NARSIS

New Approach to Reactor Safety ImprovementS

WP1: Characterization of potential physical threats due to different external hazards and scenarios

Del 1.4 - Flooding impact on industrial facilities via advanced numerical modelling

This project has received funding from the Euratom research and training programme 2014-2018 under Grant Agreement No. 755439.

Project Acronym:	NARSIS		
Project Title:	New Approach to Reactor Safety Improvements		
Deliverable:	Del 1.4 - Flooding impact on industrial facilities via advanced numerical modelling		
Month due:	36	**Month delivered:**	37
Leading Partner:	EDF		
Version:	Final		

Primary Author: EDF : Athanasios MOKOS (LHSV), Damien VIOLEAU (LNHE), Tiphaine LE MORVAN (PERICLES)

Other contributors:

- EDF/LHSV: Rémi CARMIGNIANI
- EDF/LNHE: Agnès LEROY

Deliverable Review:

- **Reviewer #1:** Luca Arpaia (BRGM) **Date:** 14/04/2020
- **Reviewer #2:** James Daniell (KIT) **Date:** 17/07/2020

Dissemination Level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Table of contents

List of Figures

List of Tables

List of Abbreviations

Acronym	Definition	Page
AWAS	Active Wave Absorption System	15
CFL	Courant-Friedrichs-Levy time step coefficient	12
DB	Dynamic Boundaries method	12
FIEC	First Internal EDF Cluster	54
GPU	Graphics Processing Unit	9
GPUSPH	Graphics Processing Unit - Smoothed Particle Hydrodynamics	8
LJ	Lennard-Jones boundaries	12
NGF	Nivellement Général de la France	14
NPP	Nuclear Power Plant	8
SA	Semi-Analytical boundaries	12
SIEC	Second Internal EDF Cluster	44
SPH	Smoothed Particle Hydrodynamics	8
WCSPH	Weakly Compressible SPH	11

1 Executive Summary

The purpose of this deliverable is to apply GPUSPH (Graphics Processing Unit – Smoothed Particle Hyrodynamics) to the modeling of waves impacting a NPP (Nuclear Power Station). The report consists of two parts. These two parts show the application of GPUSPH to complex structures similar to NPPs with high resolution modelling of 3-D wave simulations and can be recommended for use in future NPP designs where computational time is taken into account.

The first part is concerned with the applicability of GPUSPH for modelling vertical propagation along a realistic topography and the accuracy of predicting wave shoaling, breaking, etc. The best practices for creating a 3-D wave simulation using GPUSPH are investigated, focusing on the effect of SPH parameters and creating the geometry of the domain. The simulation is compared to experiments conducted within EDF with particular interest in the overtopping volume and the free surface position.

The second part focuses on the use of GPUSPH in order to simulate wave overtopping in realistic 3-D geometries. The feasibility of simulating realistic 3-D geometries using SPH and the preparation of a sufficiently accurate point cloud for pre-processing is investigated, including digitalization of the topography, preparation of the numerical data and the launch of simulations with periodic waves under several realistic wave conditions. Test simulations are conducted in a section of a generic coastline and in a section of a model dike.

2 Background and Introduction

2.1 Introduction

In the last two decades, coastal settlements in multiple parts of the world have been partially or completely flooded with significant loss of human life and material cost. The causes of these events are diverse, but they inadvertently reinforce the need for understanding and predicting the impact of water waves on coastal and river structures in order to improve the design of coastal defences. These events include hurricanes (e.g. Katrina, USA, 2005), tsunamis (e.g. Tōhoku, Japan, 2011), storm surges and other causes of sea (or river) surface variations.

A particularly important event for the nuclear energy sector was the Fukushima energy accident on 11 March 2011, the second event rated as Level 7 Major Incident in the International Nuclear and Radiological Event Scale after Chernobyl in 1986. Unlike the latter, however, this disaster was due to external factors: a 15-meter tsunami caused by the Mw9.0-9.1 magnitude Tōhoku earthquake flooded the turbine buildings disabling the cooling systems and leading to core meltdowns the following day.

This accident reinforced the need for understanding and predicting the impact of water waves on NPPs. These needs are not only restricted to tsunamis but include wind waves and other causes of sea (or river) surface variations. Of particular interest is an aftereffect of storm surges: an increase in water height, making the installation more vulnerable to wave overtopping. The main objective is to investigate whether there are possible conditions where flooding of the essential components of the NPP may occur.

This research is particularly challenging as the NPPs have complex topographies and varied geometric scales, consisting of several buildings, spread across a the coastline meaning that reproducing the rapid impact and splash-up of a wave (or a wave train) is a complicated task. Lab experiments (*i.e.* scale models) have so far been the optimal way of dealing with this issue. However, the recent advances in numerical simulations have created an attractive alternative for a fraction of the cost.

Full 3-D simulation of the domin is commonly done using Finite Element (FEM) and Finite Volume (FVM) methods, which have a variety of complex sediment and turbulence models. The disadvantage of these models is the need of a mesh, which can be expensive, especially for violent flows where reconstructing it is a necessity. As an alternative, Eulerian meshless methods, which follow the fluid rather than the mesh have emerged as an alternative.

This report will focus on the Smoothed Particle Hydrodynamics (SPH) (Gingold & Monaghan, 1977; Lucy, 1977), method whose meshless nature implicitly captures the nonlinear deformation of violent motions (Colagrossi, et al., 2010), such as wave breaking, and obviates the requirement for expensive meshing (Monaghan, 2005; Violeau & Rogers, 2016). It has already been extensively used for modelling water waves (Dalrymple & Rogers, 2006; Carmigniani, et al., 2019).

SPH has seen use in NPP analyses. It has been validated for use with high wind risk assessment (Hess, et al., 2016), used for simulating sediment flows (Fourtakas, 2015) and heat transfer within the reactor (Pahuamba-Valdez, et al., 2019). More recently, it has been used to evaluate the risk of external flooding (Ryan, et al., 2019; Lin, et al., 2020). The incompressible SPH variant was used in this case, with the particle size being 0.02m.

The main drawback of SPH is the long computational runtime. This has been mitigated by the emergence of GPUs as hardware accelerators, already been widely used to accelerate SPH simulations (Hérault, et al., 2010; Crespo, et al., 2011). In the present work the GPUSPH code (http://www.gpusph.org/), able to run on Multi-GPU systems (Rustico, et al., 2014) is used.

However, before initiating such a complex case, it is necessary to confirm that GPUSPH can capture the relevant phenomena and identify any potential areas an improvement is

necessary. As the main point of interest for the flooding is the overtopping of the coastal defence structures, an investigation on the accuracy of SPH modelling overtopping is needed.

SPH has already been successfully used to model overtopping phenomena (Shao, et al., 2006) (Didier & Neves, 2009), but their geometries are simpler than the coastal defences of NPPs. The recent experiments carried out within EDF measuring overtopping use a complex dike and are more suitable for testing the GPUSPH code.

This comparison between GPUSPH and the experiments is the main focus of the first part of this report, following a brief review of the SPH methodology. A summary of the experimental setup will be given. Of particular interest is the process of creating the domain and the effect of the SPH parameters so that GPUSPH can be easily used in future cases. Results presented will focus on the overtopping volume and the free surface position at various points in the tank.

The second part of the report is a continuation of the application of GPUSPH to wave overtopping, with its application now focusing on realistic geometries and coastlines. For the latter, comparisons to an experiment performed within EDF using a scale model of a coastline bathymetry are performed.

The main focus of this part is on measuring overtopping (if any) on the geometries considered. A secondary focus is on the feasibility of using GPUSPH for a large-scale 3-D industrial case with long runtimes. The version 5.0 of GPUSPH (released on June 13, 2019) and its Release Candidate version will be used in this section. These versions are compatible with the new pre-processing tool, included as a module in SALOME.

The new tool has allowed the introduction of more complex geometries in GPUSPH and a significant part of the report details the treatment and pre-processing of the geometry data to a format that is suitable for use with the GPUSPH code and SALOME in general. The necessary steps and filters for creating a mesh of sufficient quality will also be investigated.

2.2 Methodology

The flows simulated here are described by the Navier–Stokes equations for momentum and continuity (the system is considered adiabatic, so the energy equation is not taken into account), shown in Equation (1):

$$\begin{aligned} \frac{\mathrm{d}\,\mathbf{u}}{\mathrm{d}\,t} &= -\frac{1}{\rho}\nabla p + \nabla\cdot\left[\nu\left(\nabla\mathbf{u}+\nabla\mathbf{u}^{T}\right)\right]+\mathbf{g} \\ \frac{\mathrm{d}\,\rho}{\mathrm{d}\,t} &= -\rho\nabla\cdot\mathbf{u} \end{aligned} \qquad (1)$$

where $\boldsymbol{u}$ is the velocity vector, ρ the fluid density, p the pressure, ν the kinematic viscosity and $\boldsymbol{g}$ an external force. For the purposes of this report, gravity is considered the only external force acting on the system.

As a Lagrangian method, SPH simulates a domain as a collection of particles (Lucy, 1977; Gingold & Monaghan, 1977). Kinematic and thermodynamic quantities for each particle are calculated through a local interpolation with a weighting function, referred to as a smoothing kernel (Monaghan, 2005). The weighting depends on the pairwise particle distance and is controlled by a distance referred to as the smoothing length. Equation (2) shows the SPH approximation of a function A at a particle a with respect to neighbouring particles b:

$$\langle A\rangle_a = \sum_b V_b A_b W_{ab} \qquad (2)$$

where V_b is the volume of particle b and $W_{ab}=W(|\boldsymbol{r}_a-\boldsymbol{r}_b|)$ is the kernel value as a function of the distance between particles a and b, with $\boldsymbol{r}_a$ and $\boldsymbol{r}_b$ the position of the particles, respectively. Following (Violeau, 2012), to derive the SPH approximation of the Navier-Stokes equations,

shown in Equation (3), skew-adjoint forms of the discretised operators for the velocity divergence D_a, pressure gradient $\mathbf{G}_a$ and Laplacian second order derivative L_a are used in order to maintain energy conservation:

$$\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\right)_a = -\frac{1}{\rho_a}\mathbf{G}_a(p) + \mathbf{L}_a(\nu,\mathbf{u}) + \mathbf{g} \text{ where } \mathbf{G}_a(p) = \sum_b V_b(p_a + p_b)\nabla_a W_{ab}$$

$$\text{and } \mathbf{L}_a(\nu,\mathbf{u}) = \nabla\cdot(\nu\nabla\otimes\mathbf{u}) = \sum_b V_b(\nu_b + \nu_a)\frac{\mathbf{u}_{ab}}{|\mathbf{r}_{ab}|}|\nabla_a W_{ab}| \quad (3)$$

$$\left(\frac{\mathrm{d}\rho}{\mathrm{d}t}\right)_a = -\rho_a D_a(\mathbf{u}) \text{ where } D_a(\mathbf{u}) = \sum_b V_b(\mathbf{u}_b - \mathbf{u}_a)\cdot\nabla_a W_{ab}$$

To link the pressure and density, GPUSPH uses the WCSPH formulation with the following equation of state (Batchelor, 1967):

$$p_a = \frac{c_0\rho_0}{\xi}\left[\left(\frac{\rho_a}{\rho_0}\right)^{\xi} - 1\right] \quad (4)$$

where c_0 is the artificial speed of sound for the SPH computation, ξ is the adiabatic index (equal to 7 for water) and ρ_0 is the reference density at the free surface (set here for water as 1000kg/m^3).

Using WCSPH, the water density is allowed to vary by a small amount of 1%, controlled by an artificial speed of sound restricted to 10% of the Mach number determined by the maximum velocity of the particles (Monaghan, 1992). For monochromatic waves, that would be the wave orbital velocity near the bed (Soulsby & Smallman, 1986). However, taking into account the drop of the water after overtopping (a gravity flow), the wave celerity for a shallow-water wave is used instead as it accounts for the hydrostatic pressure.

The weighting function used for the SPH simulations is the Wendland C^2 kernel (Wendland, 1995). It is selected due to its numerical properties, as it allows larger numbers of neighbours and ensures better numerical stability and convergence than the quadratic or the cubic spline kernel while being less computationally expensive than the Gaussian kernel (Dehnen & Aly, 2012).

The weighting function is characterised by its smoothing length *h* whose value is selected to minimise the approximation error of the unity integral in an equidistant particle distribution as well as to mitigate wave damping (De Padova, et al., 2014). Its value (usually linked to particle distance) determines the radius at which the neighbour search is performed.

The smoothing length used here differs depending on the case considered. For the experimental channel a value of 1.87643 × Δ*r,* where Δ*r* is the initial distance among the particles, has been considered as a larger kernel value reduces wave damping (Mokos, et al., 2018).

To simulate realistic topography, the smoothing length is set to be 1.62374 × Δ*r,* as the larger value used for the channel greatly increases the computational cost. The present value was found to maintain minimal damping in a region of two wavelengths away from the coastline while reducing the number of neighbouring particles.

For the simulations in this report, the effect of the viscosity is minimal. This can be shown by investigating the dimensionless Roshko number *Ro*:

$$Ro = StRe = \frac{\alpha_w f\lambda}{\nu} \quad (5)$$

where *St* and *Re* are the Strouhal and Reynolds numbers respectively, α_w the wave amplitude,

f its frequency and *λ* is the wavelength. For the simulations in this paper the Roshko number is typically 2×10^5 (with the kinematic viscosity of fresh water being $1.01 \times 10^{-6}\ m^2s^{-1}$) meaning convective terms have a far greater effect on the waves.

Therefore, the inviscid form of the Navier-Stokes is preferred as the viscous term leads to significant wave decay and reduction of the overtopping volume (as has been shown in (Mokos, et al., 2018)). The lack of a viscous term can however, lead to stability issues, mitigated by using a density correction scheme introducing an upwind Rusanov flux to stabilize the pressure field (Ferrari, et al., 2009). The correction is used with the value of the correction parameter η_F being 0.1 (selected through testing) to reduce the intrinsic numerical viscosity (Mayrhofer, et al., 2013). The updated density equation is shown in Equation (6):

$$\left(\frac{\mathrm{d}\rho}{\mathrm{d}t}\right)_a = -\rho_a D_a(\mathbf{u}) + \eta_F \sum_{b \in F} V_b c_{a,b} \frac{\mathbf{r}_{ab}}{r_{ab}} \rho_{ab} \cdot \nabla W_{ab} \qquad (6)$$

Additional density correction schemes were tested for this study, in particular the density diffusion term by (Molteni & Colagrossi, 2009) and the general correction term by (Brezzi & Pitkaranta, 1984). The results for the latter were found to increase wave dissipation, while the Molteni & Colagrossi term results were found to be very similar to the Ferrari.

These simulations only use the water phase as modelling the air phase in a complex 3D simulation is computationally expensive for the long runtimes considered (Mokos, et al., 2015). A second order predictor-corrector time stepping scheme is used to advance the simulation in time with a CFL number of 0.3 (Violeau & Leroy, 2014).

To model the boundaries of the simulations, GPUSPH includes several options. For the modelling of the channel experiment, the DB method, which models the boundaries as stationary SPH water particles (Crespo, et al., 2007) will be used. Multiple layers of these particles are required so that the kernel of fluid particles is complete with 9 layers used for the channel.

The SA boundaries create a wall renormalisation function, taking into account the truncation of the fluid particles' kernel near the boundary (Ferrand, et al., 2012) (Mayrhofer, et al., 2013). Results are more accurate compared to DB but the computational time is significantly increased (Ferrand, et al., 2012). A CFL coefficient of 0.2 is used.

For the realistic topography, the LJ boundary method (Monaghan, 1994) will be used. Using the LJ boundaries, the wall is discretised using particles; however, only one layer is used unlike DB. This allows the modeling of sharp edges, where the use of multiple layers is not possible. For this method, a repulsive force is imposed between fluid and boundary particles based on the LJ potential (Lennard-Jones, 1924).

Compared to other SPH boundary methods, LJ boundaries are computationally cheap and can be used to represent complex domains, which is beneficial for the real geometries investigated here. However, while wall impermeability can be ensured, fluid particles near the wall can suffer from spurious oscillations as the accuracy of the SPH equations is lessened near the boundaries (Monaghan, 2005). This should not affect the results in the present study where the flow is mainly driven by inertia, pressure and gravity, and because we are not interested in carefully predicting the boundary layer near the walls.

At the moment, SPH does not have an industrial use within NPP safety analyses in France, notably because of its computational cost.

3 Wave Overtopping on Experimental Channel

3.1 Experimental Setup

The experiments used in this report were conducted by EDF and aim to improve understanding of the overtopping volumes in complex configurations of dikes, similar to those in NPPs, where design formulas cannot be used. Irregular waves are beyond the scope of this report.

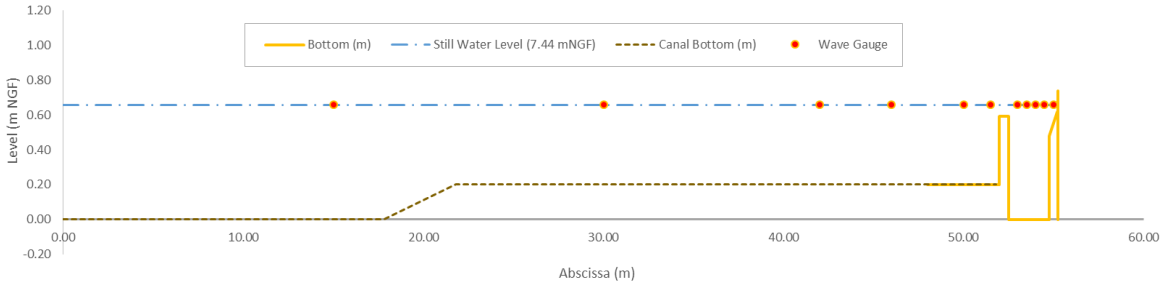


Figure 1: Channel 5 geometry. The dashed brown line denotes existing configuration, the yellow line shows new construction undertaken for this experiment.

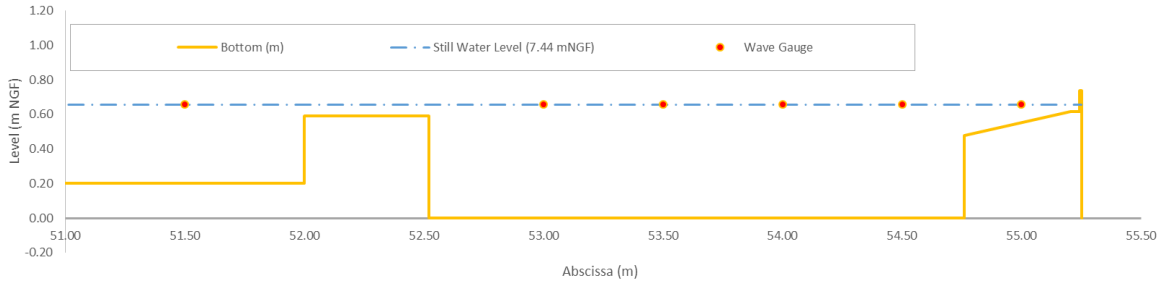


Figure 2: Close-up of the dike area in Channel 5.

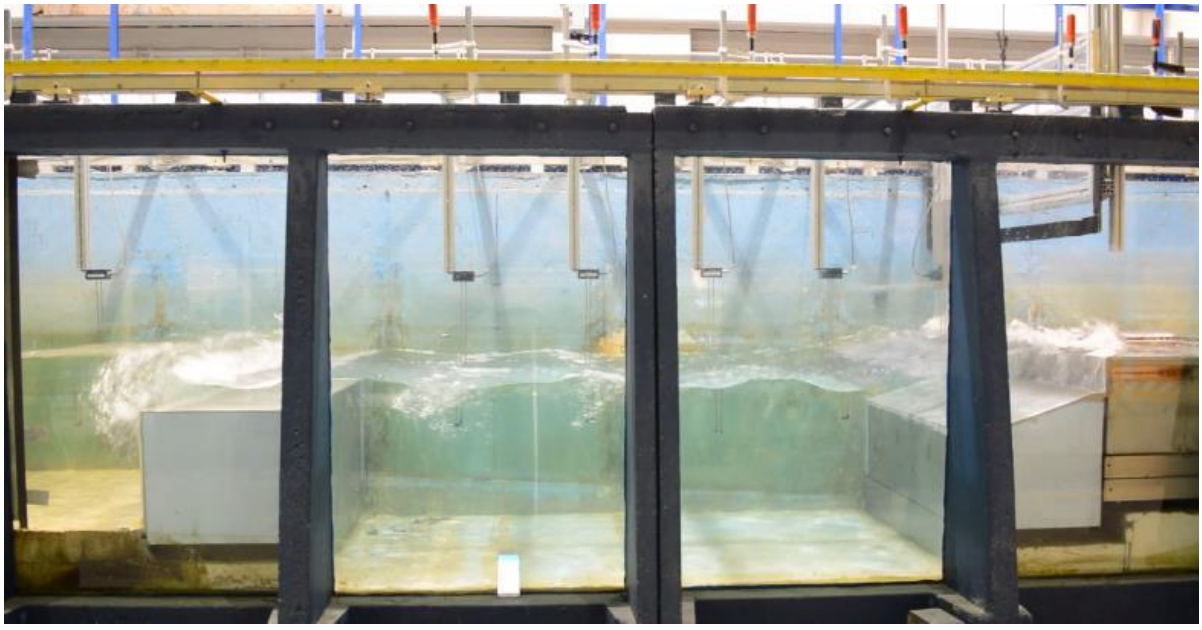


Figure 3: Snapshot of the experiment.

The experimental setup is shown in Figure 1 with a close up of the dike area in Figure 2. The experiment geometry is as follows: the piston is located at abscissa 0.0 and a slope at 17.8 m elevates the tank bottom by 0.2 m. The static water height from the bottom of the tank is H=0.66m or 7.44 m NGF. The structure (shown in yellow line in Figure 1 and Figure 2) built for this experiment consists of two parts, mirroring (simplified) designs used in coastal defence structures: the first dike is a rectangular structure while the second dike consists of a slope leading to a small plateau on which a wall is positioned. A gathering tank is placed behind the wall, allowing measurement of the overtopping volume. The width of the tank is 1.5m and a snapshot of the dike part of the experimental tank can be seen in Figure 3.

The first dike is completely submerged as the water level is considered unnaturally elevated (outside of normal working conditions), in order to simulate potential flooding conditions. For the second dike, the slope and the plateau are underwater, however the wall has a height of 0.74 m and prevents the water from entering the gathering tank. A moving gate has also been installed over the wall, preventing overtopping outside the designated times.

The geometry of the experiment described above will only be partially replicated by the simulation. The numerical analysis focuses on the latter part of the experimental tank (as shown in Figure 2) due to computational restraints. As a result some elements of the experimental tank, like the slope, are removed. More details are presented in Section 3.2.

Table 1: Wave gauge position

Wave Gauge	1	2	3	4	5	6	8	9	10	11	12	13	14	15	16
X(m)	30.01	30.2	30.42	30.67	31.65	42	46	15	50	51.5	53	53.5	54	54.5	55

During the experiment, a number of wave gauges were installed on the tank to identify the position of the free surface. Their position from the wavemaker is shown in Table 1, Figure 1 for all gauges and Figure 2 for gauges 11-16. Gauges 1-5 were used for the analysis of reflected waves.

Table 2: Wave series used in this report

Wave Series	Wave Height (m)	Wave Period (s)	Overtopping time (s)
015	0.24	2	48-68
019	0.29	2	48-68
020	0.04	2	48-68
024	0.26	2	48-68
028	0.23	2	48-68
032	0.19	2	48-68
036	0.14	2	48-68
040	0.09	2	48-68
044	0.11	2	48-68
048	0.22	2	48-68

In the laboratory experiments, 56 series of waves were tested with different wave heights and frequencies for a different number of wave periods. This report will focus on series 048 with a wave period T=2s, wavelength λ=3.941m and mean wave height H=0.22m. The wave height corresponds to the water depth in front of the paddle (D_w=0.66m). The mean wave height after the slope was H_s=0.21m corresponding to mean water depth D_s=0.46m.

In this simulation the moving gate allowed overtopping only between 48s and 68s when a stable system had been established. An investigation of overtopping volume for multiple wave series with same period and overtopping conditions but different wave heights will also be performed. The wave series used here are shown in Table 2.

3.2 Channel Reconstruction in GPUSPH

Before building the tank, an optimum resolution must be identified. For SPH, a useful rule of thumb is that any flow should be represented by, at minimum, 3-4 particles so that, for any particle, multiple neighbouring particles exist at every time step and the kernel is as complete as possible (Mokos et al., 2020). Obviously, increasing the number of neighbouring particles also increases accuracy, but this needs to be balanced with the computational cost. To better reproduce experimental studies for complex phenomena, it is recommended that any flow is represented by 8-10 particles at least.

The domain areas in need of the finest resolution can be identified by Figure 1. They are: i) above the first dike where the distance to the still free surface is 0.07m and ii) above the plateau at the second dike with the distance being 0.04m. Selecting then as an initial particle spacing Δp=0.01m (a number that will also assist in creating the domain geometry) should give sufficient accuracy for a first result, while halving this spacing to Δp=0.005m should allow for greater accuracy.

Regarding the size if the domain, recreating the entirety of Channel 5 for the simulation is prohibitively expensive. Using particle spacing Δr=0.01m the number of particles approaches 5×10^7. Since this is a 3-D simulation, the finer resolution considered (Δr=0.005m) leads to a particle number approximately 4×10^8. Even with the multi-GPU capabilities of GPUSPH, this simulation would require a significant investment of time and computational resources. This investment is further increased by the large runtime of this case (20s for measuring overtopping, in addition to the time required to develop a steady flow).

However, due to the nature of the case that is being investigated, the simulation domain can be reduced, without loss of important information. Indeed, the simulation domain will be restricted to the area most of interest to this report; an area very similar to Figure 2 that contains the two dikes but removes the larger part of the tank including the slope. The piston will then be placed on the elevated bottom. The decrease in channel length is also expected to reduce wave decay, a known problem in SPH (Carmigniani, 2017), although certain formulations (not available in GPUSPH) can circumvent this issue like renormalized operators (Leroy, 2014) (Mayrhofer, 2014).

The optimal position of the piston is unknown, and simulations will be carried with the piston placed at 2λ, 3λ and 4λ from the beginning of the defence structure (approximately at 8m, 12m and 16m respectively). Regardless, the new paddle position is significantly closer to the end of the tank meaning the reflected waves from the structure will need to be addressed. For that reason, the piston will use an AWAS as described by (Carmigniani, et al., 2017). This system corrects the paddle velocity based on the difference of the free surface position in front of the piston compared to the target incident wave.

The final issue to be addressed regarding the simulation domain is the tank width. Due to the use of a piston to create regular waves, large 3-D flow effects are not expected within the tank. Therefore, a slice in the middle of the tank can be simulated instead the full width (1.5m) of the initial tank greatly reducing the computational cost. The use of periodic boundaries is essential as boundary walls would create additional friction, necessitating a wider tank to maintain the wave. The size of that slice should be sufficiently large so particles do not reference themselves through the periodic boundaries for the resolution selected. Simulations with width 0.1m and 0.2m will be considered in this report in order to investigate the presence of any 3-D flow phenomena.

Figure 4: Side view of the tank using DB at the initial position (top) and after 10s (bottom). Blue signifies fluid and red signifies boundaries.

Figure 5: Initial side view of the tank using SA. Blue signifies the fluid and red signifies boundaries.

The process and the tools of creating the domain differ depending on the boundary formulation. For the DB formulation, GPUSPH includes a number of basic geometrical elements that can be combined to create the tank. The *Box* function, which creates a rectangular element, will be used here. That includes the slope at the second dike, with an additional *rotate* function applied to adjust the angle. It is possible to create the slope using multiple *Box* elements, however that creates a 'staircase' with intermittent increases in height which affects the simulation.

The fluid will also be created using the *Box* function. However, for coarser resolutions, the area above the first dike is not created due to the small number of particles. Similar issues occur while creating the fluid above the slope and the plateau with different options for each resolution required. Therefore, a decision was made to include these fluid volumes as a layer above the main body of the water between the piston and the first dike. The water particles need to rearrange to new positions until the velocity field is minimal. A period of 10s has been found sufficient for this process after which the piston can begin its movement. Figure 4 shows the domain created at the initial position and when the piston starts.

For the SA boundaries, pre-processing tools outside GPUSPH will be used to create the case. The geometry will be created in the SALOME platform (https://www.salome-platform.org/) as a mesh. The MGCadSurf algorithm was used in this report. The mesh is then used with the pre-processing tool Crixus (https://github.com/Azrael3000/Crixus) to create the particle distribution for both boundaries and fluid.

A particular point of interest is the periodic boundaries as it is necessary to maintain the same particle distance as the rest of the domain. This can be achieved in two ways: either by adding a Δr margin to the domain limits or by forcing the particles in either side to spawn at $0.5\Delta r$ from the domain limit. This is essential for either of the boundary formulations.
The creation of the piston is identical to the tank walls. The CHRONO library (https://github.com/projectchrono/chrono), already integrated in GPUSPH, is used for its movement. Of note however is the distance of the piston from the bottom of the tank. If the same distance to the rest of the domain (Δr) is used, fluid particles leak through the gap, especially for DB. Therefore, a $0.5\Delta r$ distance is recommended.

3.3 Results

The runtime details of the simulations examined here are shown in Table 3.

Table 3: Simulation data

Simulation	Resolution (m)	Tank Width (m)	Tank Length (m)	Runtime		Number of particles
				Single GPU	4 GPUs	
A	0.02	0.2	2λ	80 mins	-	3.1×10^5
B	0.01	0.1	2λ	11h 6min	3h 36min	9.5×10^5
C	0.005	0.1	2λ	-	45h	5.5×10^6
D	0.01	0.2	2λ	19h 20min	6h 25min	1.9×10^6
E	0.01	0.1	3λ	15h 35min	5h 15min	1.2×10^6
F	0.01	0.1	4λ	20h	6h 40 min	1.45×10^6

The main results to be investigated are the volume of water overtopping and the position of the free surface at the wave gauges. Due to the smaller dimensions of the simulation domain, only wave gauges 6, 8 and 11-16 will be considered. Figure 2 shows the position of gauges 11-16, with 6 and 8 being 8 m and 4 m before gauge 11, as seen in Figure 1 and recorded in Table 4. In the simulation, the waves start breaking after the impact on the first obstacle, though gauges 6,8,11 should nonetheless be affected by reflected waves for the second obstacle. It should be noted that the width of the tank is only 0.2 m for the coarsest resolution as the kernel size is too large for the smaller width.

Table 4: Abscissa of the probes in the experiment and simulations with different tank lengths

Probe	Abscissa (experiment)	Abscissa (simulation)		
		2λ	3λ	4λ
6	42.02	-	2.95	6.90
8	46.02	2.44	6.95	10.90
11	50.05	6.49	11.00	14.95
12	52.99	9.44	13.95	17.90
13	53.49	9.94	14.45	18.40
14	53.99	10.44	14.95	18.90
15	54.49	10.94	15.45	19.40
16	54.98	11.44	15.95	19.90

3.3.1. Free surface position

Since the simulation does not model the entire tank, only the height probes at its latter part will be considered. These include gauges 8 and 11-16, with 8 and 11 being before the first dike and 12-16 between the two dikes. Results will first be shown for Simulation C in Figure 10. The comparison is performed for the latter 20 seconds of the simulation so that a regular wave regime has been established.

The results of Figure 6 show that GPUSPH achieves a close agreement with the experimental results for both the wave phase and the height and trough prediction, even for the complex phenomena shown in gauges 12-16. A minor point of deviation is the overprediction of the wave height in gauge 8, likely caused by the proximity of the measuring probe to the piston meaning that the wave does not have sufficient time to develop. The slight increase in free surface height in gauge 11 during the wave trough is also not predicted by the simulation.

Figure 7 shows the comparison of 3 simulations (A, B, C) with different resolutions with the experiment for gauges 8 and 11-16. Convergence of the results to the experimental as the resolution increases can be observed for probes 12-16 with lesser differences for probes 8 and 11.

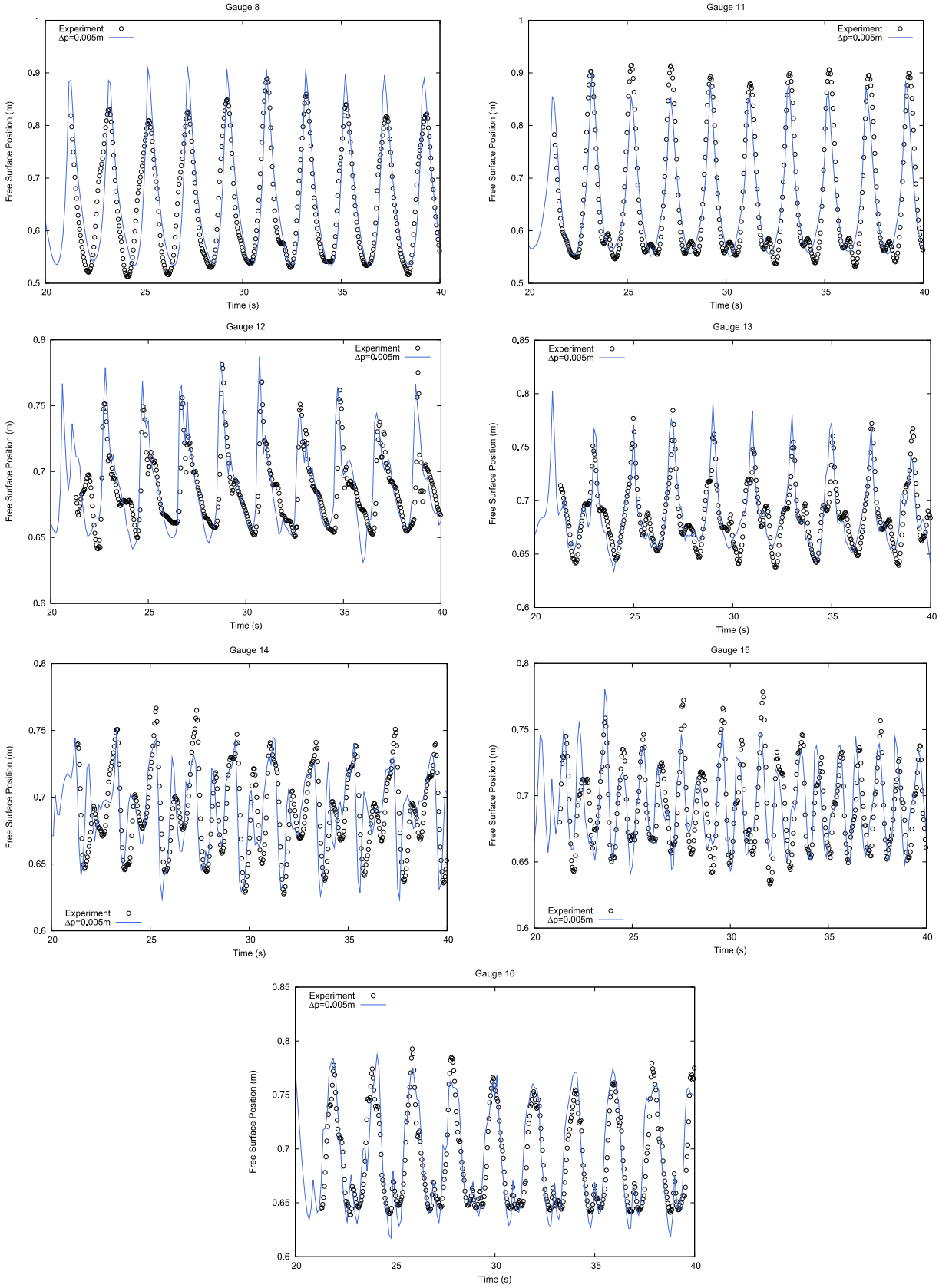


Figure 6: Comparison of the free surface position between the experiment and GPUSPH Simulation C (the finest resolution)

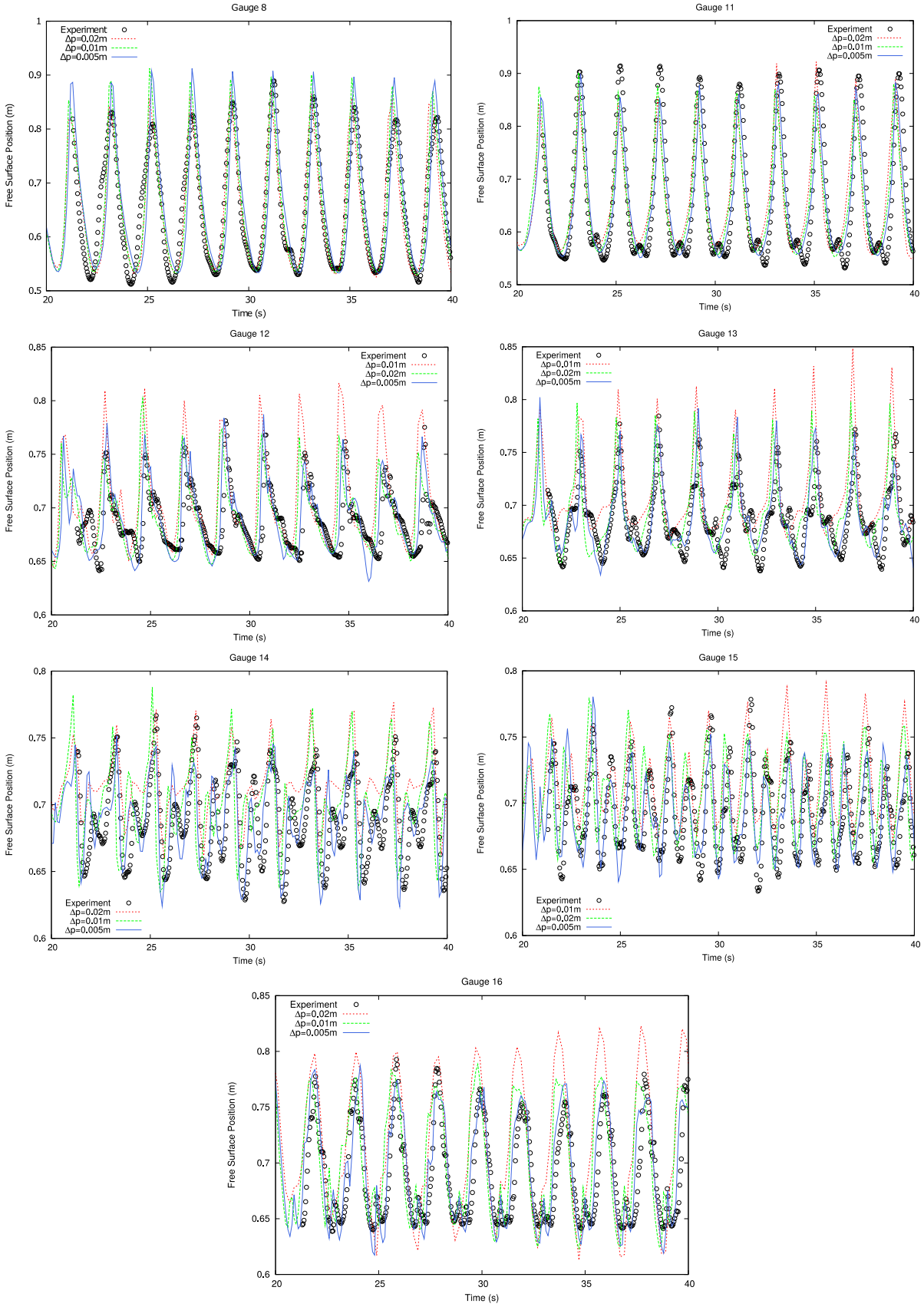
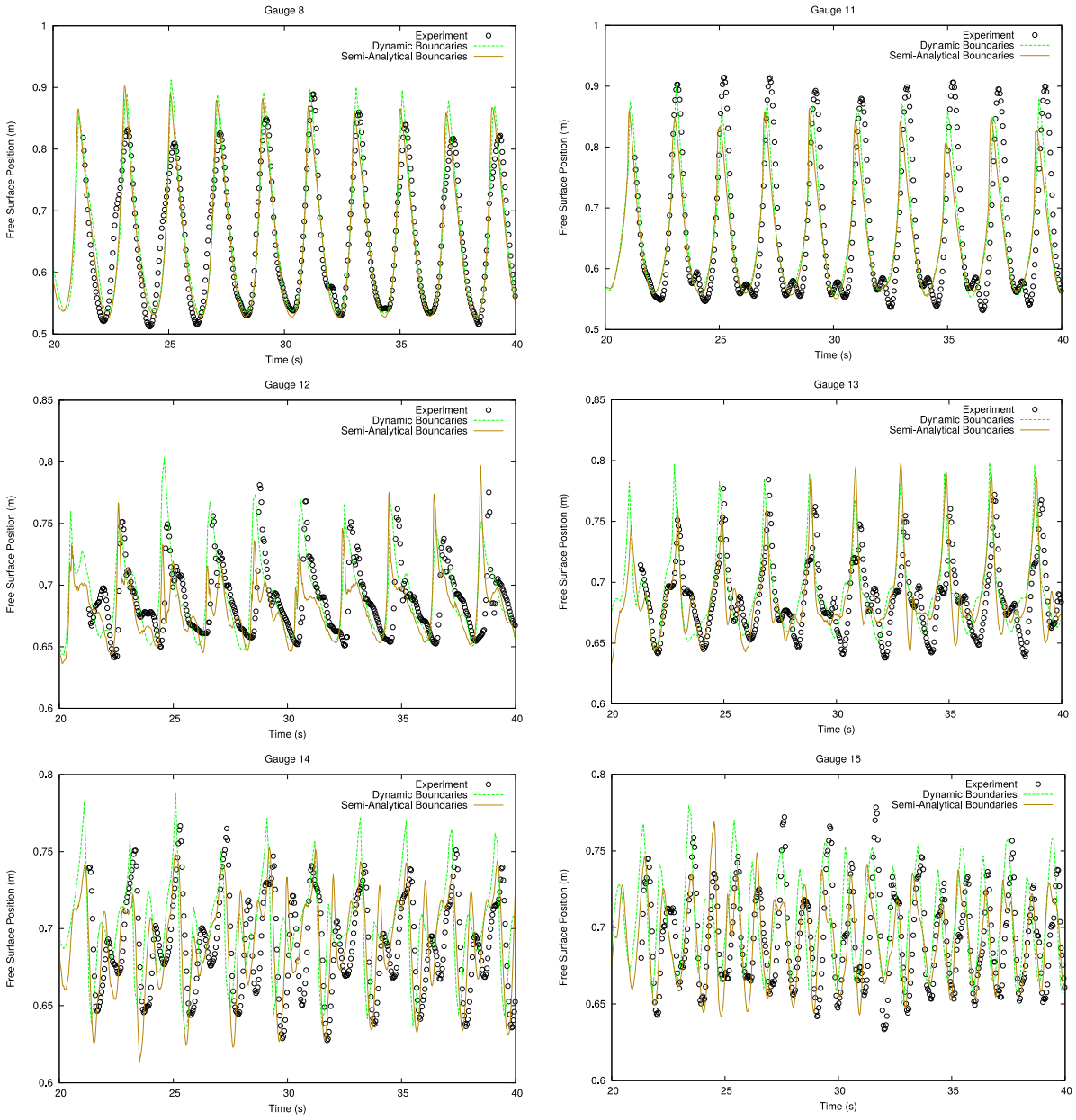


Figure 7: Comparison of the free surface position for GPUSPH Simulations A (Δp=0.02m), B (Δp=0.01m), C (Δp=0.005m) using DB and 3 different resolutions with the experimental results

For gauges 12-16, Simulation A shows a higher water level, underestimating the troughs and overestimating the peaks, the difference from the experiment increases as the simulation progresses. This behaviour is lessened for Simulation B, which follows the finer as can be seen in gauge 16 where the complex wave behaviour of the troughs is predicted. Gauge 14 however, shows that Simulation C produces consistently better results for the more complex characteristics of the flow.

The free surface results for gauge 6 can be seen in Figure 8 for simulations E and F which have longer tank lengths. A dissipation of the wave, especially for its troughs, can be seen for Simulation F (4λ length) while Simulation E (3λ length) is closer to experimental results, although a slight overestimation of the average height, in a matter similar to Gauge 8 in Figure 6 can be seen. Figure 8 also shows a comparison for Simulations B, E, F for gauges 8 and 11. The dissipation is clear for the longest tank with both peaks and troughs underestimated while a lesser effect can be seen for the 3λ simulation.

Figure 8: Comparison of the free surface position for GPUSPH Simulations B (2λ), E (3λ), F (4λ) using DB and different tank lengths with the experimental results



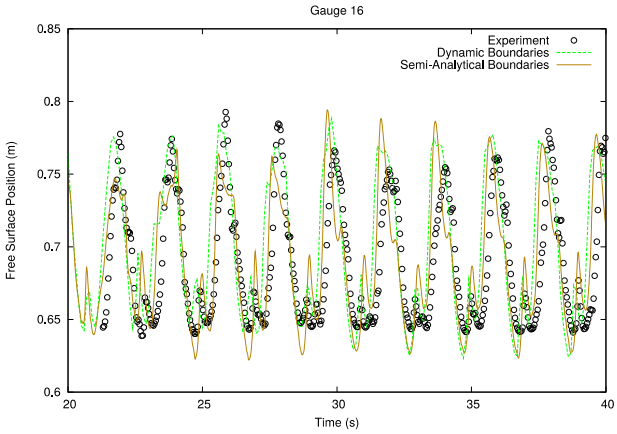


Figure 9: Comparison of the free surface position for GPUSPH Simulation B (Δp=0.01m), using DB and SA boundaries with the experimental results

shows Simulation B with two different SPH boundaries, SA and DB. Minor differences between them can be identified, but they are too inconsistent to determine whether one model has better accuracy. Therefore, considering the computational cost of the SA boundaries is up to four times the cost of DB, the latter are used for the channel simulation.

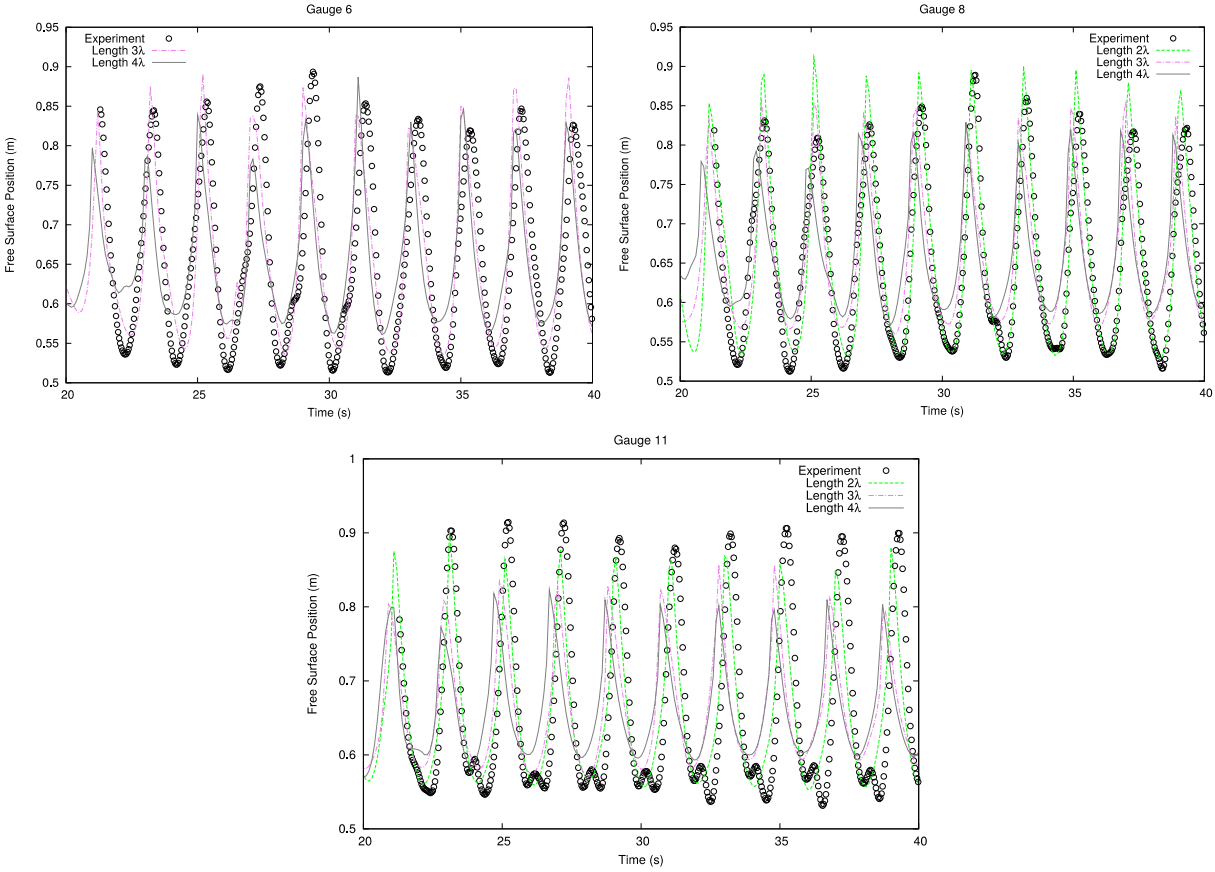


Figure 8: Comparison of the free surface position for GPUSPH Simulations B (2λ), E (3λ), F (4λ) using DB and different tank lengths with the experimental results

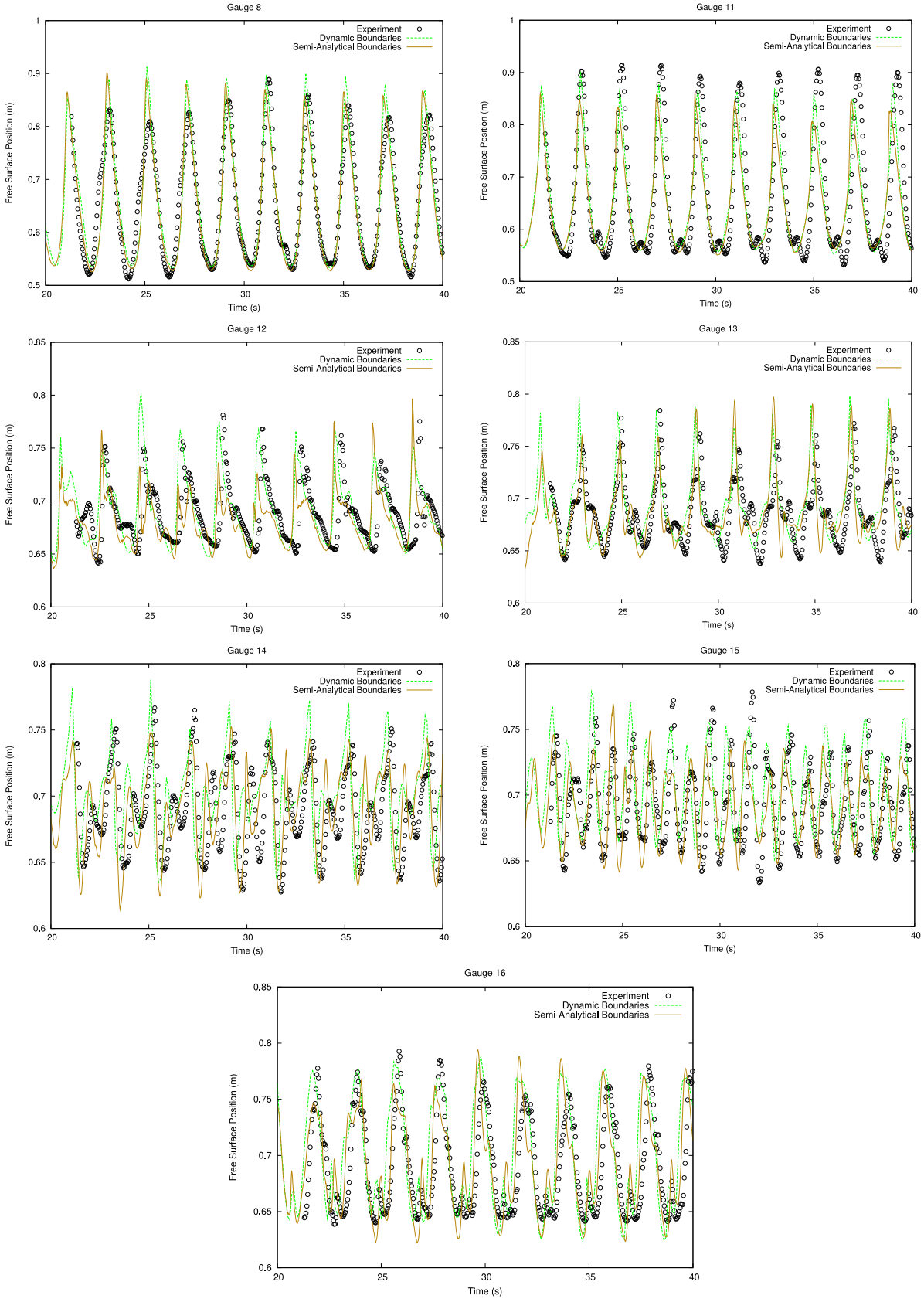


Figure 9: Comparison of the free surface position for GPUSPH Simulation B (Δp=0.01m), using DB and SA boundaries with the experimental results

3.3.2. Overtopping Volume

The data available for the experiment do not include a timeline of the overtopping volume, only the final value, as the measurement only took place when emptying the tank. Since a steady flow regime was achieved in the experiment, an assumption of a linear increase in overtopping volume will be made for the comparison. Similarly, for the OpenFOAM simulation, only 10s of measurement for overtopping are available. It will be assumed that subsequent measurements would have followed a similar pattern.

Figure 10 shows the overtopping volume for Simulations A, B, C (3 different resolutions). A clear improvement over the existing OpenFOAM simulation can be seen, even for the lowest resolution. A clear convergence to the overtopping rate of the experiment can be seen, but all simulations slightly underestimate the experiment. For the coarser resolutions, as fewer particles are used to represent the small volume of water over the dike area, this is an expected result (also discussed in Section 3.3.1).

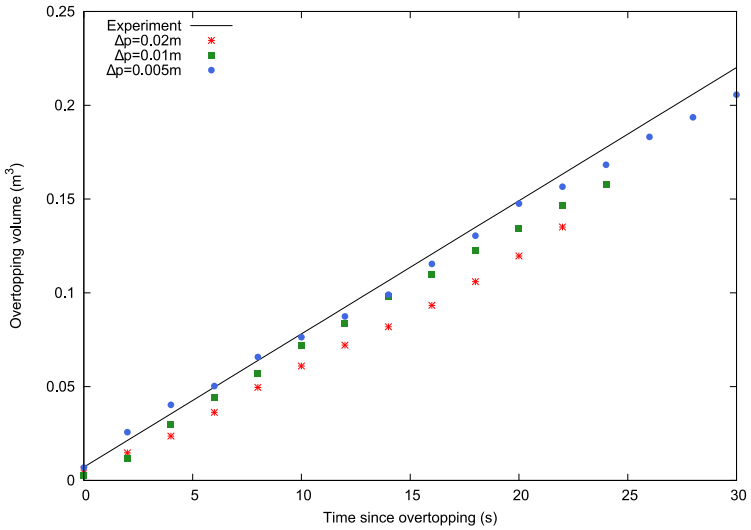


Figure 10: Overtopping volume comparison for GPUSPH Simulations A (Δp=0.02m), B (Δp=0.01m), C (Δp=0.005m) using DB and 3 different resolutions with the experimental results

Figure 11 shows the effect of the tank width on the overtopping volume. Using a larger tank (Simulation D) shows slightly less agreement with the experimental results, likely due to increased friction among the fluid and boundary particles and the absence of 3-D flow phenomena. Taking into account the higher computational cost as the number of particles is doubled, it is recommended that the smallest possible width is used.

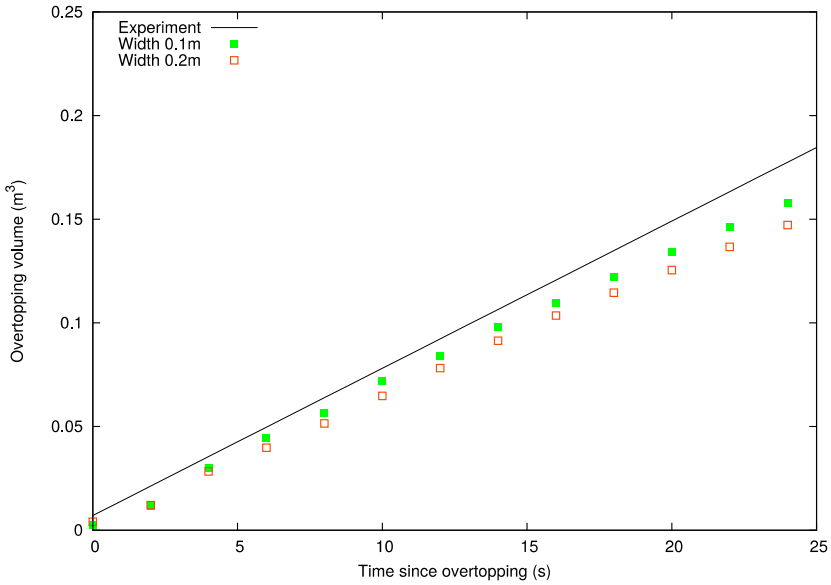


Figure 11: Overtopping water volume for GPUSPH Simulations B (width 0.1m) and D (width 0.2m)

The overtopping volume also depends on the configuration of the AWAS model. The system used here (Carmigniani, et al., 2017) depends on a height probe in front of the piston whose position, as shown in Figure 12(a) alters the simulation result. This is due to using the DB formulation which causes separation of size h at the boundary-fluid interface (Crespo, et al., 2007) (Crespo, et al., 2011). This issue is compounded near the free surface as shown in Figure 12(b) due to the incomplete kernel of the fluid particles. Therefore, if the height probe is placed too close to the paddle (a distance of $5\Delta p$ seems to be affected) the recorded height will differ significantly from the theoretical result causing the issues in Figure 12(a).

It should be noted that placing the probe further from the piston necessitates correcting the theoretical height used for controlling the velocity. Controlling the height with multiple height probes is also an option, however, the results obtained were nearly identical to the ones shown in Figure 12(a) for a $10\Delta p$ distance.

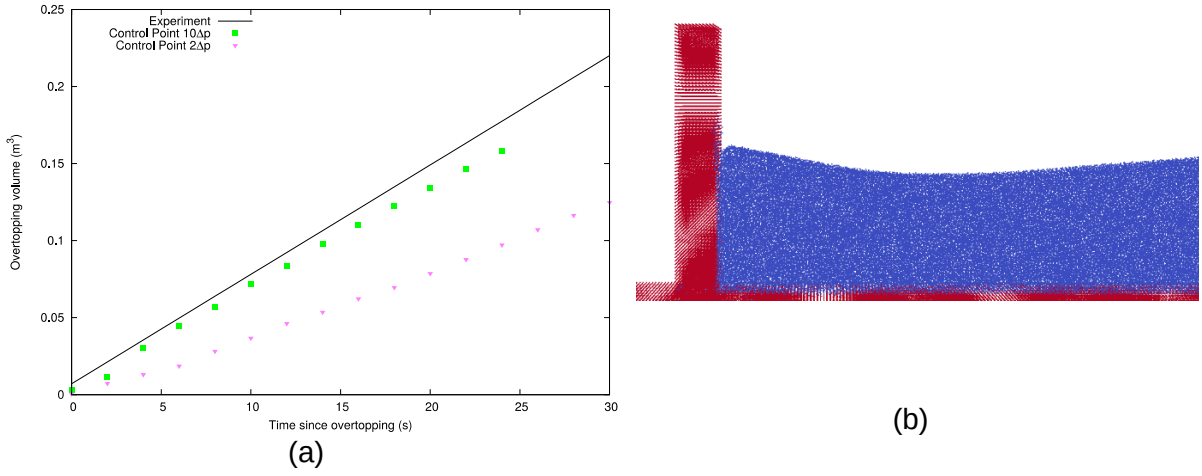


Figure 12: Overtopping water volume for different AWAS control points with Simulation B (a) and fluid-boundary separation in front of the paddle

The simulations shown so far use the DB formulation. The SA formulation has also been used, however, as shown in Figure 13, this leads to a significant reduction of overtopping volume compared to the experiment. The reason seems to be a difference in the water height after the first obstacle. Probes 8 and 11, located before the obstacle, show the two simulations having similar free surface positions, while height probes 12-16 (shown in Figure 13) show a consistently lower water level than the DB simulation, especially for the early waves.

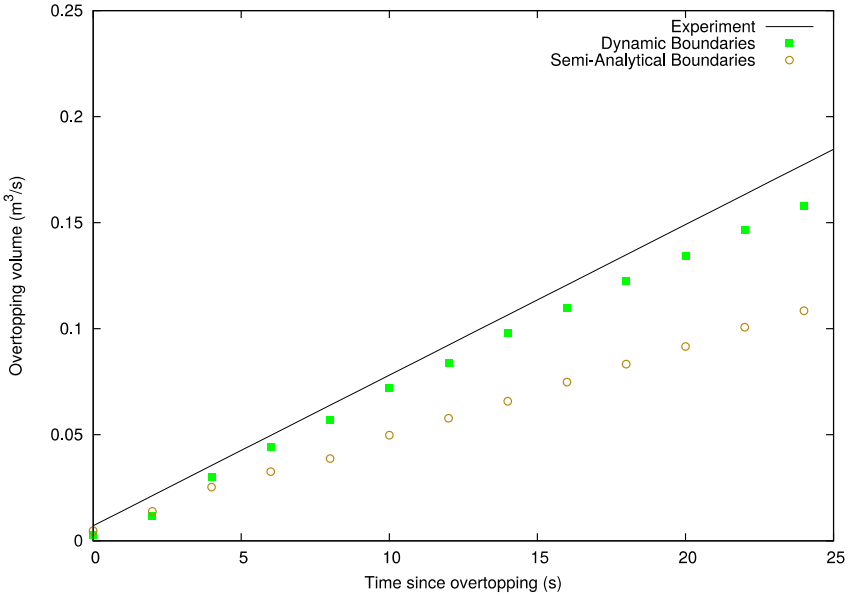


Figure 13: Overtopping water volume for different boundary formulations

An important parameter to consider for the SPH simulation is the numerical dissipation of the waves. To quantify its effect, Figure 14 shows the overtopping volume for Simulations B, E, F (3 different tanks with varying lengths). The results have been shifted with time to account for the different tank lengths. A decrease on the overtopping volume as the distance from the dike increases can be observed confirming that the SPH wave decays as the travelled distance increases. An additional factor that causes wave decay is the viscous term. Figure 15 shows that the increased dissipation by its use and its adverse effect on the overtopping volume. It is recommended that a viscous term is not used for SPH simulations of waves.

The results shown so far are focused on a wave with 0.22 m height, as measured in front of the piston. However, multiple wave heights have been used in the experiment and Figure 16 shows the comparison with the simulation results. A good agreement can be seen with most values although at 0.23 m the overtopping volume is underestimated. The experimental results are presented in two different ways due to the slope at the experimental study. Figure 16a shows the wave height created by the piston as captured by gauge 9 while Figure 16b shows the wave height after it passes the slope and was measured by gauge 1.

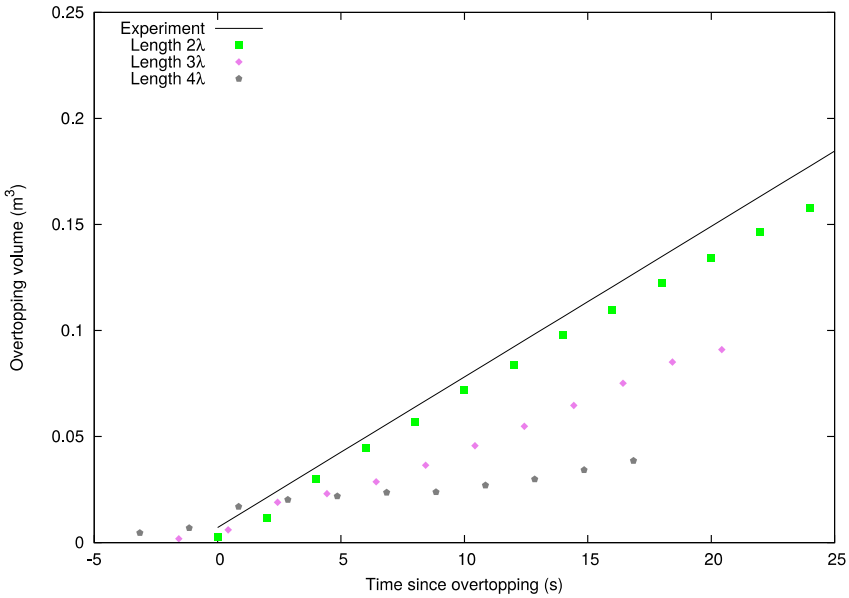


Figure 14: Overtopping water volume for GPUSPH Simulations B (2λ), E (3λ), F (4λ) using different tank lengths

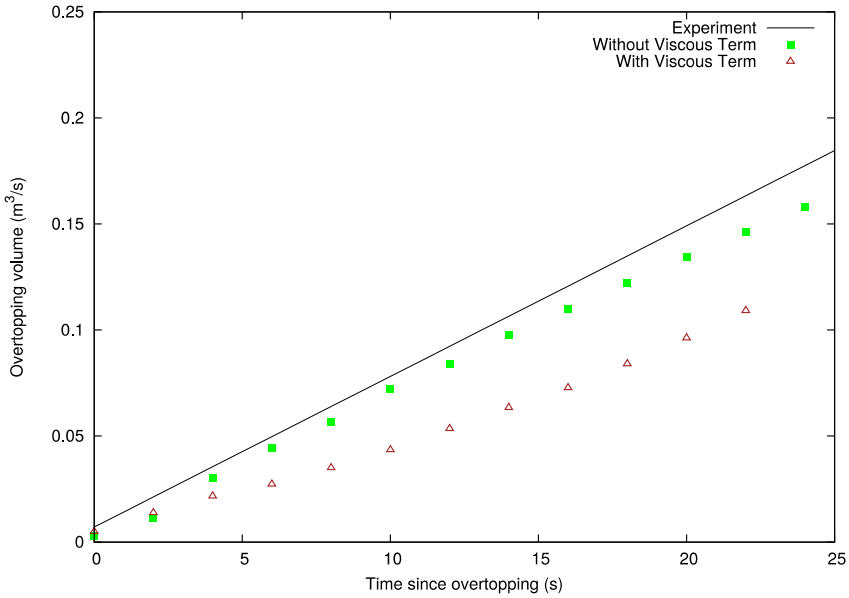
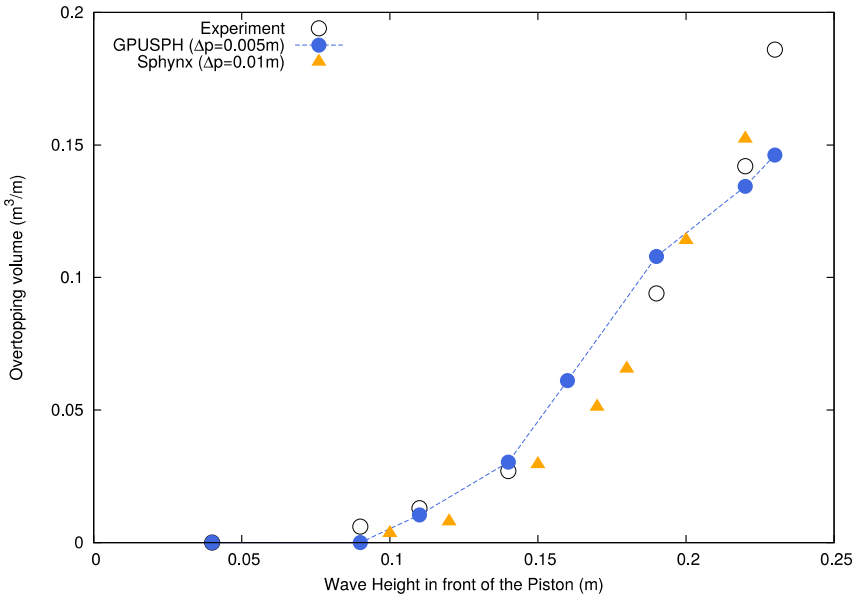


Figure 15: Overtopping water volume for GPUSPH Simulation B with and without the SPH viscous term

(a)



(b)

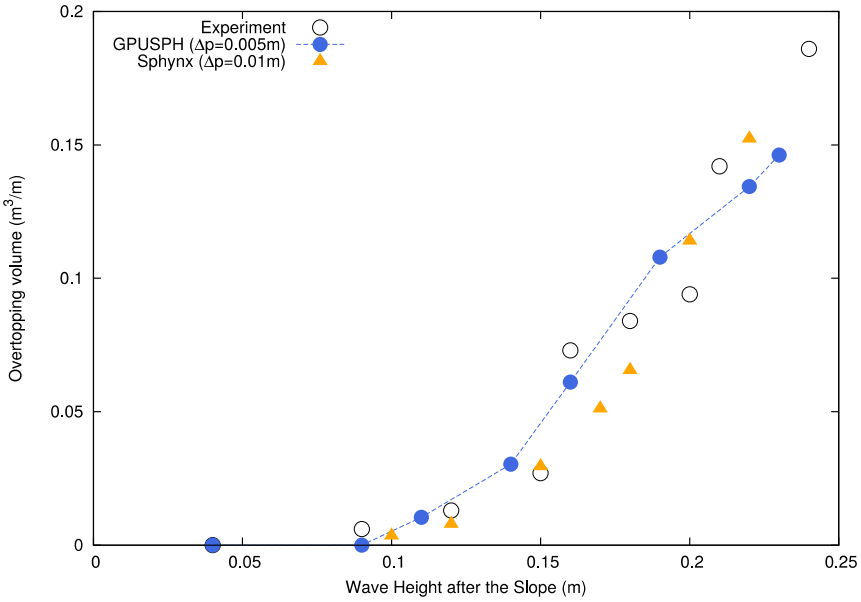


Figure 16: Overtopping water volume for different wave heights. GPUSPH results correspond to the configuration of Simulation B while experimental results are presented according to the wave height in front of the piston (a) or according to the wave height after the slope (b)

4 Wave Overtopping on Realistic Topography

4.1 Pre-processing Realistic Topographies

The simulations in this part of the report use v5.0 of the GPUSPH code (http://www.gpusph.org/) released on the 13th of June 2019, in an internal cluster of EDF R&D. Appendix 7.1 shows an investigation on the runtimes of GPUSPH in different environments.

Of particular note is the SALOME module for GPUSPH pre-processing, SALOME_PRESPH, which is introduced with v5.0 and greatly simplifies the task of introducing complex 3D geometries in GPUSPH. The pre-processor uses a triangular mesh to map automatically fluid and boundary particles using on the coordinates of the cell nodes. A complete user guide is available through SALOME, but a number of essential practices for complex domains will be discussed here. Figure 17a shows a screenshot of the module and its parameters.

In both of the geometries discussed here, the bathymetry was introduced in SALOME as a STL file. It is recommended to load the STL file using the SALOME Mesh rather than the Geometry module. The latter is not suitable to deal with complex geometries consisting of multiple parts and a significant amount of time is required each time the domain is edited or loaded. It is, however, useful for quickly creating simpler elements that are needed for the SPH computation, e.g. walls or a paddle.

(a)

(b)

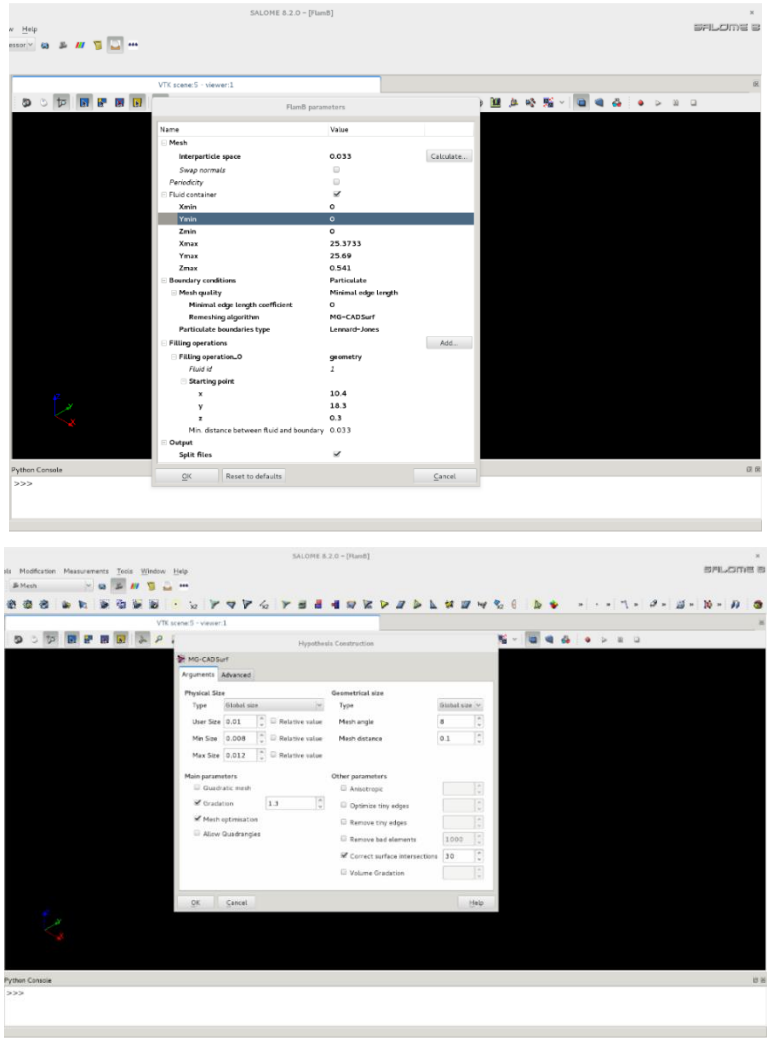


Figure 17: Screenshot of the parameters of the SPH pre-processing module (a) and the MG-CADSurf algorithm construction (b) in SALOME when used with the dike mesh including building data.

When using the Mesh module, remeshing the domain may be necessary as the eventual particle distance depends on the cell size of the mesh. While the SALOME_PRESPH module can generate the mesh automatically, the user loses control over the mesh characteristics. Therefore, preparing a suitable mesh beforehand is recommended. Either of the meshing algorithms available can be used, but empirical observation has shown the MG-CADSurf algorithm to produce the most consistent results and, as a result, it is used in the simulations throughout the report. In addition, as the pre-processor only accepts one mesh as input, the use of the Compound Mesh filter is required when the domain consists of several files.

Table 5: Main options and recommendations for using the MG-CADSurf algorithm for SPH simulations

MG-CadSurf Option	Recommendation	Comment
User Size	0.8·Δr	While the meshing algorithm will target this value, some cells will usually end with a larger size limiting the final SPH particle distance. A lower target pre-empts that.
Min Size	0.6·Δr	While the same value as the user size can be set, it is recommended to allow the algorithm some flexibility to deal with complex areas. Will not affect final particle distance.
Max Size	Δr	While it theoretically acts as a limit to the cell size, in practice it is likely some cells will be larger. This difference, however, should not be particularly large.
Gradation	1.1-1.3	Smaller values are more beneficial but may be counter-productive if the mesh used is coarse or geometric elements necessitating different cell sizes exist.
Mesh optimisation	True	Improves the shape of the mesh elements.
Mesh Angle	4-8	The default value of 8 should be sufficient in most cases but a smaller value may be beneficial for very fine meshes.
Mesh distance	10·Δr	This value was observed to produce good quality meshes while allowing for some flexibility on the algorithm.
Anisotropic	False	Avoid due to creation of different sizes for neighbouring cells
Optimise tiny edges	0.1·Δr	If small cells are observed in the final mesh, the value can be increased up to 0.5·Δr. Should not exceed Min Size.
Remove tiny edges	False	Avoid, as it may remove cells within the domain, creating holes.
Remove bad elements	False	Avoid, same as above.
Correct surface intersections	15-30	A larger value leads to better quality meshes but also increases the computational time significantly

Using certain options for the meshing algorithm and the pre-processor will assist in lessening the load of the compounding algorithm. Table 5 presents a rundown of the MG-CADSurf meshing options and the values that were used in this report while Figure 17b shows a screenshot of the parameters.

The next step is the introduction of the mesh in the pre-processing module. That includes a number of options for the SPH parameters, which are briefly examined here. It is expected that as the pre-processing module is further developed, new options and programming improvements may render these recommendations non-essential.

However, as of the writing of this report, the following options and recommendations should be taken into consideration:

- **Main Boundary:** Either of the three options (geometry, mesh, external .stl file) is viable, with the last two preferred for complex boundaries In either case, it should include the entirety of the domain including any moving boundaries. It should not include the free

surface. This report used the mesh option as it allows for greater control over the final result.

- **Free Surface:** It is recommended that a geometry object is used. If a mesh has to be used instead, its size can be much larger than the mesh size of the domain. Either is advantageous because the particle creation will be completed faster as the algorithm will check against fewer surfaces.
- **Special Boundary Grid:** Any moving boundaries should be included here as a separate mesh. It is important that the mesh size is the same as the main boundary mesh.
- **Interparticle Space:** There are two options for selecting this. If the main boundary is a geometry object or an external .stl file, the user should input the value. If it is a mesh the 'Calculate' option should be used. The latter is the preferred method in the report, due to additional options available when creating the mesh through SALOME. Because the value given by the 'Calculate' option is a float number with 8 decimal points, rounding up this number to 3 decimal points may assist the particle creation algorithm with complex geometries.
- **Swap Normals:** Not used in this report as all domains used Lennard-Jones boundaries.
- Periodicity: Not used in this report as the bathymetry is too complex to create an identical mesh in each side.
- **Fluid Container:** It is recommended to use this option instead of 'Box' in 'Filling Operations' by inputting the maximum and minimum values of the boundaries. It is not necessary to specify a free surface when using this option.
- **Boundary Conditions:** The three options included (SA, DB and LJ) follow an inverse curve between accuracy and computational expense, with the most accurate method (SA boundaries) being considerably more expensive. Due to the large domains and number of particles, the simulations in this section will use the LJ boundaries, which minimise the computational time but will cause increased wave dissipation.
- **Mesh Quality:** The minimal edge length option has been used for this report. The coefficient remained at its default value, 0.
- **Remeshing Algorithm:** Depends on the state of the main boundary: if it is a good quality mesh, the 'None' option should be used (as does this report). If the main boundary is a geometry object or an external .stl file, the MG-CadSurf algorithm is considered more effective at the time of writing.
- Filling Operations – Geometry: Should be used in tandem with the 'Fluid container' option. A seed point at the middle of the container, away from the boundaries, is preferred.
- Filling Operations – Box: Only recommended for simple geometries.
- Wall-Particle Distance: Not used in this report. The distance is considered equal to the interparticle space and resolved automatically.
- Output: Split files is always used with moving boundaries.

(a)

(b)

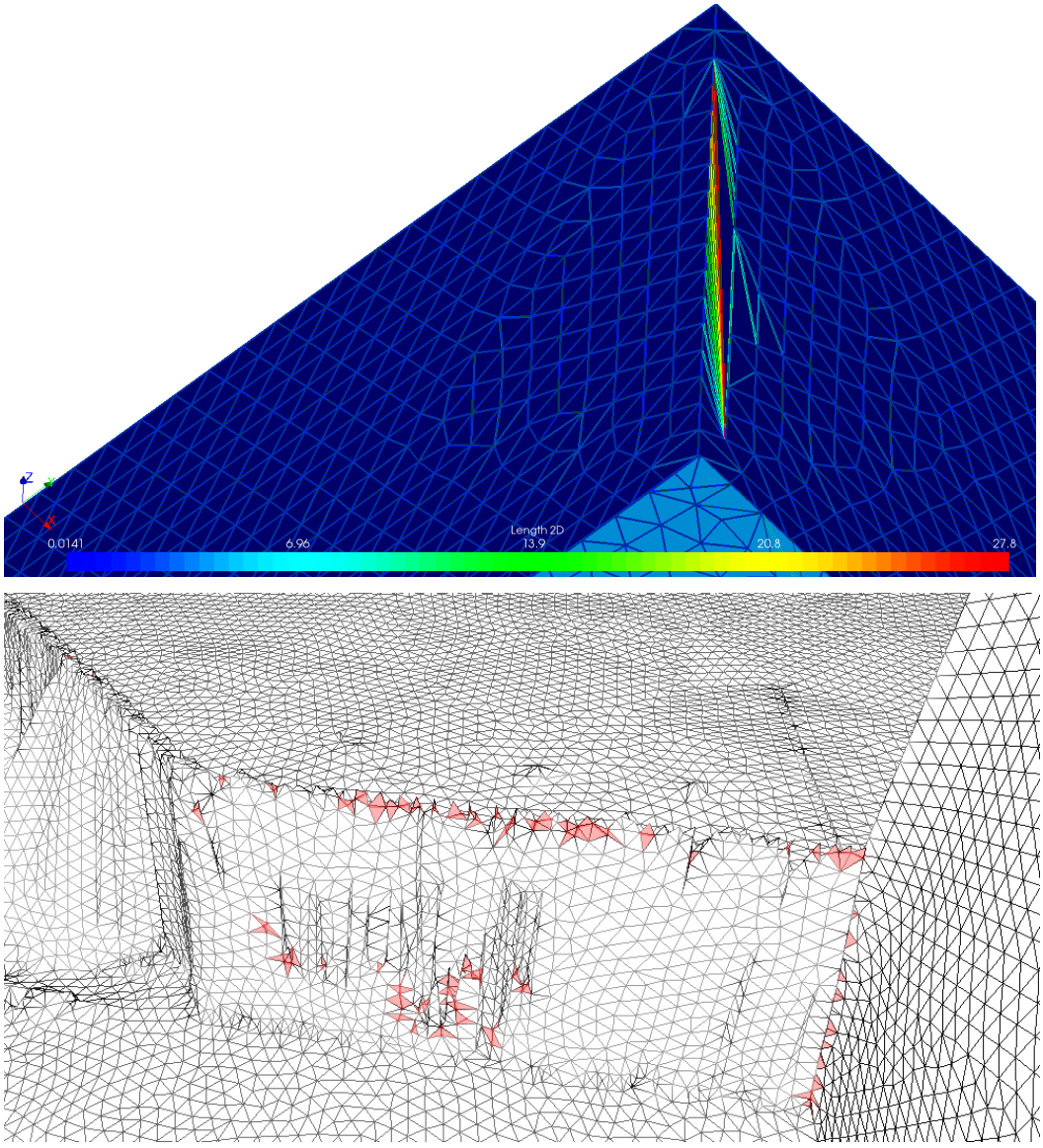


Figure 18: Instances of mesh issues: (a) shows different cell and area sizes at an intersecting edge while (b) shows non-manifold edges as identified by Meshlab.

It is essential that a good quality mesh is provided for the SPH pre-processing module, not only because the final particle arrangement is improved, but also due to the pre-processor failing or even crashing SALOME if a low quality mesh is used as input. Figure 18 is showing instances of a poor mesh occurring. The following four are the main issues that the user should be aware of when using a mesh:

a) **Different cell sizes:** As the particle distance that should ideally be used corresponds to the largest cell size, a skewed cell will increase the recommended value for the entire domain.

b) **Large cell area:** Cells with a large area, even when not skewed, may create an opening through which fluid particles can slip past, especially if neighbouring cells are similar.

However, even if no penetration occurs, a lower repulsive force will be imposed on the fluid affecting the simulation.

c) **Non-manifold mesh:** May either prevent the pre-processing from filling the domain with fluid particles or crash the program, as the module is unable to correctly identify the distance from the boundaries.

d) **Holes in the mesh:** In order to populate the domain with fluid particles, an unbroken boundary is recommended, or the pre-processor is likely to fail. However, if the hole is sufficiently small (less than half the typical cell size) the calculation may continue.

These issues occur often when the meshing algorithm cannot cleanly divide the intersecting edges between multiple geometries and stl files. As the cell sizes in each one may be slightly different, the nodes of the two meshes in the intersection are not aligned after using the Compound Mesh filter, creating issues when they are remeshed. Of course, this heavily depends on the complexity of the intersections. While, for example, a straight-line intersection can be divided by cells of equal size, so the nodes of each mesh have the same coordinates, a complex coastline with differing heights is highly likely to have differently sized cells on each side.

A solution to this is changing the merging tolerance of the Compound Mesh filter, allowing the program to merge vertices that are further apart. The maximum recommended value would be half of the minimum cell size, so that only the vertices in the intersection (which should have a smaller distance) are affected. Otherwise, internal cells and vertices will be merged, altering the geometry of the domain. A smaller tolerance value would be, of course, preferable. Empirical observations have also shown that this step is easier using finer meshes.

Depending on the complexity of the domain, the above solution may not be applicable as the minimum cell size could differ between different files. In that occasion, minimising the number of intersecting edges the algorithm has to deal with is recommended. In SALOME, this can be done by using the Compound Mesh filter multiple times in succession affecting only two or three meshes each time.

As an example, the coastline meshes used in later sections typically consist of the main bathymetry and four additional geometry elements; walls to restrain the fluid particles within the domain. If the meshes are compounded at the same time, the algorithm has to deal with 8 intersecting edges simultaneously (4 between the bathymetry and the walls and 4 between the walls themselves). On the contrary, if each wall is compounded separately, the algorithm has to deal with 3 edges at most (when incorporating the last wall segment, it will be connected to the bathymetry and two other wall segments) making it easier to calibrate.

If the above options do not solve the issue, use of the open-source mesh editing software Meshlab (http://www.meshlab.net/) may. The Cleaning and Repairing section of the Filters option includes several options that are especially useful in avoiding these issues, the following in particular:

- **Remove Duplicate Faces/Vertex:** Duplicates, especially at intersecting edges can lead to different cell sizes used for each side (issue a).
- **Remove Faces from Non-Manifold Edges:** Deals with issue c. however, as holes may be created from the face removal, a remeshing is often necessary.
- **Remove Zero Area Faces:** Necessary to identify unreferenced vertices.
- **Remove Unreferenced Vertex:** In case of remeshing, unreferenced vertices lead to the creation of unwanted faces. Erroneous particles are also be created by the pre-processor.

An additional option for alleviating issue d is the Close Holes filter in the Remeshing, Simplification and Reconstruction category. As the name implies, it creates new faces between existing vertices. It is especially useful when modifying the mesh as well. It is recommended that it is only used with a selected area, rather than a global application as, depending on the

max size selected, it may erroneously connect other vertices, especially in the edges of the domain. An example of closing a gap between two intersecting meshes is shown in Figure 19.

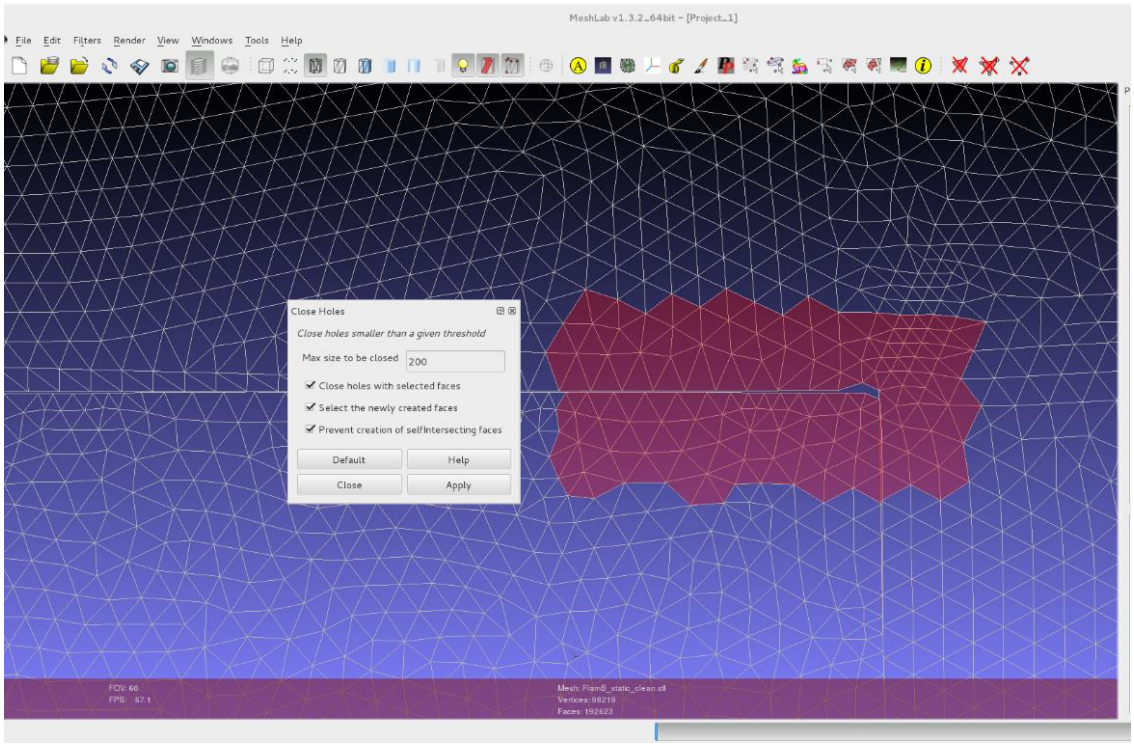


Figure 19: An example of using the Close holes filter in Meshlab.

Meshlab also includes a suite of options and filters for remeshing and editing, but since the files will be used by SALOME, the latter has been selected as the primary meshing tool for this report.

4.2 Simulation of a Realistic Coastline

4.2.1. Source File Treatment

In order to assess the flooding risk, a realistic topography including the bathymetry of the surrounding area is used. The data used here originate from a low-resolution, altered version of a real coastline. A low height area in the south-east corner to collect any overtopping water is also included. The data were provided in two different files to investigate the pre-processing procedure. The first is an ascii file containing data regarding the height of the area as well as the bathymetry of the surrounding coast. The results were presented using the Lambert 93 coordinate system (EPSG:2154). They did not include water height data. A screenshot of the data can be seen in Figure 20.

(a)

(b)

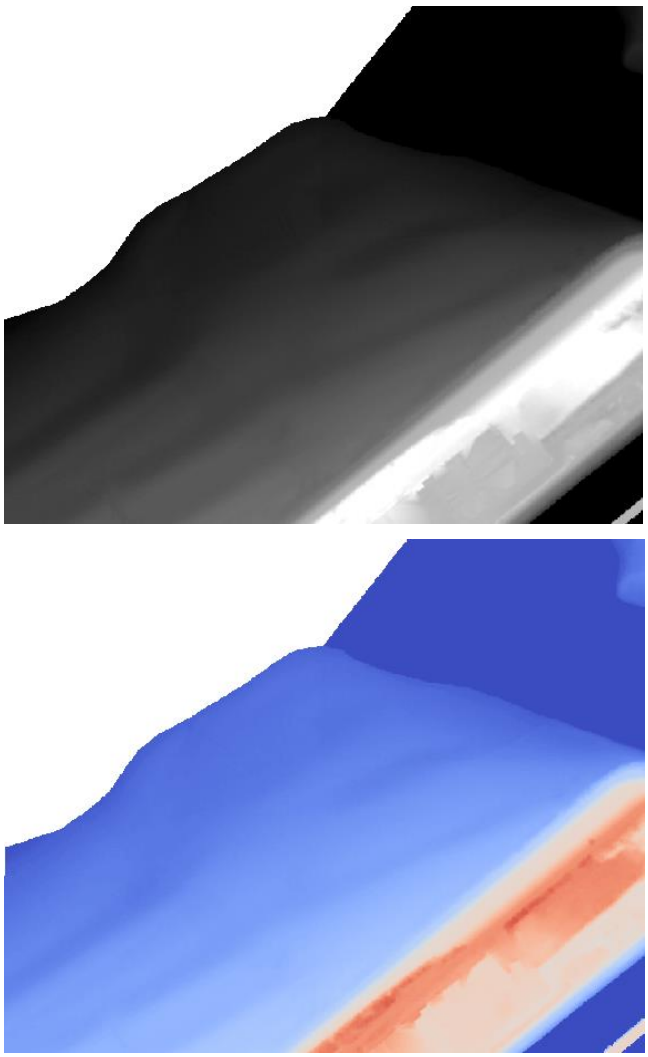


Figure 20: (a) is the original ascii file, exported from QGIS and coloured in grayscale with the lighter areas signifying an increased height. (b) is the stl file used for SALOME input, after being treated by ParaView.

As an .ascii file, it is not immediately usable by SALOME. It can, however, be opened and treated by the QGIS open-source geographic information system (https://www.qgis.org/fr/site/), imported with the Add Raster Layer option. To export it to workable format, the plugin GdalTools is needed (depending on the version it may need to be enabled). Using then the Processing Toolbox -> GDAL/OGR -> [GDAL] Extraction path, the user can find the gdal2xyz tool which can export the asci file to a .csv format.

The second file batch (Source B) was a .csv file, also provides additional bathymetry data for the same area. Comparing Figure 20 and Figure 21, we can see that the .ascii file has a more dense point cloud, but the .csv file covers a much larger area. Therefore, a merge of the two will be necessary in order to portray the domain in its entirety.

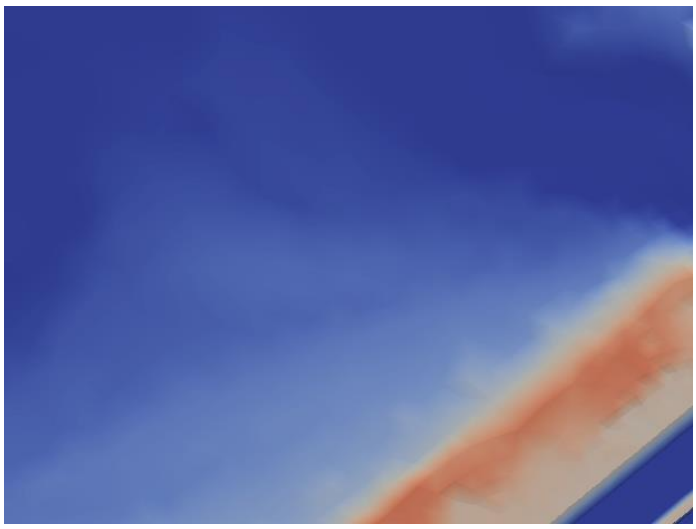


Figure 21: The .csv file as seen in ParaView after using the TableToPoints filter.

As a .csv file, it cannot be directly imported in SALOME and the pre-processor module. To create a usable file, they are imported in ParaView (https://www.paraview.org/), an open-source data analysis and visualisation application. The TableToPoints filter is then used to transform the point cloud to 3D coordinates. Transforming them into coordinates allows the use of the Delaunay 2D triangulation filter to create a mesh.

The use of the 2D instead of the 3D Delaunay filter is preferred because, even though there is variation in height, it is 2-3 orders of magnitude smaller than the variation across the width and the length of the domain, so it can be approximately considered 2D. The default options were used in the filter as its purpose is just to save the data in a more useful format. In addition to the original point cloud lacking in detail, the Delaunay filter in ParaView lacks sufficient options to create a good quality mesh, oftentimes creating multiple erroneous faces that need to be removed.

The removal is done in open-source mesh editing tool Meshlab (http://www.meshlab.net/) using the Select and Delete Faces tool. It is important that the vertexes themselves are not removed, only the faces erroneously connecting them. The mesh can then be inserted in SALOME for further editing and remeshing.

The .csv file can also be imported in QGIS using the Add Delimited Text Layer option, but QGIS does not include any triangulation filters, so it is necessary to use ParaView to create an initial mesh.

The pre-processing steps listed here do not require significant processing time, with the exception of the meshing and the pre-processor module, which may require 1-2 days each, depending on the complexity of the geometry and the fineness of the resolution.

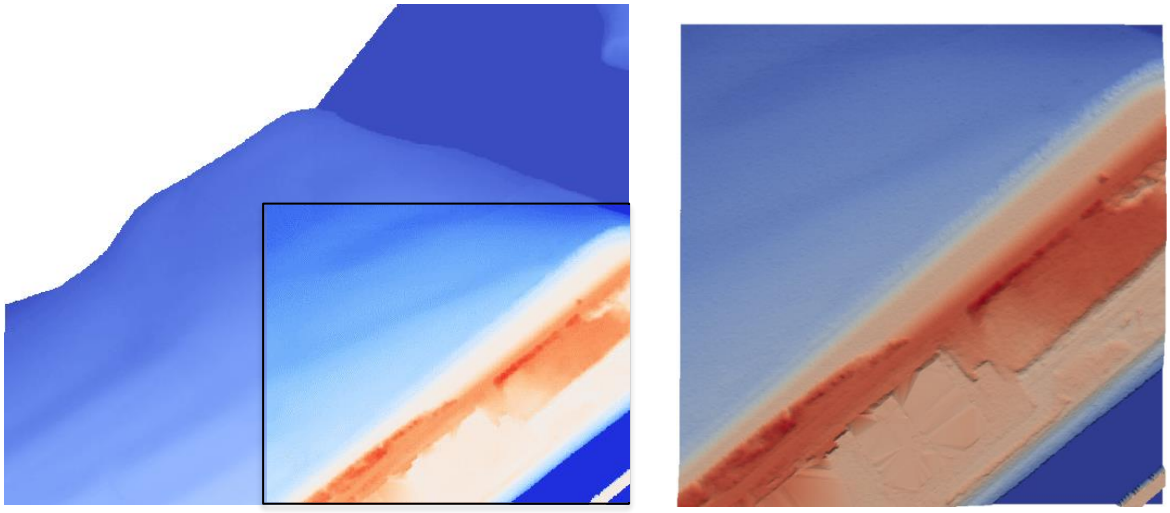
4.2.2. Domain Selection and Simulation Results

As mentioned, the different files contain different data of differing quality. To create a mesh for the GPUSPH simulation, a unified mesh will be created, combining data from both sources. The .csv mesh will be used for the far-off bathymetry and the .ascii file will cover the coastline. The mesh files will be combined using Meshlab and specifically the Automatic Pair Alignment (to resolve any overlapping issues) and the Flatten Visible Layers (to merge the meshes) filters. The resulting unified mesh can then be remeshed with a smaller cell size in SALOME in order to create a more isotropic arrangement.

The primary issue about the SPH simulation is the size of the domain. The domain is about 1000 m by 2000 m, making the simulation of the waves, where important phenomena happen

on the centi- or millimetre scale computationally unfeasible. Therefore, the simulation domain needs to be reduced. As the aim is to investigate overtopping the focus should be on the land strip.

The focus here is on waves overtopping over normally dry areas, as well as the feasibility of an implementation of complex topologies and bathymetries for SPH. The SPH models that are being used have been selected to achieve these objectives and are not optimal for the modelling of a complex channel flow with multiple entry points. This would require a different SPH approach focusing on mitigating dissipation and implementing inflow/outflow boundaries. This is possible with GPUSPH by using semi-analytical boundaries but is beyond the scope of this work which will focus on the land strip.



(a) (b)

Figure 22: The highlighted area is the one selected for the SPH simulation (a), shown in more detail in (b)

The land strip has a relatively constant width, measuring about 170 m. It also, as can be seen in Figure 20, contains its highest points on its north side. Therefore, it is reasonable to define as a catastrophic overtopping a significant volume of water clearing the north side.

To simulate the waves impacting on the land strip, a portion of the seabed needs to be modelled. As discussed in section 3, the SPH method used here dissipates waves traveling beyond two wavelengths. At the same time, the bathymetry in front of the strip consists of a long slope, which is important on the eventual wave characteristics (Dally, et al., 1985). Figure 22 shows the final domain selected for the SPH simulation; an area with dimensions 350 m by 350 m. It also contains a low height area beyond the strip, which will be used for collecting any overtopping water.

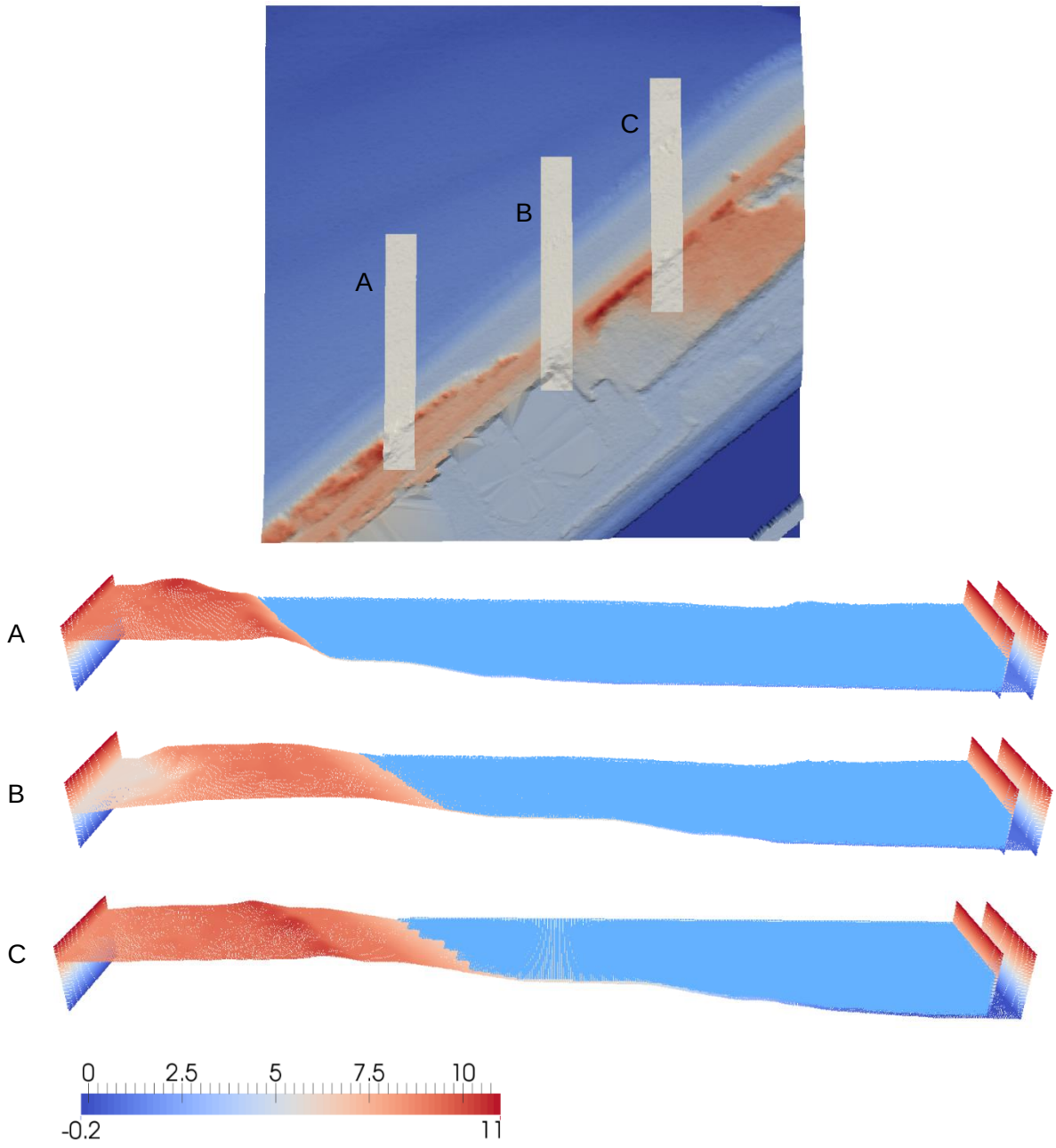


Figure 23: The domains A, B, C selected for a first simulation of the realistic coastline.

Before working on the domain of Figure 22, it was decided for GPUSPH to be tested in a smaller domain. Three parts of the domain with dimensions 150 m by 20 m were selected as shown in Figure 23. The selection was largely arbitrary, as they are meant to be used for a test bed. However, they were selected to represent three different areas of the land strip: the cliff near the sea with a very short slope on A, the gentler slope and lower overall height in B and the gentle slope but high maximum height in C. Each one also includes differing water volumes.

The simulation was run with a series of regular waves. Initially, a period of 8 seconds and a wave height of 2 m was investigated. The free surface is considered to be at a higher level than usual, at 7.44 m NGF. The wave direction is southward, across the length of the domains. The wave will be created using a piston placed at the edge of the domain. Because of the complexity of the bathymetry, it is not possible to use periodic boundaries as with channel in

section 3, which require the mesh to be the same on each side. Instead, walls will be created to contain the particles. It is likely this will increase energy dissipation.

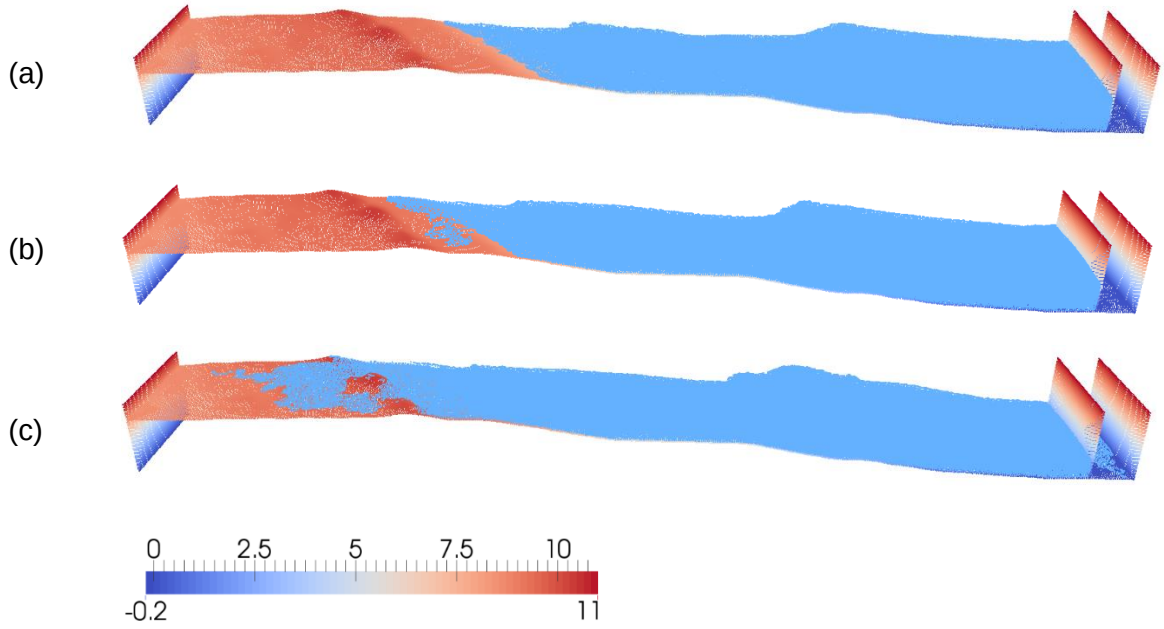


Figure 24: Results at 95s for domain C for different wave heights (a) 2m (b) 4m (c) 8m

Figure 25: Results at 95s for domains A and B for wave height of 8m

The domains were executed for 100s with $\Delta r = 0.29$m and $4 \cdot 10^5$ particles. Neither of the three showed the water overtopping the highest boundary point. Therefore, the height of the wave was increased sequentially to 4 m and 8 m sequentially. Figure 24 shows the results for domain C for the three different wave heights at 95s. Both cases show an increased runup compared to the 2 m case and the 8m wave is able to clear the highest point. However, the water volume is not large and should not be able to clear the entire land strip.

Following the results of domain C, the other two domains were also simulated using a 8m wave height. Similar results to C were obtained, with water clearing the highest point in both cases. The water volume that overtopped, however, is too small to even clear the remainder of the domain and fall in the collecting area.

The next step is the simulation of the entire domain in Figure 22. The simulation is executed for 120s with a resolution of 0.05m, so about $4 \cdot 10^6$ fluid particles. The simulation time was about 20 hours. The case, following the results at Figure 24 and Figure 25 was executed for a wave height of 8m as shown in Figure 26. The larger case did not demonstrate the overtopping

shown previously, most likely due to the decreased resolution. The increased distance from the beach also affected the results, with run-up increased in the area closest to the paddle. The larger domain was also simulated with a wave height of 12m. Run-up was increased across the beach, most notably in the area close to the paddle, where some overtopping was observed. As with the results of Figure 25, the water did not reach the lower area at the south-east corner.

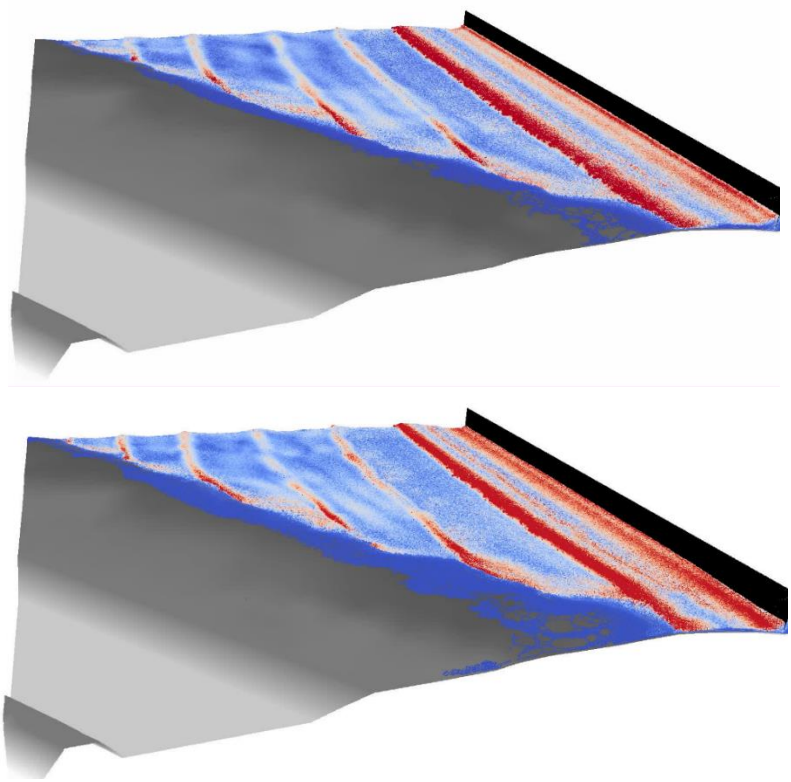


Figure 26: Results at 90s for the coastline simulation domain for wave height of 8m (a) and 12m (b)

In conclusion, the coastline simulation has shown some encouraging results, regarding the use of SPH to measure overtopping. However, some of the disadvantages of SPH make this a difficult case, due to the disparity between the size of the domain and the scale of the phenomena occurring. It was decided that simulation of a smaller experimental dike model would be more beneficial for this project.

4.3 Simulation of a Model Dike

4.3.1. Creation of the Simulation Domain

An experiment, containing a model of a complex dike, took place within EDF. This report will focus on using SPH to simulate the model and compare with the experiment. The main focus is the feasibility of using SPH to model a real geometry and the accuracy and speed of the pre-processing using the new tool on SALOME.

The aim of the model (constructed in a 1/72 scale and shown in Figure 27) is to measure the overtopping volume as well as the erosion of the dike. Four troughs (BAC 1-4, numbered from west to east) were created behind the dike, each one collecting the overtopped water. The water in the troughs was not allowed to spill and was periodically collected. Collected data include the volume of overtopping, the distribution between the four troughs and free surface levels at gauges 6-12 shown in Figure 27.

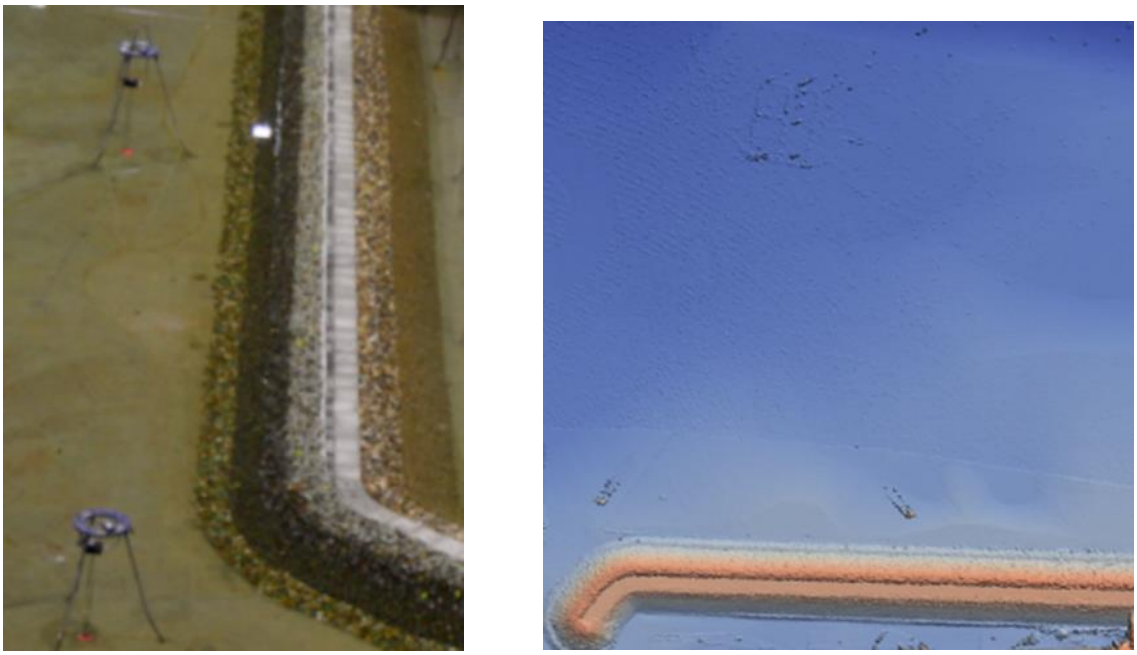


Figure 27: The point of interest in the dike scale model.

The experiment considered separate cases regarding storm surges and wave heights. This report will focus on a case investigating conditions beyond the baseline similar to the ones that caused the Fukushima incident on 2011. The free surface position is assumed to be 9.29 m NGF, conforming to the framework of a study carried out after the Fukushima incident. Different storm surge conditions can be studied by changing the starting free surface conditions.

The wave being investigated has a height of 9.3 m and a period of 13s. It corresponds to the expected hundred-year wave (which is usually taken into account by offshore structure designers) with a 70% statistical probability. As the model is in a 1/72 scale, the wave height in the simulation is 0.129 m. The period is subject to Froude number scaling with a 1/8.5 scale being used. Therefore, the wave period in the simulation is 1.5294s.

The full experimental structure, shown in Figure 27, included a number of absorbers and reflective plates. A snake paddle was used to induce water movement in the experiment. A laser scan file of the model was provided, containing both the environment bathymetry and the testing apparatus. The laser scan was provided as a .vtk file (scan-cuve-3d – Cloud.vtk) and was treated by the open source point cloud processing software CloudCompare (www.cloudcompare.org) to export a .ply file. The .ply file can then be imported to SALOME.

For the GPUSPH simulation, only the environment bathymetry was used. Figure 27 shows the initial mesh after the testing apparatuses have been removed. While the bathymetry mesh remains unchanged, the surrounding structure is simplified for the GPUSPH simulation, replacing the reflectors and absorbers with simple walls. While the wave absorbers could be modelled, a large area is required to do so accurately and would lead to an increase in computational cost. The absorber behind the paddle will be replaced with a wall closer to the dike structure, reducing the size of the domain.

The dike in the model was allowed to erode and change its shape throughout the computation. As erosion and non-Newtonian models were not readily available for GPUSPH at the time of this report, the erosion of the dike will not be modelled, with the shape of the dike remaining constant throughout the computation.

The snake paddle will be approximated by a piston, situated at the north of Figure 27b, which will create waves travelling directly along the walls. That means the waves will have a slightly different angle from the experiment, where the wave direction is selected to be perpendicular

to the cross-section of the intake channel, in order to maximise the chances of overtopping. The approximation for the simulation should be, however, appropriate for a first investigation.

The area around the piston is, as can be seen in Figure 27, completely flat. It has been placed there to allow for the paddle movement. The presence of this area allows for the implementation of an Active Wave Absorption System (AWAS) on the piston. This was not possible on the coastline test case due to the constantly changing water depth. The AWAS used is based on (Park, et al., 1999; Perić & Abdel-Maksoud, 2016; Carmigniani & Violeau, 2018). This system corrects the paddle velocity by measuring the free surface position in front of the piston and comparing it to the target incident wave. This method prevents the decrease of the water surface as the paddle moves forward to maintain the initial surface elevation (Carmigniani & Violeau, 2018).

The AWAS correction computes the wavelength of the wave λ through the dispersion relation for intermediate waves:

$$\omega^2 = gk \tan kH \tag{7}$$

where ω is the angular frequency and k the wave number. Considering only forward traveling waves, the equation can be solved for a known angular frequency by using a Newton-Raphson algorithm, initialized using the approximate solution of (Chamberlain & Porter, 1999) with $\Omega=H\omega^2/g$, shown in Equation (8):

$$\begin{aligned} k &= \frac{\Omega}{H}\left(1-\frac{4\left(15-P_A e^{-2\Omega}\right)}{15P_B}\right)^{-1/4} \text{ where} \\ P_A &= 15+30\Omega+30\Omega^2+5\Omega^3-10\Omega^4+2\Omega^5 \\ \text{and } P_B &= 2\Omega+\sinh 2\Omega-4\left(1-(\Omega+1)e^{-2\Omega}\right) \end{aligned} \tag{8}$$

Through the wavenumber k a corrected paddle velocity u_c can be computed. Shown in Equation (8), it uses the wave heights of the incident wave (η) and of the reflected wave (η). The former is measured every time step, while the latter is calculated (Equation) as the difference between the current measurement and the theoretical value expected at this point. The measurement of the incident wave height is taken at a 0.1λ distance from the piston, identified in report as the optimal position:

$$u_c = \left(\eta_{inc}-\eta_g\right)\omega\frac{2kH+\sinh 2kH}{4\sinh^2 kH}, \tag{9}$$

$$\eta_R = \eta_{inc}-\eta_{curr} \text{ with } \eta = 0.5H\sin\omega t, \tag{10}$$

After the implementation of the AWAS system, the full mesh can be constructed. Figure 28 shows the main structure of the mesh used for the GPUSPH simulation. Four walls (not shown in Figure 28), the piston and its additional area have been added to the original bathymetry mesh in Figure. All the additional objects have been created in the geometry module of SALOME.

The walls have been created so that they follow the edge of the bathymetry. In order to prevent a gap between the two, several of the outside points of the bathymetry mesh were recorded and inserted as vertices in the SALOME geometry module. The vertices were then linked with lines and fused to create a wire. The wire was then extruded along the Z-axis by 1.2m to create the wall. For this report, each wall will be referred according to its closest cardinal direction (North, East, West and South) as shown in Figure 28.

(a) (b)

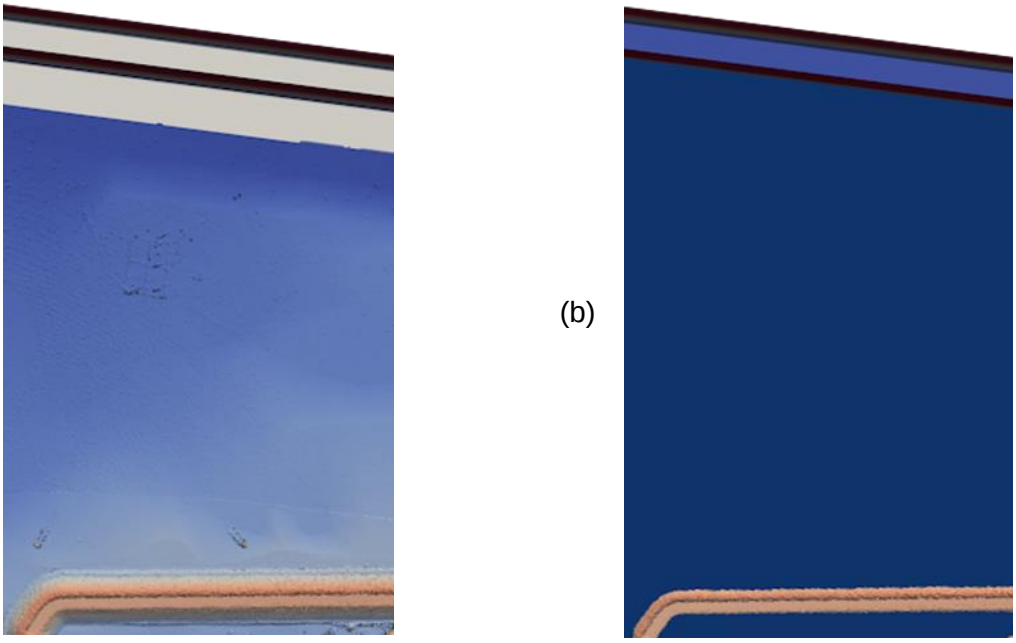


Figure 28: The SPH simulation domain without (a) and with fluid particles (b). The piston area has been marked with grey on (a). No fluid particles are placed behind the piston.

The same procedure was followed for creating the piston area. However, because this area should remain on a constant height it was not possible to exactly follow the edge of the bathymetry. To avoid a gap forming, the piston area mesh was increased by 0.5Δr so that it would overlap with the bathymetry mesh. This leads in a denser, more anisotropic area in the final mesh, but the fluid particles do not exit the system. For the piston, following the conclusions of section 3, it was constructed so that the distance between it and the surrounding meshes would be 0.5Δr. The piston area was extended behind the piston to allow for backwards movement. No fluid particles were placed in that extension.

The construction of the additional areas for the case required an additional 2-3 days as there was not an automated way to identify the geometry changes in the outer edges of the mesh, so parts of the wall had to be manually adjusted. For a simpler geometry, this work could be done in a single day.

4.3.2. Simulation Results and Resolution Analysis

In order to investigate the feasibility of running the GPUSPH case, identify the necessary runtimes and test the pre-processing module, it was deemed necessary to create a simulation with very fine resolution. To that end, a simulation with Δr=0.009m was created (which corresponds to 14 particles over the wave height), containing $56 \cdot 10^6$ fluid particles. Figure 29 shows three snapshots of the simulation at different times. The simulation was executed for 60s and its runtime was about 14 days for 8 Tesla V100 GPU cards.

The results in Figure 29 show that for this wave height, the dike is overtopped which is consistent with the experimental data. The overtopping occurs across the entire length of the dike. As the simulation progresses, the water overflows and covers the area behind the dike. The overflow did not happen in the experiment, as the water entering this area was removed.

The overflow increases as the simulation progresses and, in the west side, significant flooding can be seen. However, the presence of the wall, rather than an absorber and the lack of the experimental structure for draining will have an effect on the simulation. The amount of water deposited on the dike also increases as the simulation progresses.

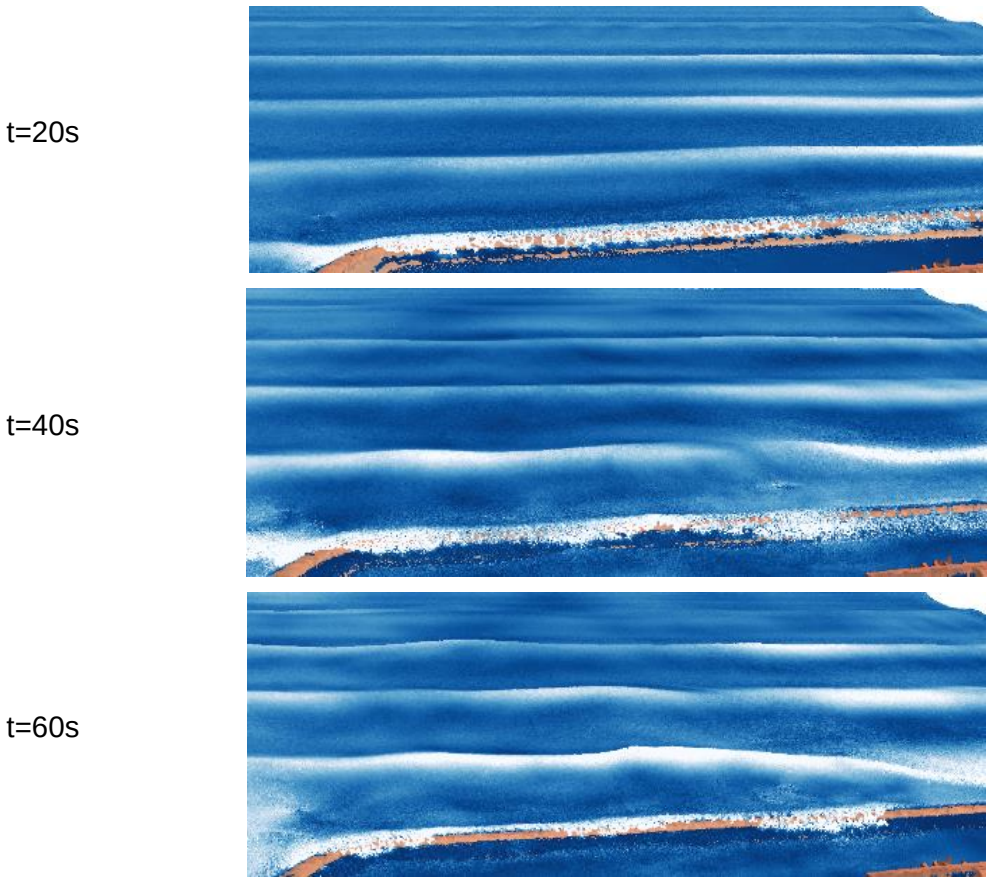


Figure 29: Snapshots of the simulation with Δr=0.009m. View at an angle behind the dike.

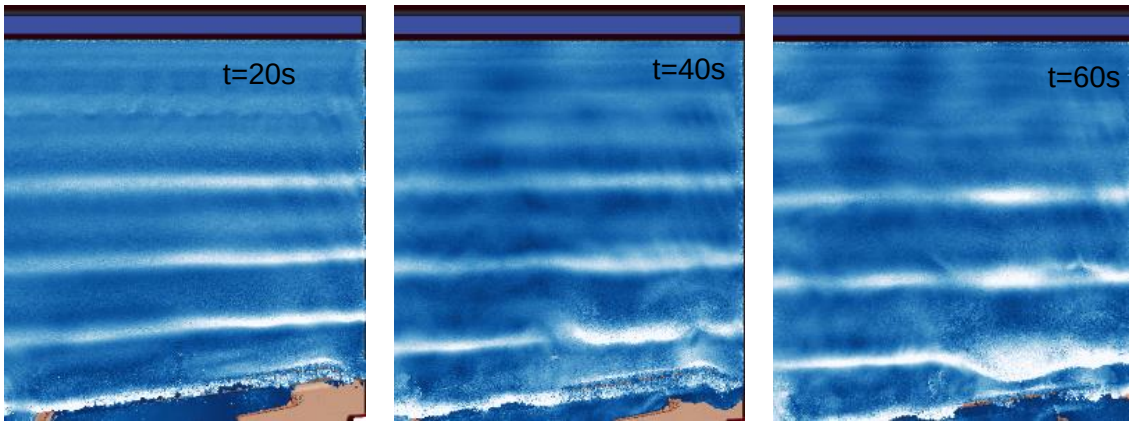
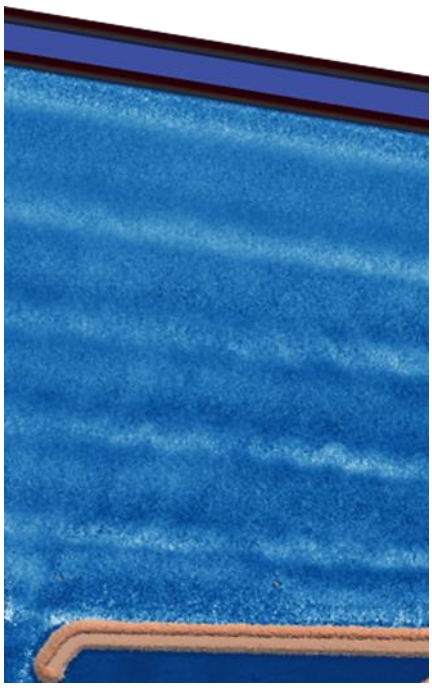
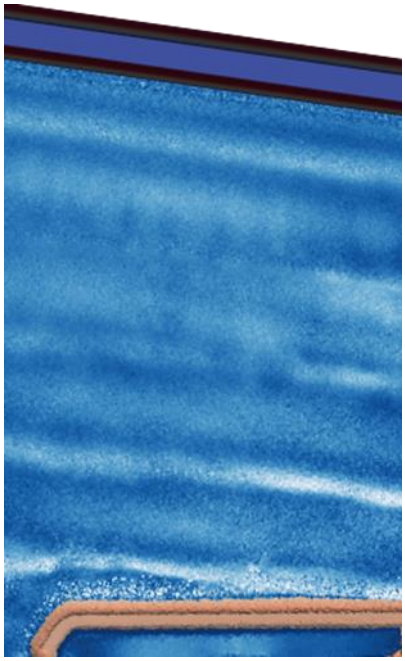


Figure 30: Snapshots of the simulation with Δr=0.009m. Top-down view.

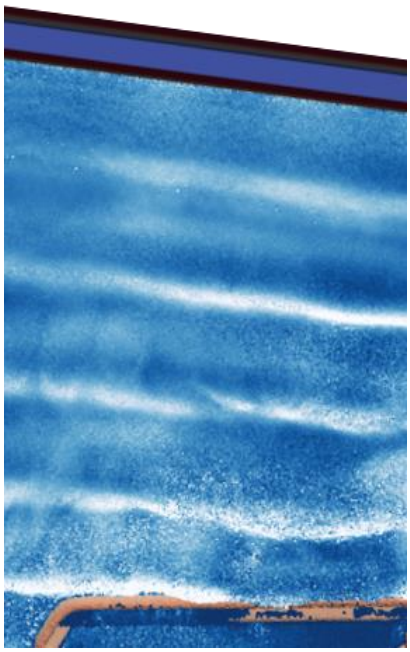
(a)



(b)



(c)



(d)

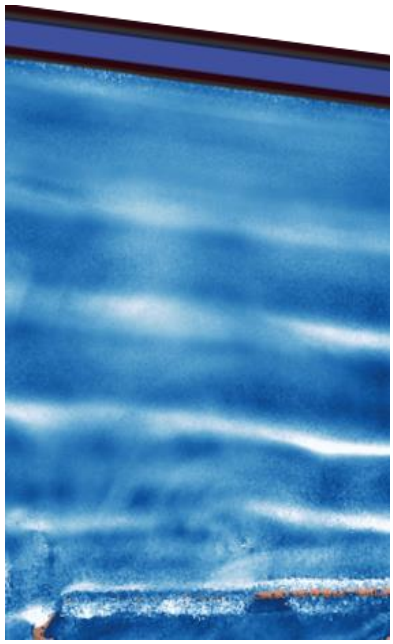


Figure 31: Snapshots of simulations at 45s with different resolutions: (a) 0.035m (b) 0.028m, (c) 0.018m and (d) 0.009m

Multiple simulations with different resolutions were executed using this domain, in order to identify any differences and identify the coarsest resolution that can reasonably reproduce the effects seen in the experiment. Figure 31 shows snapshots of the different resolutions, including the large-scale case mentioned previously. They make clear that the results are partially dependent on the resolution, as the overtopping does not happen before 45s for the lower resolutions. For the finer ones, the amount of water overtopping seems to depend on the resolution, with the finest one having higher volumes.

For lower resolution cases, there are fewer particles in the area near the dike, meaning that any particle trying to cross over will likely have an incomplete kernel and the repulsive force will have a greater effect. Figure 31 also shows discrepancies between the wave velocities,

with the velocity near the dike being greater for the finer resolutions, an effect of the numerical dissipation, which is more prevalent for low particle numbers. The use of the Lennard-Jones boundary conditions, which apply a repulsive force to the particles, will also have an effect.

The cases so far use a piston placed in the same area as the snake paddle. However, using the snake paddle, it was possible to change the angle of the resulting wave. This is not possible with the current configuration. Therefore, to create a wave with the same direction as the experimental, the piston area has been redesigned, so that the piston has a 15° angle compared to the previous configuration.

As can be seen in Figure 32, the size of the piston area has been significantly increased, meaning more fluid particles will be needed for the same resolution. The distance the wave travels is also different; while at the east side the distance has not changed; the centre and the west side of the piston are now further away from the dike. Given the increased travel distance, it is likely that it will slightly increase wave dissipation.

Table 6: Runtime data for the different resolutions on the SIEC using 8 cards

Resolution (m)	Total Number of Particles ($\cdot 10^6$)	Number of Particles per Wave Height	Simulation time per second of physical time (s)	Simulation time for a wave period (s)
0.035	2.81	3	460	703.524
0.028	5.58	4	1150	1758.81
0.018	19.59	7	4700	7188.18
0.009	56.37	14	25969	39716.99

(a) (b)

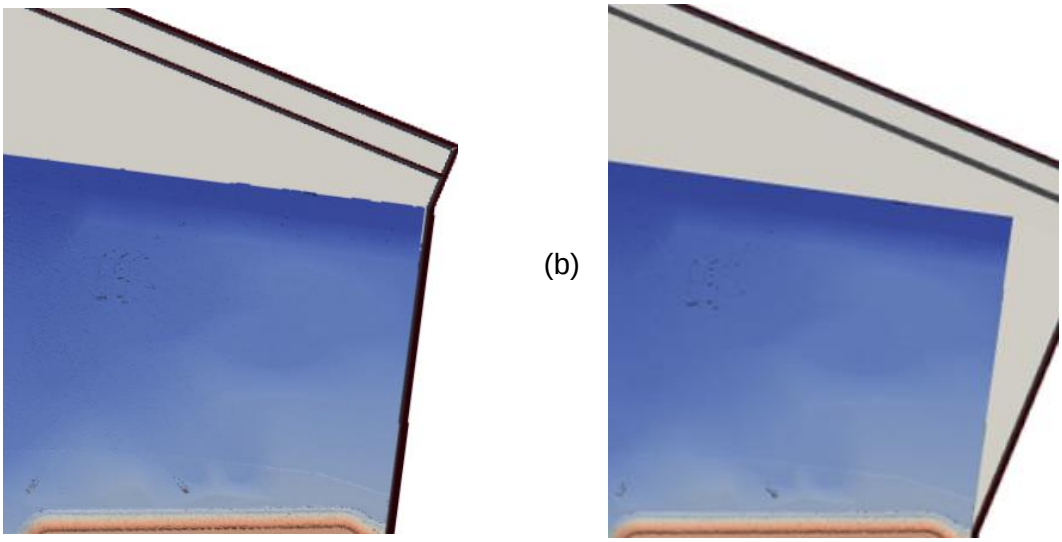


Figure 32: The SPH simulation domain with the angled piston before (a) and after (b) the domain limits have changed. The additional areas have been marked with grey.

Of particular note, however, is the situation on the sides of the domain. As can be seen in Figure 32, part of the wave will hit the west wall, partially reflecting and altering the waves that arrive on the dike. In addition, no waves will arrive at the east side of the dike, where the channel ends.

To avoid these issues, the side walls are redesigned so that they follow the wave direction. This prevents the reflection but requires a part of the bathymetry mesh to be excluded from the computation. Figure 32 shows the new mesh. The excluded bathymetry has been removed using Meshlab; its presence led to irregularities on the intersecting mesh preventing the pre-processing of the new domain. On the east side, the domain is expanded using a flat surface to allow for the wave to reach the entire length of the dike. This domain increase, in addition

to the far larger piston area, leads to an overall increase (about 20%) to the number of fluid particles.

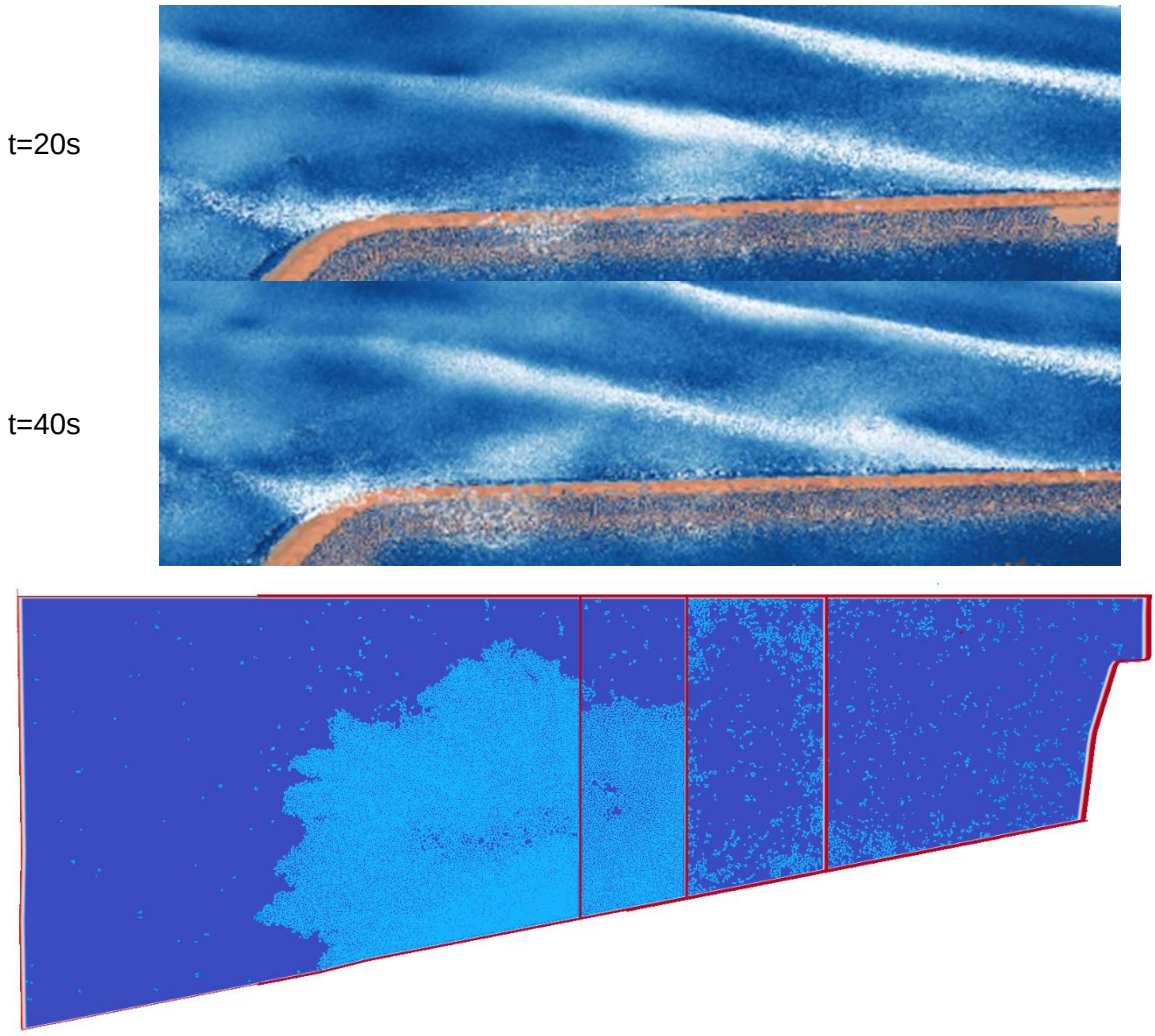


Figure 33: Snapshots of the SPH simulation (Δr=0.017m) with angled piston. Particles overflowing are collected in a tank placed below. The last image is the gathering tank at 60s.

The water volume has, so far, been monitored by measuring the particles present behind the dike. However, during the experiment, the area behind the dike was not flooded. Rather, the water entering the troughs was removed and set apart to be measured after the experiment ended.

To replicate this effect, Meshlab was used to remove the faces and vertices placed at the bottom of the troughs. In addition, a tank divided into four areas, each corresponding to a different trough, was placed under the simulation domain. In that way, water particles entering the troughs drop to the tank below, where they can be collected, while preventing the flooding of the experimental structure.

The results using this method are shown in Figure 33 for a simulation with Δr=0.017 m and $25 \cdot 10^6$ fluid particles. The simulation was executed for 100s and required 80h in 8 Tesla V100 GPUS. The removal of the faces at the troughs has prevented the water particles from overflowing the area behind the dike, unlike the previous simulations as seen in Figure 29. As a result, the area behind the dike is not flooded.

The volume of the collected water particles can be measured to give a more accurate comparison of the simulation with the experimental results. Figure 34 shows a comparison of

the distribution of removed water volume between the simulation and the experiment. The experimental values have been extrapolated from the recorded values at 300s. The distribution of the water volumes among the troughs is very close between the experiment and the simulation.

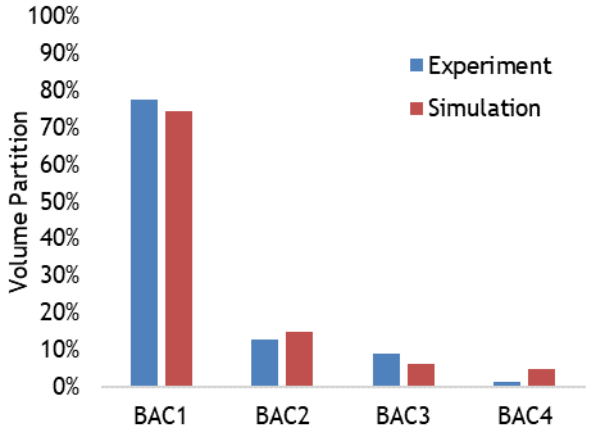


Figure 34: Comparison of the volume partition of the simulation after 100s and the experiment. The experimental values have been extrapolated from the first available values at 300s.

4.3.3. Solitary Wave Simulation

The case so far has been executed exclusively using the conditions of a storm surge: an increased free surface height with regular waves. A different, possibly dangerous sea event, though unlikely around Europe, would be a tsunami (as examined within D1.2 in NARSIS in more depth).

As an approximation of a tsunami wave, a solitary wave will be used (Farahani & Dalrymple, 2014). The wave height will initially remain the same as in the regular wave cases. The generation of the solitary wave will be with a piston according to (Goring, 1978; Goring & Raichlen, 1980). This follows the Boussinesq solitary wave solution where the free surface displacement η can be described as (Korteweg & de Vries, 1895):

$$\eta = H \operatorname{sech}^2 k(x - ct) \tag{11}$$

where the wave celerity c and the wavenumber k are respectively computed by:

$$c = \sqrt{g(H + d)} \tag{12}$$

$$k = \sqrt{\frac{3H}{4d^3}} \tag{13}$$

The paddle stroke S and the duration of paddle motion τ can then be computed by the following:

$$S = \frac{2H}{kd} \tag{14}$$

$$\tau = \frac{2}{kc}\left(3.8 + \frac{H}{d}\right) \tag{15}$$

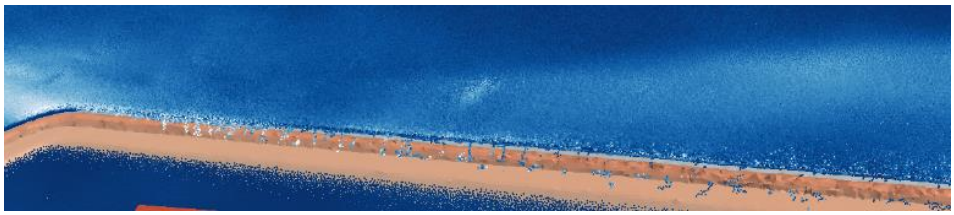
The generation equation is (Goring & Raichlen, 1980; Farahani & Dalrymple, 2014):

$$\zeta^{n+1} = \frac{S}{2}\left(1+\tanh 2\left(\left(3.8+\frac{H}{d}\right)\left(\frac{t}{\tau}-\frac{1}{2}\right)-\frac{H}{d}\left(\frac{\zeta^{n}}{S}-\frac{1}{2}\right)\right)\right), \quad (16)$$

where ζ is the wave generator displacement and n denotes the time step.

Simulating a solitary wave is computationally easier than regular waves, as it is a single event, rather than a succession. Therefore, the overall time of the computation is reduced to 10-15s after the paddle movement compared to 60s in the main simulations. Figure 35 shows snapshots of the simulation for the angled paddle for a simulation with Δr=0.025m and 10^7 fluid particles. The domain used is in fact the same as in the previous section The simulation took approximately 24 hours in two GeForce GTX Titan Black GPUs or 5.5hours on 8 Tesla V100 GPUs.

t=17.4s



t=19.4s

Figure 35: Snapshots of the solitary wave simulation (Δr=0.025m) for the angled paddle.

The snapshots show the moments the solitary wave impacts the dike. Minor overtopping happens in the dike, with some fluid particles remaining after the impact, however, only a small number can be seen in Figure 35. An additional simulation was performed for the solitary wave, using the same domain and resolution, but increasing the wave height by 3, meaning it would now be 0.3875m for the scale model and 27.9m for the natural scale dike; a tsunami wave that is highly unlikely to occur. The other simulation characteristics remain the same.

t=15s

t=17s

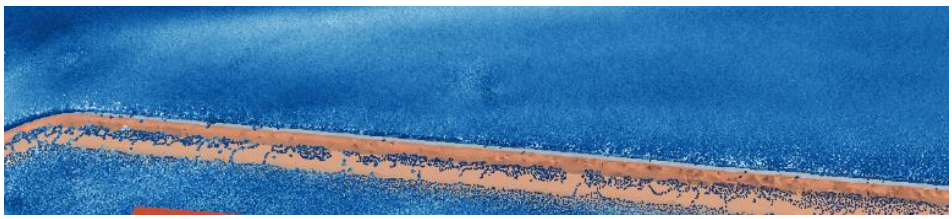


Figure 36: Snapshots of the simulation (Δr=0.025m) including the building data for the angled paddle. The solitary wave height is three times the height of the hundred-year wave.

As can be seen in Figure 36, the increased wave height has, as expected, a more severe impact on the dike. The first snapshot shows that the dike is completely overtopped by the wave, with large volumes of water remaining after the impact. It is likely that any material in these areas would suffer severe damages. However, the dike also absorbs the major impact of the wave. The general analysis for the solitary wave confirms the results of the regular wave simulations, with the dike being overtopped.

This case, however, shows the feasibility of using GPUSPH as a tool for investigating wave impacts even more than the regular wave cases. The need to simulate only a single wave greatly reduces the runtime needed and allows for further refinement of the domain. In addition, using the *resume* function in GPUSPH it would be possible to remove the time needed for the particles settling by storing the necessary hotfiles before the piston begins its movement. As long as the particle resolution and the domain do not change, it would then be possible to specify different wave heights, bringing the simulation time for each instance to a couple of hours for the EDF internal computing cluster.

5 Conclusions and Recommendations

The GPUSPH code has been successfully used in this report to simulate complex geometries for cases with long runtimes in multiple GPU cards. Three cases were investigated, an experimental wave tank with a complex dike geometry and two cases with realistic topography: a simplified representation of a coastline and a scaled model of a dike.

The first part of this report presented the creation of a GPUSPH simulation for measuring regular wave overtopping and its comparison to experiments conducted inside EDF. Good agreement with the experimental results was demonstrated for both overtopping volume and free surface position including the complex flow between the two 'dikes'. The creation of the simulation domain and the selection of SPH parameters was investigated, focusing on the future use of the report for addressing similar flows with GPUSPH.

Potential improvements on these results can be achieved by solving the wave decay issue. The suggested method is the implementation of kernel renormalisation functions (Leroy, 2014) (Mayrhofer, 2014), the δ-SPH approach (Molteni & Colagrossi, 2009), (Antuono, et al., 2009) AND (Marrone, et al., 2011) or an Arbitrary Lagrange-Euler model (Vila, 1999) (Oger, et al., 2016) in GPUSPH.

On the second part, a realistic coastline was modelled, investigating water overtopping a land strip. Overtopping was observed after continually increasing the wave height. However, the volume was insufficient to clear the entire land strip with only residual water pools remaining on it. The case was executed at a 350 m by 350 m domain for 100 seconds of physical time with $\Delta r = 0.05$ m. The simulation took about 20 hours.

A dike scale model used for experiments was simulated using up to $6 \cdot 10^7$ particles. The scaled domain size was 22 m by 20 m (actual domain size 1584 m by 1440 m) and the simulation took 14 days for 60s (corresponding to 510s of real time). Coarser resolutions were simulated up to 100s. Overtopping was observed at the dike, similar to the experiment and the water volume in the troughs qualitatively matched the behaviour at the beginning of the experiment. In addition, a different domain with a 15° angle was tested, showing similar behaviour to the original domain.

For the final test, solitary waves instead of a regular wave train were used. Both a solitary wave with the hundred-year wave height and a wave with three times this height were tested; the results were similar to the regular wave train with the main reactors being unaffected by the impact, which was absorbed by the dike.

The advancements in GPUSPH v5.0 showed that its use is feasible for large-scale projects as it scales well with the use of more GPU cards. Regular wave trains in the dike case required about 25 minutes per wave period for a particle size of 0.025 m; this was increased to about 60 minutes for 0.015m and about 300 minutes for 0.009 m. For solitary wave cases, this means that multiple simulations can be executed in the same day.

The dike case also shows that scaling is a promising way for SPH to deal with larger domains. This would be beneficial for future examinations of the coastline case, where only part of the domain was simulated.

The report also demonstrated the optimal SPH parameters and methods to improve runtime on large cases without sacrificing accuracy, where possible. The new pre-processing SALOME module for GPUSPH was tested and used to create the realistic topography domains. A trained user is estimated to need 2-4 days, depending on the required to resolution to build a new case if a point cloud is provided. Best practices have been listed both for the module and for creating a mesh of sufficient quality.

Some shortcomings can be identified. The pre-processing module could still be improved, as it is still prone to crashing rather than adapting when faced with an anisotropic mesh. And while the computing performance of SPH has been improved, it is still computationally expensive for

cases requiring both fine resolution and simulation of multiple minutes of real time. Variable resolution may significantly reduce computational times in the future (option does not exist in GPUSPH v5.0); the fast progress on GPU capabilities will also assist.

Further advancement in the accuracy of the simulations can be achieved by the inclusion of a sediment flow model, capable of modelling the sediment and the erosion of the dike and suspended sediment within the water. Non-Newtonian models have been included in the latest release of GPUSPH (Ghaïtanellis, et al., 2018) but have not yet been tested in such a case.

6 References

Antuono, M., Colagrossi, A., Marrone, S. & Molteni, D., 2009. Free-surface flows solved by means of SPH schemes with numerical diffusive terms. Computer Physics Communications, 181(3), pp. 532-549.

Batchelor, G., 1967. An introduction to fluid dynamics. Cambridge: Cambridge University Press.

Boufenara, O., 2017. Simulation numérique des efforts et des franchissements dus aux vagues déferlantes sur une protection côtière à l'aide d'un code CFD diphasique., Marseille: ECM, IRPHE, LNHE.

Brezzi, F. & Pitkaranta, J., 1984. On the Stabilization of Finite Element Approximations of the Stokes Equations. Efficient Solutions of Elliptic Systems, pp. 11-19.

Carmigniani, R., 2017. Redresseurs de vagues: vers une novelle stratégie d'extraction de l'énergie houmolotrice, Paris: Université Paris-Est.

Carmigniani, R., Leroy, A., Joly, A. & Violeau, D., 2017. Submerged Plate Wave Energy Converter SPH simulations: Wave Mass Transport. Ourense, Spain, Proceedings of the 12th SPHERIC International Workshop.

Carmigniani, R., Leroy, A. & Violeau, D., 2019. A simple SPH model of a wave pump: Waves above a submerged plate. Coastal Engineering, 61(1), pp. 96-108.

Carmigniani, R. & Violeau, D., 2018. Optimal sponge layer for water waves numerical models. Ocean Engineering, Volume 163, pp. 163-182.

Chamberlain, P. & Porter, D., 1999. On the solution of the dispersion relation for water waves. Applied Ocean Research, 21(4), pp. 161-166.

Colagrossi, A., Colicchio, G., Lugni, C. & Brocchini, M., 2010. A study of violent sloshing wave impacts using an improved SPH method. Journal of Hydraulic Research, Volume 48, pp. 94-104.

Colagrossi, A., Souto-Iglesias, A., Antuono, M. & Marrone, S., 2013. Smoothed-particle-hydrodynamics modeling of dissipation mechanisms in gravity waves. Physical Review E, Feb 87 (2)(0233302).

Crespo, A. et al., 2011. GPUs, a New Tool of Acceleration in CFD: Efficiency and Reliability on Smoothed Particle Hydrodynamics Methods. Plos One, 13 June.

Crespo, A. J. C., Gómez-Gesteira, M. & Dalrymple, R. A., 2007. Boundary conditions generated by dynamic particles in SPH methods. Computers, Materials and Continua, Volume 5, pp. 173-184.

Dally, W. R., Dean, R. G. & Dalrymple, R. A., 1985. Wave height variation across beaches of arbitrary profile. Journal of Geophysical Research, 90(C6), pp. 11917-11927.

Dalrymple, R. A. & Rogers, B. D., 2006. Numerical modeling of water waves with the SPH method. Coastal Engineering, Volume 53(2/3), pp. 141-147.

De Padova, D., Dalrymple, R. & Mossa, M., 2014. Analysis of the artificial viscosity in the smoothed particle hydrodynamics modelling of regular waves. Journal of Hydraulic Research, Volume 52:6, pp. 836-848.

Dehnen, W. & Aly, H., 2012. Improving convergence in smoothed particle hydrodynamics simulations without pairing instability. Monthly Notices of the Royal Astronomical Society, Volume 425, pp. 1068-1082.

Didier, E. & Neves, M. G., 2009. Wave Overtopping of a Typical Coastal Structure of the Portuguese Coast Using A SPH Model. Journal of Coastal Research, pp. 496-500.

Farahani, R. & Dalrymple, R., 2014. Three-dimensional reversed horseshoe vortex structures under broken solitary waves. Coastal Engineering, Volume 91, pp. 261-279.

Ferrand, M. et al., 2012. Unified semi-analytical wall boundary conditions for inviscid, laminar or turbulent flows in the meshless SPH method. International Journal for Numerical Methods in Fluids, Volume 71, pp. 446-472.

Ferrari, A., Dumbser, M., Toro, E. F. & Armanini, A., 2009. A new 3D parallel SPH scheme for free surface flows. Computers & Fluids, Volume 38, pp. 1203-1217.

Fourtakas, G., 2015. Modelling multi-phase flows in Nuclear Decommissioning using SPH, Manchester: University of Manchester.

Ghaïtanellis, A. et al., 2018. A SPH elastic-viscoplastic model for granular flows and bed-load transport. Advances in Water Resources, Volume 111, pp. 156-173.

Gingold, R. A. & Monaghan, J. J., 1977. Smoothed Particle Hydrodynamics - Theory and Application to Non-Spherical Stars. Monthly Notices of the Royal Astronomical Society, Volume 181, pp. 375-389.

Goring, D., 1978. The propagation of long waves onto a shelf, Pasadena: California Institute of Technology.

Goring, D. & Raichlen, F., 1980. The generation of long waves in the laboratory. Coastal Engineering, pp. 763-783.

Grenier, N. et al., 2009. An Hamiltonian interface SPH formulation for multi-fluid and free surface flows. Journal of Computational Physics, Volume 228, pp. 8380-8393.

Guilbert, N., 2018. Modélisation de Vagues et de Franchissements avec Code_Saturne, Paris: EDF R&D - LNHE.

Hérault, A., Bilotta, G. & Dalrymple, R., 2010. SPH on GPU with CUDA. Journal of Hydraulic Research, Volume 48, pp. 74-79.

Hess, S. et al., 2016. Smoothed-particle Hydrodynamics based Wind Representation, Idaho Falls: Idaho National Laboratory.

Korteweg, D. & de Vries, G., 1895. On the Change of Form of Long Waves Advancing in a Rectangular Canal, and on a New Type of Long Stationary Waves. Philosophical Magazine, 39(240), pp. 422-443.

Lennard-Jones, J. E., 1924. On the Determination of Molecular Fields. Proceedings of the Royal Society of London. Series A, 106(728), pp. 463-477.

Leroy, A., 2014. Un nouveau modèle SPH incompressible: vers l'application à des cas industriels, Paris: Université Paris-Est.

Lin, L. et al., 2020. Adequacy evaluation of smoothed particle hydrodynamics methods for simulating the external-flooding scenario. Nuclear Engineering and Design, Volume 365.

Lucy, L. B., 1977. Numerical Approach to Testing of Fission Hypothesis. Astronomical Journal, Volume 82, pp. 1013-1024.

Marrone, S. et al., 2011. δ-SPH model for simulating violent impact flows. Computer Methods in Applied Mechanics and Engineering , 200(13), pp. 1526-1542.

Mayrhofer, A., 2014. An Investigation into Wall Boundary Conditions and Three-Dimensional Turbulent Flows using Smoothed Particle Hydrodynamics, Manchester: University of Manchester.

Mayrhofer, A., Rogers, B. D., Violeau, D. & Ferrand, M., 2013. Investigation of wall bounded flows using SPH and the unified semi-analytical wall boundary conditions. Computer Physics Communications, Volume 184, pp. 2515-2527.

Mokos, A., Carmigniani, R., Leroy, A. & Violeau, D., 2018. Simulating overtopping by regular waves in Channel 5 using GPUSPH, Chatou: Laboratoire d'Hydraulique Saint-Venant.

Mokos, A., Rogers, B. D., Stansby, P. K. & Domínguez, J. M., 2015. Multi-phase SPH modelling of violent hydrodynamics on GPUs. Computer Physics Communications, Volume 196, pp. 304-316.

Molteni, D. & Colagrossi, A., 2009. A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH. Computer Physics Communications, 180(6), pp. 861-872.

Monaghan, J., 1992. Smoothed Particle Hydrodynamics. Annual review of Astronomy and Astrophysics, Volume 30, pp. 543-574.

Monaghan, J., 1994. Simulating Free Surface Flows with SPH. Journal of Computational Physics, 110(2), pp. 399-406.

Monaghan, J. J., 2005. Smoothed particle hydrodynamics. Reports on Progress in Physics, August, Volume 68, pp. 1703-1759.

Oger, G., Marrone, S., Le Touzé, D. & de Leffe, M., 2016. SPH accuracy improvement through the combination of a quasi-Lagrangian shifting transport velocity and consistent ALE formalisms. Journal of Computational Physics, Volume 313, pp. 76-98.

Pahuamba-Valdez, F. et al., 2019. Lagrangian Approach for the Study of Heat Transfer in a Nuclear Reactor Core Using the SPH Methodology. Communications in Computer and Information Science, Volume 1151, pp. 108-124.

Park, J.-C., Kim, M.-H. & Miyata, H., 1999. Fully non-linear free-surface simulations by a 3d viscous numerical wave tank. International Journal of Numerical Methods in Fluids, 29(6), pp. 685-703.

Perić, R. & Abdel-Maksoud, M., 2016. Reliable damping of free-surface waves in numerical simulations. Ship Technology Research, 63(1), pp. 1-13.

Rustico, E. et al., 2014. Advances in Multi-GPU Smoothed Particle Hydrodynamics Simulations. IEEE Transactions on Parallel and Distributed Systems, Volume 25, pp. 43-52.

Ryan, E. D., Savage, B. M., Smith, C. L. & Pope, C. L., 2019. Comparison of free surface flow measurements and smoothed particle hydrodynamic simulation for potential nuclear power plant flooding simulation. Annals of Nuclear Energy, Volume 126, pp. 389-397.

Shao, S. et al., 2006. Simulation of Wave Overtopping by an Incompressible SPH Model. Coastal Engineering, Volume 53, pp. 723-735.

Soulsby, R. & Smallman, J., 1986. A direct method of calculating bottom orbital velocity under waves, s.l.: Hydraulics Research Limited, Report SR 76.

Vila, J., 1999. On particle weighted methods and SPH. Mathematical Models and Methods in Applied Sciences, 9(2), pp. 161-209.

Violeau, D., 2012. Fluid Mechanics and the SPH Method: Theory and Applications. First ed. Oxford: Oxford Univeristy Press.

Violeau, D. & Leroy, A., 2014. On the maximum time step in weakly compressible SPH. Journal of Computational Physics, Volume 256, pp. 388-415.

Violeau, D. & Rogers, B. D., 2016. Smoothed particle hydrodynamics (SPH) for free-surface flows: past, present and future. Journal of Hydraulic Research, Volume 54, pp. 1-26.

Wendland, H., 1995. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. Advances in Computational Mathematics, 4(1), pp. 389-396.

7 Appendix: Run Data Comparison for GPUSPH

Runtime comparisons were undertaken both for simple dam break case (included with GPUSPH) and with the model dike topography. The aim is to investigate the time scales required to run a simulation in GPUSPH and identify the runtime improvements gained from using multiple cards.

Table 7: Computational capabilities of the GPUs used in this deliverable

	GeForce GTX TITAN Black	**Tesla K80**	**Tesla V100**
Host Name	SPH Machine	First EDF cluster (FIEC)	Second EDF cluster (SIEC)
Architecture	Kepler	Kepler	Volta
Release Year	2014	2014	2017
CUDA Cores	2880	2x2496	5120
Clock (MHz)	889	560	1370
Memory Size (GB)	6.144	2x12	16
Memory Type	GDDR5	GDDR5	HBM2
Single precision processing power (GFLOPS)	5121	5591–8736	14028
Double precision processing power (GFLOPS)	1707	1864–2912	7014
CUDA Compute Capability	3.5	3.7	7.0

The different cards used in this study are housed on different EDF structures. The SPH Machine is a local desktop box housing four cards, FIEC and SIEC are the internal EDF clusters. Table shows the different cards, their availability, and computing capabilities. It is expected that the newer Tesla cards in the SIEC cluster, using the new Volta architecture and specialized for computing, will significantly outperform the others. GPUSPH is capable of using one of the two boards present on the K80 card in FIEC, therefore, the GeForce card in the GPUSPH SPH Machine is expected to have a slight advantage over FIEC. All the cards have been constructed by NVIDIA.

The first results presented concern the performance of GPUSPH across different GPU cards. Runtime results are presented in Figure for 1s of physical time for a dam break simulation. As mentioned, the Tesla V100 in SIEC is a significant improvement in computing capabilities. Figure 37 presents the speed up obtained by running the simulation on SIEC; it is close to 4 compared to the SPH Machine and close to 7 compared to FIEC (using half the capabilities of K80).

The simulations presented in Figure 37 only use a single graphics card. Figure 38 presents runtimes for the two faster configurations (SIEC and SPH Machine) when using two or four cards. It should be noted that in the 4-card simulation, SIEC uses two nodes with two cards per node, while the SPH machine uses one node with four cards. The results show that the V100 is significantly faster, as with the single card, but it also achieves very good scaling for multiple cards, above 90%. In comparison, the GeForce card shows scaling close to 40%. The reason for the discrepancy is likely to be the smaller memory bandwidth, cache size and number of registers available to the card, which act as a bottleneck to the simulation preventing from taking full advantage of the increased number of cores.

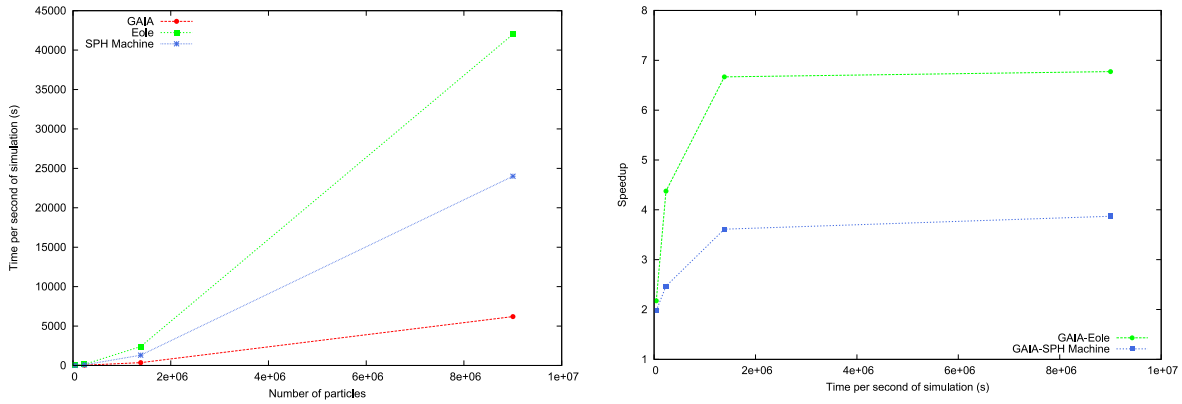


Figure 37: (a) Runtime comparison between the three configurations and (b) speedup gained by the Tesla V100.

The final test considered here is the behaviour of the simulation with a more complex geometry. Figure 39 shows the results for four different resolutions, investigating the runtime for one wave period. Every simulation was executed in SIEC using the V100 cards. The number of cards each resolution has been executed for differs, due to restrictions on memory size. The results show the significant increase in computational time as the resolution decreases. They also show, that depending on the number of particles, there are diminishing returns for using a larger GPU number, as the bottleneck of the simulation starts to be the data exchange between GPUs rather than the computation.

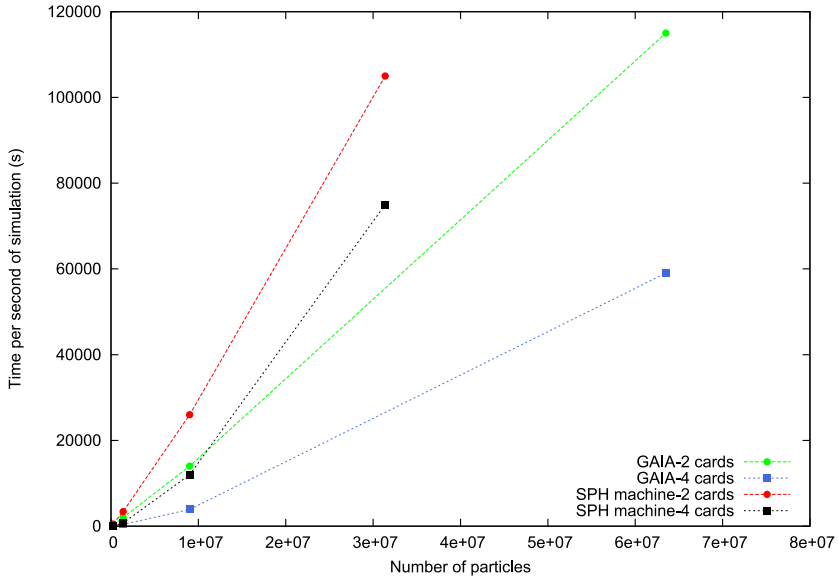


Figure 38: Runtime comparison between the Tesla V100 and the GeForce when using multiple cards.

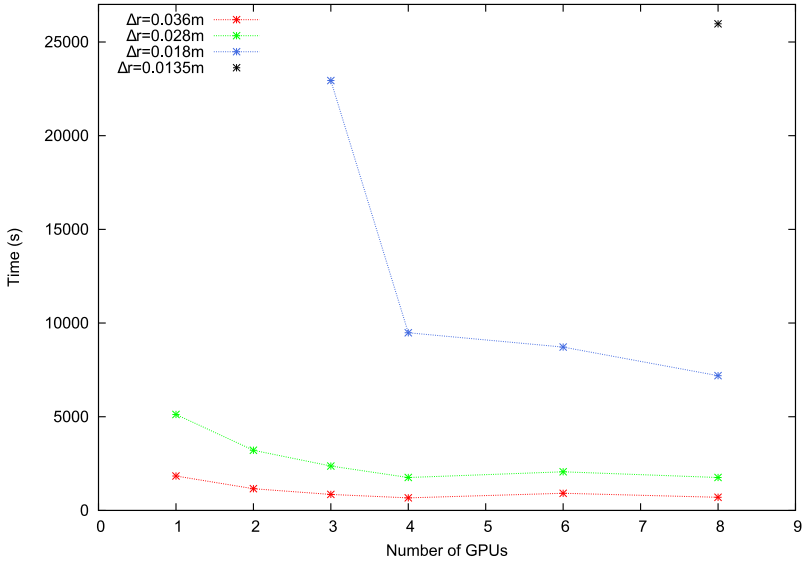


Figure 39: Runtimes and scaling for different resolution for the dike case for one wave period.