémoires sur la torision des prosmes. *Mémoire des Savants éntrangers Acad.Sci. Paris*, 14:223–560, 1855.

F. Dias and P. Milewski. On the fully-nonlinear shallow-water generalized serre equations. *Physics Letters A*, 374(8):1049–1053, 2 2010. ISSN 0375-9601. doi: 10.1016/j.physleta.2009.12.043.

G. Dorfmann, C.; Zenz, editor. *The depth-averaged Mixing Length turbulence model for Telemac-2D*, Paris, France, October 2016. XXIIIrd TELEMACMASCARET User Conference, HR Wallingford. TELEMAC-MASCARET Core Group.

S. Douglas and I. Nistor. On the effect of bed condition on the development of tsunami-induced loading on structures using openfoam. *Natural Hazards*, 76, 12 2014. doi: 10.1007/s11069-014-1552-2.

R. F. Dressler. New nonlinear shallow flow equations with curvature. *Journal of Hydraulic Research*, 16(3):205–222, 1978. doi: 10.1080/00221687809499617.

ESI Group. Opencfd ltd, 2012. URL `https://www.openfoam.com/`.

Federal Emergency Management Agency. *Guidelines for Design of Structures for Vertical Evacuation from Tsunamis*, third edition edition, 2019.

C.A. Felippa. A historical outline of matrix structural analysis: a play in three acts. *Computers and Structures*, 79:1313–1324, 2001.

J.D. Fenton and G.V. Nalder. Long wave equations for waterways curved in plan. *26th Congress of the International Association for Hydraulic Research*, 1:573–578, September 1995.

Joel H. Ferziger and Mi. Peric. *Computational Methods for Fluid Dynamics*. Springer-Verlag, 1996. ISBN 3-540-59434-5.

P.R. Garabedian. *Partial Differential Equations*. John Wiley & Sons, Inc., New York, London, Sydney, 1965.

V. K. Garg. *Applied Computational Fluid Dynamics*. Taylor & Francis Inc, May 1998.

I. Y. Gejadze and J. Monnier. On a 2d 'zoom' for the 1d shallow water model: Coupling and data assimilation. *Computer Methods in Applied Mechanics and Engineering*, 196(45-48): 4628–4643, 2007.

N Gerstner, F. Belzner, and C. Thorenz, editors. *Simulation of Flood Scenarios with Combined 2D/3D Numerical Models*, Hamburg, 2014. Federal Waterways Engineering and Research Institute, Karlsruhe, Germany, ICHE. SBN 978-3-939230-32-8.

A. Goris. *Bautabellen für Ingenieure mit Berechnungshinweisen und Beispielen*. Werner Verlag, Köln, Germany, 2006.

C.J. Greenshilds and CFD Direct Ltd. Openfoam the open source cfd toolbox user guide. Technical report, OpenFOAM Foundation Ltd, December 2015.

G. L. Asawa H. Bonakdari, G. Lipeme-Kouyi. Developing turbulent flows in rectangular channels: A parametric study. *Journal of Applied Research in Water and Wastewater*, 1(2), 2014. ISSN 51-56.

W.H. Hager and K. Hutter. Approximate treatment of plane channel flow. *Acta Mechanica*, 51:21–48, 1983.

W. Hereman. *Shallow Water Waves and Solitary Waves*, pages 8112–8125. 01 2009. doi: 10.1007/978-0-387-30440-3_480.

J.M. Hervouet and A. Petitjean. Malpasset dam-break revisited with two-dimensional computations. *Journal of Hydraulic Research*, 37(6):pp. 777–788., 1999.

C. W. Hirt and B. D. Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39, December 1981.

H.Tennekes and J. L. Lumley. A first course in turbulence. *MIT Press*, 1972.

R.I Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, 62(1):40 – 65, 1986. ISSN 0021-9991. doi: https://doi.org/10.1016/0021-9991(86)90099-9. URL `http://www.sciencedirect.com/science/article/pii/0021999186900999`.

H. Jasak. Wikki ltd, 2004. URL `http://www.wikki.co.uk/`.

W. P. Jones and B. E. Launder. The prediction of laminarization with a two-equation model of turbulence. *International Journal of Heat and Mass Transfer*, 15 (2):301–314, 1972a.

W.P Jones and B.E Launder. The prediction of laminarization with a two-equation model of turbulence. *International Journal of Heat and Mass Transfer*, 15(2):301–314, 1972b. ISSN 0017-9310. doi: https://doi.org/10.1016/0017-9310(72)90076-2. URL `http://www.sciencedirect.com/science/article/pii/0017931072900762`.

N. D. Katopodes. *Free-Surface Flow: Environmental Fluid Mechanics.* Matthew Deans, August 2018. ISBN: 978-0-12-815489-2.

S. Keye. Fluid-structure coupled analysis of a transport aircraft and flight-test validation. *Journal of Aircraft*, 48(2):381–390, 2011.

F. Kilanehei, S. T. O. Naeeni, and M. M. Namin. Coupling of 2dh-3d hydrodynamic numerical models for simulating flow around river hydraulic structures. *World Applied Sciences Journal*, 15(1):63–77, 2011.

S. N. Kuiry, D. Sen, and P. D. Bates. Coupled 1d-quasi-2d flood inundation model with unstructured grids. *Journal of Hydraulic Engineering*, 136(8):493–506, 2010.

B. E. Launder, G. J. Reece, and W. Rodi. Progress in the development of a reynoldsstress turbulence closure. *Journal of Fluid Mechanics*, 68 (3):537–566, 1975.

J. Leandro, A. S. Chen, S. Djordjevic, and D.A. Savic. Comparison of 1d/1d and 1d/2d coupled (sewer/surface). *Journal of Hydraulic Engineering*, 135(6):495–504, 2009.

M. A. Leschziner. *Reynolds-Averaged Navier-Stokes Methods.* American Cancer Society, 2010. ISBN 9780470686652. doi: 10.1002/9780470686652.eae054.

Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems.* Cambridge Texts in Applied Mathematics. Cambridge University Press, August 2002.

OpenCFD Ltd. Openfoam: Documentation, 2017-2019. URL `https://www.openfoam.com/documentation/guides/latest/doc/guide-overset.html`.

M. Manhart. A zonal grid algorithm for dns of turbulent boundary layer. *Computers and Fluids*, 33(3):435–461, 2004.

R. Manning. On the flow of water in open channels and pipes. *Transactions of the Institution of Civil Engineers of Ireland*, 20:161–207, 1981.

G. D. Matthew, editor. *higher order one-dimensional equations of potential flow in open channels*, volume 2, London, 1991. Institution of Civil Engineers.

Academic Research Mechanical. Ansys, 2017. URL `https://www.ansys.com/`.

F. Mintgen. *Coupling of Shallow and Non-Shallow Flow Solvers - An Open Source Framework.* PhD thesis, Technische Universität München, 2017.

F. Mintgen and M. Manhart. A bi-directional coupling of 2d shallow water and 3d reynolds-averaged navier-stokes models. *Journal of Hydraulic Research*, 56(6):771–785, 2018. doi: 10.1080/00221686.2017.1419989.

Munich Re Group. Annual review: Natural catastrophes 2004. Technical report, M unchener R uck Munich Re Group, February 2005.

E. M. Murman and J. A. Krupp. Solution of the transonic potential equation using a mixed finite difference system. In *Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics*, pages 199–206. Springer, 1971.

C.L. Navier. Memoire sur les lois de l'équilibre et du mouvement des corps solides élastiques (memorandum on the laws of equilibrium and motion of elastic solids. Annales de Chimie et de Physique, May 1821. 375-393.

I. Nezu and H. Nakagawa. Turbulence in open-channel flows. *Balkema*, 1993.

J. E. O'Connor and J. E. Costa. The world's largest floods, past and present: their causes and magnitudes, 2004. U.S. Geological Survey Circular 1254.

D. Palermo, I. Nistor, T. Al-Faesly, and A. Cornett. Impact of tsunami forces on structures: The university of ottawa experience. *Science of Tsunami Hazards*, 32, 09 2012.

Suhas V. Patankar. *Numerical Heat Transfer and Fluid Flow.* Taylor & Francis, 1981.

C. Pedersen, E. Ver Loren van Themaat, L. McGuffin, J. C Abbott, T. A Burgis, G. Barton, L. Bindschedler, X. Lu, T. Maekawa, R. Weßling, R. Cramer, H. Thordal-Christensen, R. Panstruga, and P. Spanu. Structure and evolution of barley powdery mildew effector candidates. *BMC Genomics*, 13(1)(694), 2012. doi: 10.1186/1471-2164-13-694.

N. Peller, A.L. Duc, F. Tremblay, and M. Manhart. High-oder stable interpolation for immersed boudary methids. *Int. J. Num. Meth. Fluids*, 52:1175–1193, 2006.

N. Perovic, B.M. Ginting, R.-P. Mundani, H. Rank, E. Zeng, F. Mintgen, and M. Manhart, editors. *HPC simulations of large flooding events: Coupled 2D shallow water and 3D free surface Navier-Stokes models*, Rhodes, 2017. 8rd International conference on Coupled Problems in Science and Engineering,Rhodes, Greece.

E. Plate. Flood risk and flood management. *Journal of Hydrology*, 267:2–11, 10 2002. doi: 10.1016/S0022-1694(02)00135-X.

L. Prandtl. Bericht uber untersuchungen zur ausgebildeten turbulenz. *Zeitschrift für angewandte Mathematik und Mechanik*, 5 (2), 1925.

P. Qi and Y. Hou. A 2d/3d coupled model for wave forces on moored ships in a harbor. *Journal of Hydrodynamics*, 5(5):633–639, 2004.

A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations.* Oxford Science Publications, 1999.

A.K. Rastogi and W. Rodi. Prediction of heat and mass transfer in open channels. *ASCE Jounal of the Hydrulics Division*, 104(3):397–420, 1978.

O. Reynolds. An experimental investigation of the circumstances which determine whether the motion of water shall be direct or sinuous, and of the law of resistance in parallel channels. *Philosophical Transactions of the Royal Society of London*, 174, 1883.

W. Rodi. A new algebraic relation for calculating the reynolds stresses. *Zeitschrift für angewandte Mathematik und Mechanik*, 56:219–221, 1974.

W. Rodi. Turbulence models and their applications in hydraulics: A state-of-the-art-review, 1993. Balkema.

J. Roenby, H. Bredmose, and H. Jasak. *IsoAdvector: Geometric VOF on general meshes.* 01 2018. ISBN 978-3-319-60846-4.

H. Rusche. *Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions.* PhD thesis, Imperial College, London, 2002.

P. G. Saffman and D. C. Wilcox. Turbulence-model predictions for turbulent boundary layers. *AIAA Journal*, 12 (4):541–546, 1974.

S. Djordjević ; A. J. Saul ; G. R. Tabor ; J. Blanksby ; I. Galambos ; N. Sabtu ; G. Sailor. Experimental and numerical investigation of interactions between above and below ground drainage systems. *Water Sci Technol*, 67(3):535–542, 2013.

Inc. Santa Fe, NM: Flow Science. Flow-3d, 2019. URL `https://www.flow3d.com`.

S. Sarjamee, I. Nistor, and A. Mohammadian. Large eddy simulation of extreme hydrodynamic forces on structures with mitigation walls using openfoam. *Natural Hazards*, 85: 1689–1707, 10 2017. doi: 10.1007/s11069-016-2658-5.

A. Sayma. *Computational Fluid Dynamics.* BookBoon, 2009. ISBN-13: 9788776814304.

J. Schanze. Flood risk management - a basic framework. In J. Schanze, E. Zeman, and J. Marsalek, editors, *Flood Risk Management: Hazards, Vulnerability and Mitigation Measures*, pages 1–20, Dordrecht, 2006. Springer Netherlands.

S. Schlauß and M. Grottker, editors. *Coupling Process for 1D-2D Numerical Flash Flood Simulation: A Parameter Study of Involved Variables for Gullies and Manholes*, 2016. IAHR IJREWHS 2016.

Thorsten Schwarz, Frank Spiering, and Norbert Kroll. Grid coupling by means of chimera interpolation techniques. pages 1–12, 06 2010.

F. Serre. Contribution to the study of permanent and non-permanent flows in channels. *Houille Blanche*, (3):374–388, 1953. doi: 10.1051/lhb/1953034.

F. Simões. Finite volume model for two-dimensional shallow environmental flow. *Hydraulic Engineering*, 137(2):173–182, 2011.

N. S. Sivakumaran, T. Tingsanchali, and R. J. Hosking. Steady shallow flow over curved beds. *Journal of Fluid Mechanics*, 128:469–487, 1983. doi: 10.1017/S0022112083000567.

P. R. Spalart and S. R. Allmaras. A one-equation turbulence model for aerodynamic flows. *Recherche Aerospatiale*, 1, 1994.

J. L. Steger, F. C Dougherty, and J. A. Benek, editors. *A chimera grid scheme*, June 1983. IN: Advances in grid generation; Proceedings of the Applied Mechanics, Bioengineering, and Fluids Engineering Conference.

A. Stephan, F. Holz apfel, and S. Zholtovski. The effect of gusts on aircraft wake vortices. *Aircraft Engineering and Aerospace Technology: An International Journal*, 89(5):692–702, 2017.

G.G. Stokes. On the theories of the internal friction of fluids in motion, and of the equilibrium and motion of elastic solids. Cambridge University Press, 1880. 75-129.

E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics.* A Practical Introduction. Springer, 2009. ISBN 978-3-540-49834-6.

A. Valiani, V. Caleffi, and A. Zanni. Case study: Malpasset dam-break simulation using a two-dimensional finite volume method. *Journal of Hydraulic Engineering*, 128(5):pp. 460–472., 2002.

H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method.* Pearson Education, 2007.

H. Weller, C. Greenshields, and J. Collings. Cfd direct ltd, 2015. URL `https://cfd.direct/`.

H.G. Weller. Derivation, modeling, and solution of the conditionally averaged two-phase flow equations. Technical report, Nabla Ltd, 2002. Technical Report TR/HGW/02.

H. Yeh. Maximum fluid forces in the tsunami run-up zone. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 132:496–500, 2006.

H. Yeh. Design tsunami forces for onshore structures. *Journal of Disaster Research*, 2(6): 531–536, 2007.

H. Zeng, L. Grbčić, I. Lučin, and L. Kranjčević. Mesh creation for realistic terrain cases for shallowfoam - 2d openfoam solver. In *Proceedings of the 29th DAAAM International Symposium*, pages 1065–1070, 2018.

X. Zhang and W. Bao. Modified saint-venant equations for flow simulation in tidal rivers. *Water Science and Engineering*, 5:34–45, March 2012.

Appendix A.

Type of Shallow Water Equations

A.1. Unsteady 2D SWE

The fully Shallow Water equations system is a nonlinear second-order PDE. In our study, we consider an incompressible and inviscid fluid. The bottom friction is treated as a body force. Thus the SWE turns into a nonlinear first-order PDE, which can be written in the following form in the case of two space dimensions:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} + \mathbf{B}\frac{\partial \mathbf{U}}{\partial y} = \mathbf{F} \tag{A.1.1}$$

with

$$\mathbf{U} = (u, v, h)^T$$

$$\mathbf{F} = (F_x, F_y, 0)^T$$

$$\mathbf{A} = \begin{pmatrix} u & 0 & g \\ 0 & u & 0 \\ h & 0 & u \end{pmatrix}, \ \mathbf{B} = \begin{pmatrix} v & 0 & 0 \\ 0 & v & g \\ 0 & h & v \end{pmatrix}$$

Equations (2.2.22) are the unsteady 2D-SWE. Here we have 3 independent variables which makes it is complicated to solve. A simple way is to solve it on the $x - t$ plane and $y - t$ plane respectively.

For the $x - t$ plane,

$$\tfrac{\partial \mathbf{U}}{\partial t} + A\tfrac{\partial \mathbf{U}}{\partial x} = \mathbf{F},\ \mathbf{C} = \mathbf{A}.$$

$$|\mathbf{C} - \lambda \mathbf{I}| = \begin{vmatrix} u-\lambda & 0 & g \\ 0 & u-\lambda & 0 \\ h & 0 & u-\lambda \end{vmatrix} = 0$$

upon expansion, resulting in

$$(u-\lambda)^3 - c^2(u-\lambda) = (u-\lambda)[(u-\lambda)^2 - c^2] = 0$$

with $c = \sqrt{gh}$

We get 3 distinct real eigenvalues, which means that the system is hyperbolic.

$$\lambda_1 = u, \ \lambda_{2,3} = u \pm c$$

The same holds for the $y - t$ plane. Hence the system of equation (A.1.1) is hyperbolic.

A.2. Steady 2D SWE

Equations (A.1.1) reduces to equation (A.2.1) if we consider the steady case, which means that time-derivative term vanishes. The problem is now a quasi-linear system of partial differential equations of the first order in two independent variables.

$$\mathbf{A}\frac{\partial \mathbf{U}}{\partial x} + \mathbf{B}\frac{\partial \mathbf{U}}{\partial y} = \mathbf{F} \tag{A.2.1}$$

The equation above can transform to:

$$\frac{\partial \mathbf{U}}{\partial x} + \mathbf{C}\frac{\partial \mathbf{U}}{\partial y} = \mathbf{F} \tag{A.2.2}$$

with

$$\mathbf{C} = \mathbf{A}^{-1}\mathbf{B}$$

NOTE If $\mathbf{A}$ is an invertible matrix, then $|\mathbf{A}| = u(u^2 - c^2) \neq 0$. Thus $u \neq 0$ and $u^2 - c^2 \neq 0$.

$$\mathbf{A}^{-1} = \frac{\mathbf{A}^*}{|\mathbf{A}|}$$

Then

$$|\mathbf{C} - \lambda \mathbf{I}| = \begin{vmatrix} \frac{uv}{u^2-c^2} - \lambda & -\frac{c^2}{u^2-c^2} & -\frac{gv}{u^2-c^2} \\ 0 & \frac{v}{u} - \lambda & \frac{g}{u} \\ -\frac{hv}{u^2-c^2} & \frac{hu}{u^2-c^2} & \frac{uv}{u^2-c^2} - \lambda \end{vmatrix} = 0$$

upon expansion, resulting in

$$(\tfrac{v}{u} - \lambda)(u^2 - c^2)[\lambda^2(u^2 - c^2) - 2\lambda uv + v^2 - c^2] = 0$$

We get 3 eigenvalues:

$$\lambda_1 = u,\ \lambda_{2,3} = \tfrac{uv \pm c\sqrt{u^2+v^2-c^2}}{u^2-c^2}$$

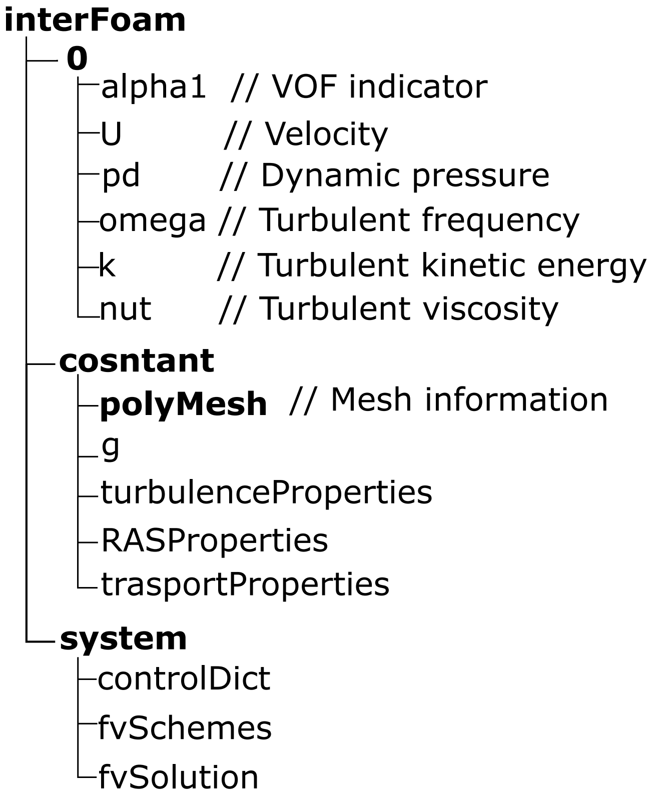
According to the definition, the equation (A.2.1) is hyperbolic if is has three distinct real eigenvalues ($u^2 + v^2 - c^2 > 0$); is parabolic if it has less then three distinct real eigenvalues ($u^2 + v^2 - c^2 = 0$); and is elliptic if it has complex eigenvalue ($u^2 + v^2 - c^2 < 0$).

Appendix B.

OpenFOAM directory structure

All simulated applications are required to have three compulsory directories: `0`, `constant` and `system`. Computing fields data for the 0-timestamp including boundary conditions are stored and specified in `0`, `constant` contains geometry information as well as constant properties such as the fluid density and the gravity acceleration; and `system` provides stipulated solution techniques and general controls of running time and recording time. Additional directories can be generated by OpenFOAM utilities.

B.1. `interFoam` simulation structure



```
interFoam
├─0
│  ├─alpha1  // VOF indicator
│  ├─U       // Velocity
│  ├─pd      // Dynamic pressure
│  ├─omega   // Turbulent frequency
│  ├─k       // Turbulent kinetic energy
│  └─nut     // Turbulent viscosity
├─cosntant
│  ├─polyMesh  // Mesh information
│  ├─g
│  ├─turbulenceProperties
│  ├─RASProperties
│  └─trasportProperties
└─system
   ├─controlDict
   ├─fvSchemes
   └─fvSolution
```

Figure B.1.1: `interFoam` structure tree with $k - \omega\ SST$ turbulence model

As shown in figure B.1.1, in directory `0` all necessary computing fields are prescribed with initial and boundary conditions, of which alpha1 is the free surface indicator function, U is the velocity, pd stands for the dynamic pressure calculated via equation (3.4.2) and turbulent parameters *i.e.* omega, k and nut if $k-\omega\ SST$ turbulence model is applied. The constant settings are defined under constant directory such as the mesh information stored under constant/polyMesh, the chosen turbulence model and the constant physical properties for the application concerned. A system directory is requisite for setting parameters associated with the solution procedure itself. To fulfil a stability criterion, the PISO algorithm requires a small CFL which has to be no larger than one. Thus a maxCo is needed and all other run control parameters *e.g.* start/end time, time step and so on are set under system/controlDict. Discretisation schemes and the equation solvers are determined under system/fvSchemes and system/fvSolution respectively.

B.2. shallowFoam simulation structure

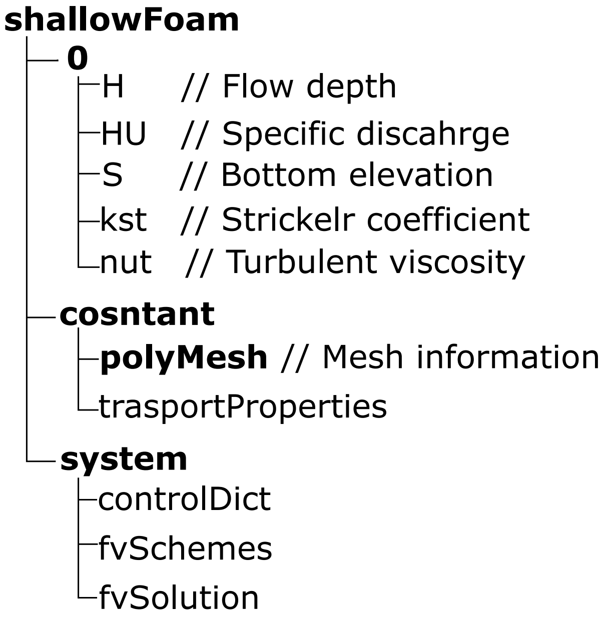
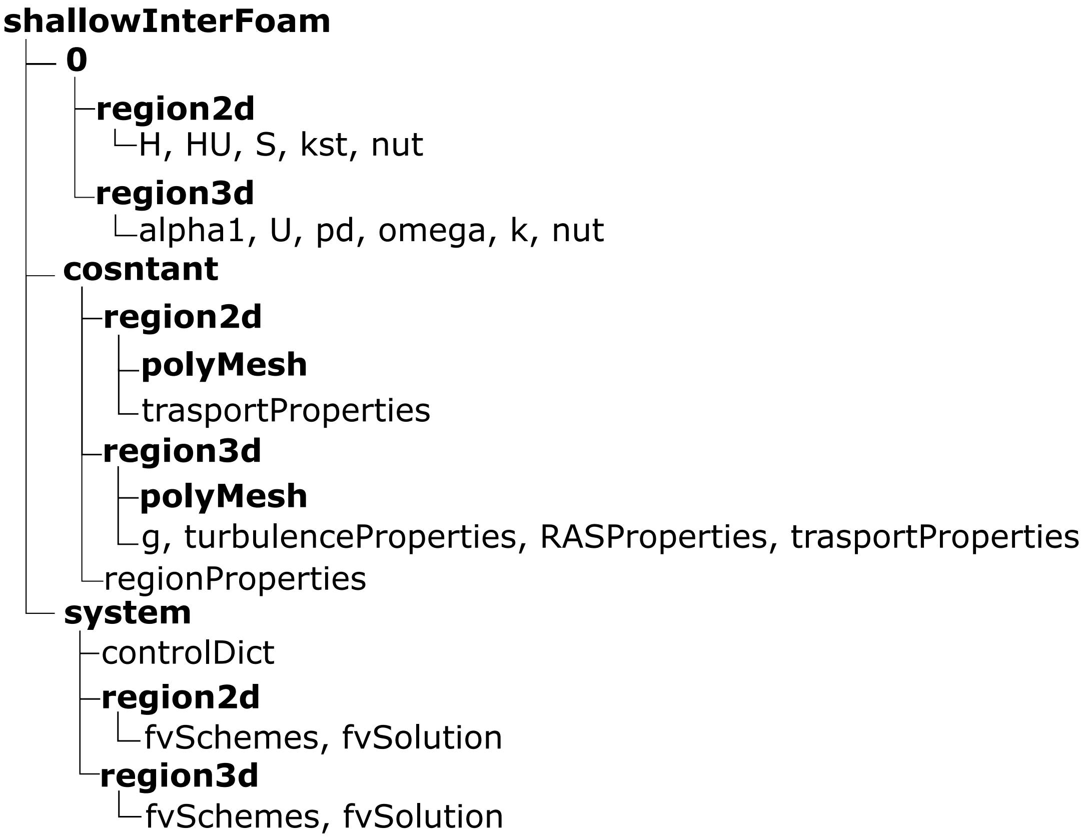


Figure B.2.1: shallowFoam structure tree

Similar directory tree of shallowFoam cases exhibited in figure B.2.1 displays the different requirement from interFoam cases. In accordance with the governing equations, five simulation fields: H, HU, S, kst, nut stands for flow depth, specific discharge, bottom elevation, Strickler value, turbulent viscosity respectively are indispensable for setting up a shallowFoam case. There is no more turbulent properties file in constant because only eddy viscosity model is covered as the 2D turbulent model.

In OpenFOAM the mesh are always have three dimensions. In shallowFoam cases the vertical direction mesh are not taken into account that a uniform height of 1 [m] in z-direction is obligatory. That is the reason why the bottom elevation field S is introduced to represent the bottom geometry. Boundary conditions for all fields in z-direction are set to the empty boundary condition.

B.3. shallowInterFoamOL simulation structure



```
shallowInterFoam
├─ 0
│   ├─region2d
│   │   └─H, HU, S, kst, nut
│   └─region3d
│       └─alpha1, U, pd, omega, k, nut
├─ cosntant
│   ├─region2d
│   │   ├─polyMesh
│   │   └─trasportProperties
│   ├─region3d
│   │   ├─polyMesh
│   │   └─g, turbulenceProperties, RASProperties, trasportProperties
│   └─regionProperties
└─ system
    ├─controlDict
    ├─region2d
    │   └─fvSchemes, fvSolution
    └─region3d
        └─fvSchemes, fvSolution
```

Figure B.3.1: shallowInterFoam structure tree

As a coalition of interFoam and shallowFoam, every single essential file for setting up the corresponding case is also blended in shallowInterFoam case directory tree. In accordance with the sub solver, elements belongs to the 2D solver are getting to the region2d sub-directory and for which affiliated with the 3D solver are under the region3d sub-directory. Apart from that, regionProperties is in need of specifying the sub-regions name.

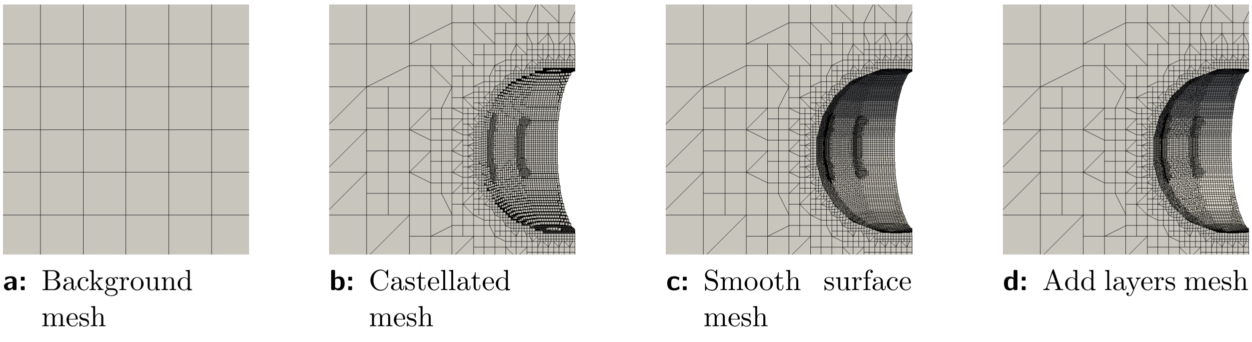
Appendix C.

OpenFOAM Mesh Strucutre

As an integral part of the numerical solution, a mesh is a set of points or cells that each point or cell represents an individual solution of the equation and when combined for the whole network, results in a solution for the entire domain. In another words, a mesh is a discretization of a domain existing in one, two or three dimensions. On basis of FVM, OpenFOAM is designed originally for solving 3D problems so that the definition of a mesh a group of arbitrary polyhedral cells in three dimensions.

The general mesh structure in OpenFOAM is known as **polyMesh**, that for each cell can have an infinite number of faces and no limit on the number of edges applies on each face. This type of mesh is usually called unstructured mesh that offers a wide freedom in mesh generation when dealing with a complex geometry problem.

The default mesh generator in OpenFOAM is `blockMesh`, which is mostly used in simple geometry cases and it can create a regular rectangular mesh domain without many effort. When creating a mesh with complex geometry, another mesh manipulation tool in OpenFOAM, `snappyHexMesh`, is applied. The `snappyHexMesh` utility generates body-fitted 3D meshes by approximately adjusting the background mesh (*i.e.* the background mesh generated by `blockMesh`) to the object surface and shifting the resulting hex or split-hex mesh to the surface. Within execution of `snappyHexMesh`, the mesh refine the appointed region, approximately re-adapt to the surface and morphing the resulting split-hex mesh to the surface. Additional step is possible to shrink back the resulting mesh and insert cell layers. Mesh generation procedures for a half sphere hollow in a cubic domain are exhibited in figure C.0.1.



a: Background mesh

b: Castellated mesh

c: Smooth surface mesh

d: Add layers mesh

Figure C.0.1: 3D mesh generation for a half sphere hollow in a cubic domain

The background mesh in figure C.0.1 a is obtained by blockMesh and the rests in figure C.0.1 are produced by snappyHexMesh. The requirements and the technique of snappyHexMesh is shown as follows:

1. Background mesh generation (blockMesh).
2. Surface data files (in STL or OBJ format) of the target geometry supply.
3. Mesh refinement and splitting at feature edges and surfaces on basis of the background mesh (snappyHexMesh).
4. Cells removal of which lies out of the specific region (snappyHexMesh).
5. Snapping the resulting mesh to better fit the surface (snappyHexMesh).
6. Adding additional layers of hexahedral cells aligned to the boundary surface (snappyHexMesh).

The following sections will take the same geometry, a culvert, as an example for mesh representation in different dimensions, more specifically for 3D and 2D mesh generation.

C.1. 3D solver mesh

The side view of the 3D mesh in figure C.1.1 demonstrates explicitly the outline of the geometry, a culvert in this example. Multiple layers of cells are constructed in all directions of which the boundary faces define the geometry information, *e.g.* bottom elevation.

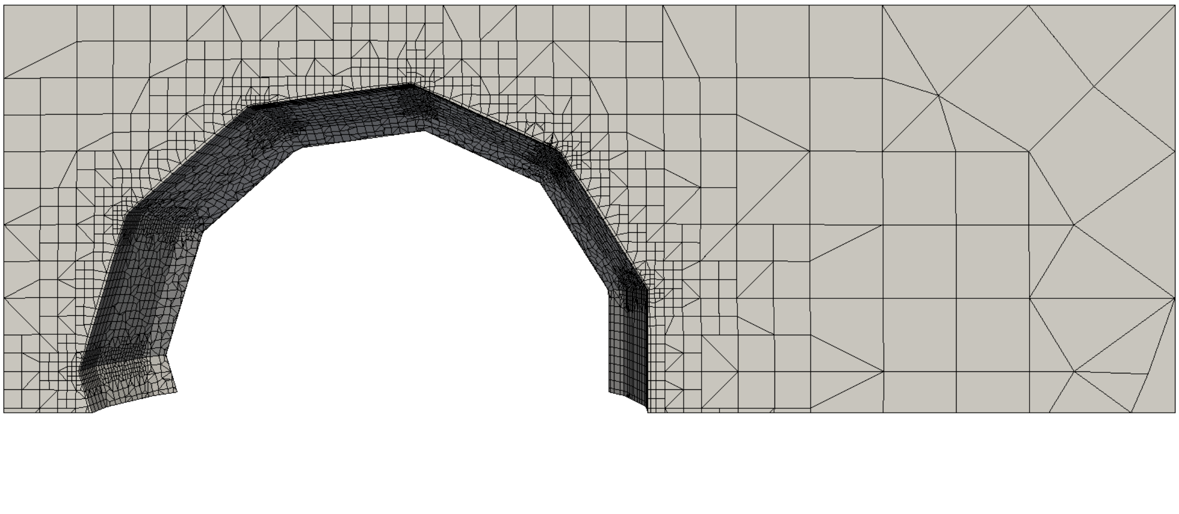


Figure C.1.1: Side view of 3D mesh representation

C.2. 1D/2D solver mesh

The same case will be illustrated in a different way of mesh structure when a lower dimension solver applies. The 2D mesh used in OpenFOAM still has three dimensions in formal but only depicts the geometry in two dimensions denoting the third dimension contains no geometry information. Comparing to the 3D mesh side view of the culvert in figure C.1.1, figure C.2.1 has an absolute flat bottom with only one layer in the vertical direction. It implies that the bottom elevation data of the case is missing in mesh representation. Since no message is carried in the vertical direction, only one cell is required along this direction and the boundary conditions for those, which are perpendicular to the vertical direction, are set as an empty boundary. Analogically, if 1D solver applies, two dimensions in the mesh have one layer of cells and the connecting boundary condition are empty.

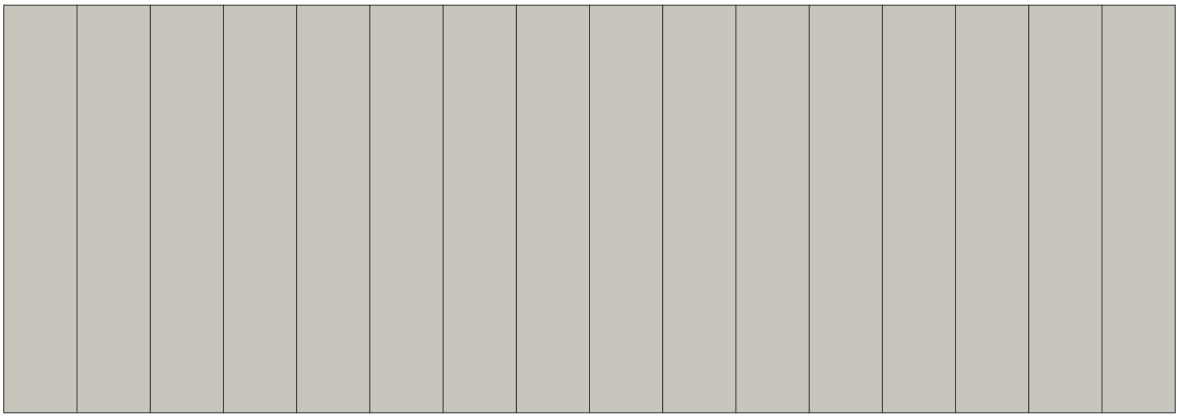


Figure C.2.1: Side view of 2D mesh representation

Appendix D.

Wave celerity derivation

The wave speed relative to the flow velocity v_{w} is called wave celerity. The absolute wave velocity is $v \pm c$, with positive sign for wave travelling downstream, and the negative sign is used for wave travelling upstream.

We assume that before discontinuous wave arrives, the flow in an open channel to be steady. The positive wave propagate downstream with a surge height ζ. The flow attributes before and after the wave arrives are shown in the following Table D.0.1

	before (cross-section $1-1$)	after (cross-section $2-2$)
flow rate	Q_1	Q_2
flow depth	h_1	h_2
water width	B_1	B_2
cross-section area	A_1	A_2
flow velocity	v_1	v_2
pressure force	P_1	P_2

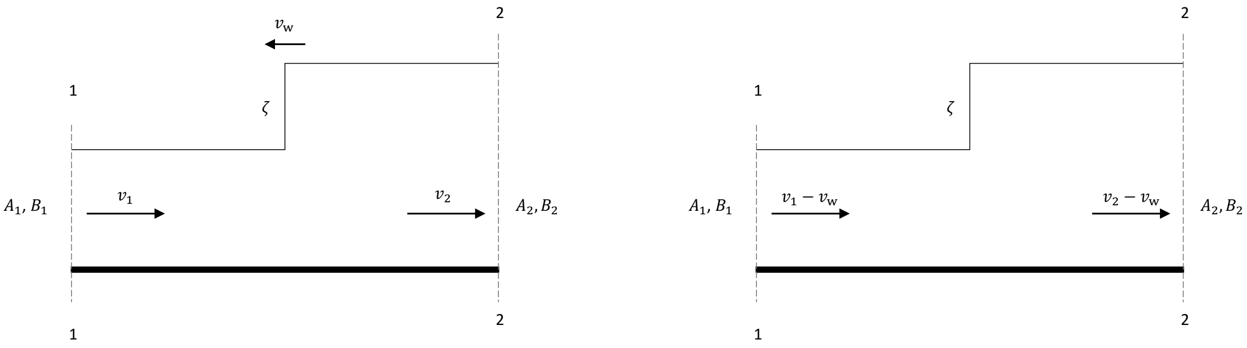
Table D.0.1: Flow attributes before and after wave arrive

Here we only consider the upstream-travelling wave. As shown in figure D.0.1 on the left, the flow velocities in upstream and downstream cross-sections point downstream, while the wave velocity towards upstream. Neglecting the viscosity, this problem can be treated as the flow stays in a moving coordinate system with a constant wave speed v_{w}. Then the relative flow velocity becomes $v_1 + v_{\mathrm{w}}$ and $v_2 + v_{\mathrm{w}}$, respectively on the right of the figure. The surge height $\zeta = h_2 - h_1$.

D.1. Continuity equation for discontinuous wave

Applying the continuity principle for cross-section $1-1$ and $2-2$:

$$(v_1 + v_{\mathrm{w}})A_1 = (v_2 + v_{\mathrm{w}})A_2 \qquad \text{(D.1.1)}$$



a: absolute coordinate system **b:** reletive coordinate system

Figure D.0.1: Flow problem coordinate systems

Rearranging gives:

$$v_1A_1 - v_2A_2 = Q_1 - Q_2 = \Delta Q = v_w(A_2 - A_1) \tag{D.1.2}$$

D.2. Momentum equation

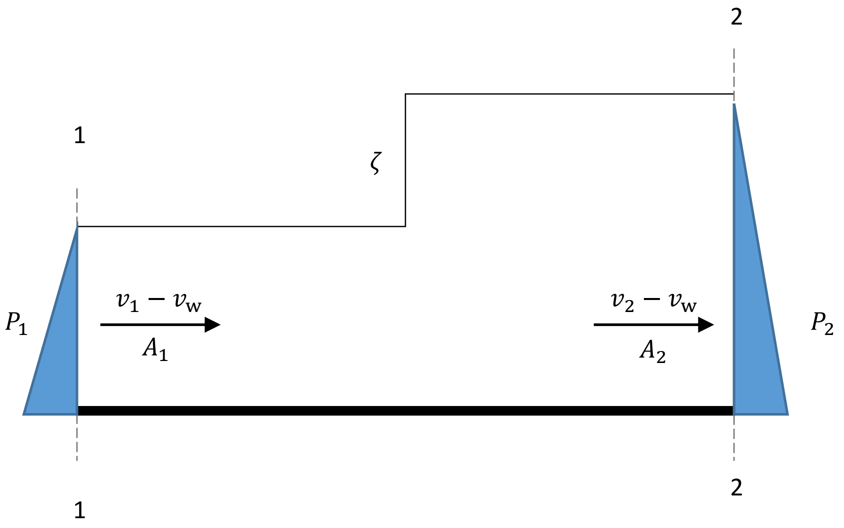


Figure D.2.1: Pressure difference

The same, applying the relative flow momentum equation for cross-section $1 - 1$ and $2 - 2$ by neglecting body force, only pressure force works:

$$P_1 - P_2 = \rho[Q_2(v_2 + v_w) - Q_1(v_1 + v_w)] \tag{D.2.1}$$

Substituting Q_1 and Q_2:

$$P_1 - P_2 = \rho[A_2(v_2 + v_{\mathrm{w}})(v_2 - v_1)] \tag{D.2.2}$$

Hydrostatic pressure distribution in cross-section $1\!-\!1$ and $2\!-\!2$:

$$P_2 - P_1 = \rho g(h_{\mathrm{c}2}A_2 - h_{\mathrm{c}1}A_1) \tag{D.2.3}$$

with $h_{\mathrm{c}2}$ and $h_{\mathrm{c}1}$ are the centroid depth of h_2 and h_1 respectively.

D.3. Wave celerity for uniform rectangular channel

For a uniform rectangular channel: $B_1 = B_2$, $h_{\mathrm{c}1} = \frac{1}{2}h_1$, $h_{\mathrm{c}2} = \frac{1}{2}h_2$, $A_1 = h_1B_1$, $A_2 = h_2B_2$. Equation (D.2.3) can be written as:

$$P_2 - P_1 = \frac{1}{2}\rho g\frac{A_2}{h_2}\zeta(h_1 + h_2) \tag{D.3.1}$$

Putting Equation (D.2.2) and Equation (D.3.1) together:

$$\rho[A_2(v_2 + v_{\mathrm{w}})(v_2 - v_1)] = -\frac{1}{2}\rho g\frac{A_2}{h_2}\zeta(h_1 + h_2) \tag{D.3.2}$$

v_1 can be reformulated from equation (D.1.1):

$$v_1 = \frac{(v_{\mathrm{w}} - v_2)\zeta}{h_1} \tag{D.3.3}$$

Substituting equation (D.3.3) in equation (D.3.2):

$$\frac{h_2(v_2 + v_{\mathrm{w}})^2}{h_1} = \frac{g}{2}(h_1 + h_2) \tag{D.3.4}$$

Rearranging gives the wave speed:

$$v_{\mathrm{w}} + v_2 = \sqrt{\frac{g(h_1 + h_2)h_1}{2h_2}} \tag{D.3.5}$$

Hence wave celerity:

$$v_{\mathrm{w}} = \sqrt{\frac{g(h_1 + h_2)h_1}{2h_2}} \tag{D.3.6}$$