Understanding gravel beach formation: Tracking waves and wave-bed dynamics through experiments and simulations

Erietta-Eilikrineia Moulopoulou

Submitted in accordance with the requirements for the degree of Doctor of Philosophy

The University of Leeds
Department of Applied Mathematics

November 2018

Declaration

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

The work in Section 2.2.1 (Chapter 2) and Sections 3.1.2 and 3.2.2 (Chapter 3), have appeared in the following publication:

Kalogirou, A., Moulopoulou, E.E., and Bokhove, O.,Variational finite element methods for waves in a Hele-Shaw tank, *Appl. Math. Model.*,40(17-18): 7493-7503, 2016.

The work carried out in this paper was the result of discussions with Anna Kalogirou and Onno Bokhove. Onno Bokhove contributed the original idea while Sections 3.1.2, 3.2.2 and an initial version of the numerical code, extended by the author in Chapter 4, are attributable to Anna Kalogirou. The experiment (Section 2.2.1), along with the data analysis and the finite volume simulation are attributable to the author.

Acknowledgements

“*As you set out for Ithaka hope your road is a long one, full of adventure, full of discovery.*

Laistrygonians, Cyclops, angry Poseidon-don’t be afraid of them: you’ll never find things like that on your way as long as you keep your thoughts raised high, as long as a rare excitement stirs your spirit and your body.

Keep Ithaka always in your mind. Arriving there is what you’re destined for. But don’t hurry the journey at all. Better if it lasts for years, so you’re old by the time you reach the island, wealthy with all you’ve gained on the way, not expecting Ithaka to make you rich.

Ithaka gave you the marvelous journey. Without her you wouldn’t have set out. She has nothing left to give you now. And if you find her poor, Ithaka won’t have fooled you. Wise as you will have become, so full of experience, you’ll have understood by then what these Ithakas mean.”
(Constantinos P. Cavafys)

As this four-year journey comes to an end, I would like to thank all those people who helped me reach the end of the line. I would like to thank all the members of staff in the University of Leeds for their support; in particular my supervisor Onno Bokhove for his patient guidance, his constant enthusiasm, perseverance and attention to detail, which helped me evolve and often kept me going during the PhD. I am extremely grateful to Anna Kalogirou for her friendly attitude and for always making time for me, especially after her departure from the University of Leeds. Our numerous discussions were filled with valuable advice, positivity and patience that cannot be overestimated. Thank you both for proofreading my work. I am very grateful to Sam Pegler, for the helpful discussions, his invaluable insight into image analysis and for always being willing to meet with me and advise me. I am also grateful to Tom Kent for his help and support, in more ways than one and for proofreading my work. I would also like to thank my research group for the friendly, useful advice and constructive feedback I have generously received throughout the years. Thanks also to my ‘laboratory assistants’ Luca Cantarello and Dario Domingo for helping me set up and clean my wave tank whenever I asked.

I am very thankful to NERC Spheres Doctoral Training Partnership (grant number

NE/L002574/1), for funding this project as well as organising very interesting training events, which were a nice break from academic work. Thanks also to Unicolour for always providing me with food dye for my experiments, promptly and free of charge. Special thanks to Steve Tobias and Stephen Griffiths for their help. I would also like to thank the Astrophysical and Geophysical Fluid Dynamics group, and in particular the postdocs Mouloud, Kumiko and Stephen, for the pints after work. I would also like to thank my examiners, Chris Jones and Nathalie Vriend for reading my thesis and providing me with useful feedback.

A big thank you goes to my parents Giorgo and Eleni, my sister Ioanna, little Thanasaki and my best friend Giorgo whose love, support of my dream to live abroad, and reminding me who I am at difficult times has made all the difference. Thanks to my maths teachers Thanasi and Ria, you greatly influenced my choice to enter this field. I would also like to thank Lauren and Dominic for their support, optimism and for including me in their family. Finally, special thanks to Lydia for the patience, continuous support and encouragement; I wouldn’t have done it without you.

Abstract

Coasts all over the globe are subjected to enhanced erosion as a result of rising sea levels and negative effects of engineering work. In an effort to further understand the processes and timescales of the underlying mechanisms that drive beach formation and erosion, a vertical Hele-Shaw cell, representing a 'slice' of gravel beach, was introduced [90]. Within this set-up, wave-sloshing and beach formation can be observed in a matter of seconds to minutes rather than months to decades. An image-analysis algorithm was developed with the aim of translating images from the experiment into useful data. Its core operation is based on a colour-channel-detection method that can detect and export the depth and area occupied by the water and the moving bottom topography in the tank over time. The simple and easy-to-use algorithm has been tested successfully against a variety of experiments involving water and water-beds, with its novelty being the precise tracking of water infiltrating the moving bed. The acquired wave-only data were imposed as the boundary conditions during the validation process of a shallow-water and a potential-flow shallow-water model, discretised using a one-dimensional finite-volume/Godunov method and a second-order finite-element method respectively [47, 42]. It was found that the finite-volume method handles steep waves of higher amplitudes better than the finite-element method. The tracking results were further analysed with plunging, collapsing and surging breakers present in the examined water-bed cases. It was also found that lower wave frequency cases lead to faster beach formation, with the angles of repose of the corresponding final bed profiles being in the same range as those for real-life gravel beaches [4], confirming the gravel-like nature of the beach present in the Hele-Shaw cell.

Contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Motivation

Climate change is a growing issue in the modern world. One of the impacts of climate change is the increase in temperature, which results in the melting of ice caps and ultimately leads to the rise in sea levels, as well as an increase in the frequency and severity of coastal flooding events [53, 99]. Despite the fact that beach erosion is already a widespread problem, it has been found that increased sea levels can lead to severe flooding, enhancing beach erosion [59]. Engineering works such as dredging, water extraction and the building of dams can also affect beach erosion levels [1, 97]. Some beaches are even called “erosion hotspots” due to the high level of erosion that occurs there [6]. However, it has been observed that erosion in one part of a coastal system creates accretion in another, so that sediment is conserved and not lost [60].

Beach erosion threatens coastal properties and infrastructure such as roads, homes and businesses. Figure 1.1 illustrates the effects that erosion has had in some coastal areas in the United Kingdom. Note how close the cliff face is to the buildings in the left panel. Beach erosion such as this can affect property and the travel of goods and people from one place to another, as well as decreasing revenue from tourism in coastal areas. Erosion combined with the increased sea-water levels causes waves to break further inland, intensifying erosion’s physical and social impact. The

(a) (b)

Figure 1.1: Cliff erosion at Ulrome, East Riding of Yorkshire (left panel). Source: Channel Coastal Observatory [55]. Coastal erosion at Minehead, Somerset (right panel). Source: Channel Coastal Observatory [50].

amount of coastal area lost in Europe due to erosion has been calculated to be around $15km^2$ per annum, with the alarmingly high cost of 3.2 billion euros spent to protect coastal areas at imminent risk of erosion [1].

Current solutions for managing erosion for beaches with high erosion rates include beach nourishment/re-nourishment and building of holding structures such as seawalls to protect the sea fronts (see Figure 1.2). Beach re-nourishment can create opportunities for recreation, as a part of a soft-engineering approach, and reduces the risk of property damage and the impact on revenue from tourism. However, they are not permanent solutions and can often harm beaches and nearshore habitats [70].

Beach re-nourishment involves moving sediment to a beach that has lost sediment and lacks natural re-nourishment. This sediment transportation can be either from an external site, or from the accretion site of the same beach. Re-nourished beaches require constant monitoring and “topping up”, as regularly as every six months to two years [36], resulting in high costs. Beach re-nourishment processes vary from one beach to another. The type of sediment required, the amount needed and the total costs of transport all depend on a number of factors including the

(a) (b)

Figure 1.2: Beach re-nourishment, Pevensey bay (left panel). The dredger is located as close as possible to the beach and is pumping sediment to the beach through the tube. Source: Pevensey Coastal Defence Ltd [49]. Rough seas near Saltdean, Brighton (right panel). Seawall, built in 1933 to protect the coastline as part of the Undercliff Walk. Source: The Geograph, Britain and Ireland project [32].

height and width of the desired beach profile, the rate at which the shoreline is expected to erode as well as the availability and cost of the new sediment. In some cases it is difficult to determine the desired beach profile, and in the case of gravel beaches in particular, the height and the width of the desired profile is determined empirically [36].

Seawalls are another way of preventing coastal flooding and beach erosion. They are solid structures that hold the coastline at the same position thus affecting the natural equilibrium of the coastline. Building seawalls is a medium-term solution, with lower maintenance costs than beach re-nourishment schemes [40], providing protection from a couple of decades up to a century [9]. The main problem with this hard-engineering approach is the fact that seawalls cannot absorb the energy from waves without often causing erosion at the beaches located in front of the seawalls [29]. With the combined effect of waves, currents and storms, erosion is a ceaseless phenomenon. It is therefore crucial to determine alternative ways and strategies to protect our coasts from erosion.

(a) (b)

Figure 1.3: Shingle beach at sunset at Hurst Spit, Hampshire (left panel). Source: Channel Coastal Observatory [2]. Waves overturning as they approach the shingle beach of Leivada at Tinos island, Greece (right panel).

In general, shallower and wider beaches are dissipative, while steeper and narrower beaches are reflective [29]. Gravel beaches, similar to those illustrated in Figure 1.3, form an efficient and natural form of sea defence [75]. Their advantage lies in the fact that they can dissipate wave energy effectively and, in conjunction with beach re-nourishment and a wider and shallower shape, provide the most economic and natural form of coastal protection [9]. Mathematical investigations of the interaction between waves and gravel beaches, although poorly understood within the scientific community, could provide significant insight on how to protect coastal communities from possible changes to beach morphology, in addition to measuring and predicting erosion timescales. These insights could be used to make new beach-protection policies, improving coastal protection whilst adapting to the effects of climate and minimising costs. There thus lies a need to gain further understanding of the interaction between waves and gravel beaches.

1.2 Modelling

Coastal morphodynamics is the study of changes that affect the shape of a river or sea bed over time [88]. It is meant for cases where wave dynamics and the variation of beach morphology over time are interlinked as they interact with each other [83]. Many morphodynamic models have been developed with the aim to investigate the variation of beach morphology over time under the effect of waves, tides or both. These models are instrumental in measuring the impact of defence structures on erosion levels [27], or providing assistance with the design of the latter [45]. In models where the beach is paramount to the study, a moving bottom topography (bed) is present and the wave model is coupled with a sediment transport formula and a bed evolution equation. The sediment transport formula describes the dominant processes, affected by the sediments' own properties, that cause the sediments to move. However, the sediment transport formulas are not well-understood and are highly empirical. The bed evolution equation computes the changes in the bottom topography, given the particles' porosity, and ensures that the computational time scales match real time scales by using a morphodynamic factor. Once the bed equation has been updated, it feeds back to the hydrodynamic equations to evolve the wave dynamics. The wave models involved above are also known as wave- or phase-resolving. While wave- or phase-averaged (spectral) models exist too, they are used independently of the morphodynamic modelling and are not in the spectrum of this work [83, 88]. The main differences between the two types of models are briefly described below.

Wave- or phase-averaged models are based on a spectral energy or wave action balance in which the phase information of the individual waves is not retained [83]. These models are thus more appropriate for cases where computing wave conditions in large spatial domains is significant. They aim to provide a statistical description of the waves in an effort to better understand their random large-scale behaviour. Consequently, these models often fail to describe in detail smaller-scale events such as wave breaking and nonlinear wave interaction in random ocean waves [94].

In contrast to wave- or phase-averaged models, *wave- or phase-resolving* models are time-dependent models which solve the hydrodynamics at all timescales, making them more suitable for small-scale studies [83]. They are often depth-averaged which involves solving the hydrodynamics in a two-dimensional horizontal plane (2DH) using the nonlinear shallow-water equations at shallower depths. The nonlinear shallow-water equations admit discontinuous solutions, thus capturing wave-breaking, and are valid under the assumption of a hydrostatic pressure distribution, negligible vertical accelerations and a velocity profile which does not vary with depth. Alternatively, Boussinesq-type equations are used for intermediate to shallower water depths, which become fully nonlinear and highly dispersive in deep water where the effects of frequency dispersion on the free surface become stronger [16].

Using two-dimensional models poses a problem for modelling particle transport and breaking waves as these phenomena are intrinsically three-dimensional. For example, wave-induced currents such as the bed return flow (undertow) are a direct consequence of the breaking waves and their height variations [41]. Undertow occurs when water that is carried to the shore through the incident waves returns offshore after the waves have reached the coast. During this process, sediment is transported in the shoreward direction. It is hence relevant to the structure of the vertical flow field which is not resolved in a two-dimensional depth-averaged model [62].

In contrast to two-dimensional models, quasi-three-dimensional or three-dimensional models allow for the vertical flow field to vary, accounting for undertow and its effects on sediment transport. The river engineering field is already using three-dimensional approaches successfully, however this is not the case for coastal applications [45]. A model that can take into account three-dimensional properties is needed, combined with a deeper understanding of the main processes affecting coastal sediment transport. To overcome this problem, a thin three-dimensional "slice" of beach can be used so that the particle dynamics can exist within this framework. The vertical Hele-Shaw cell was introduced as the ideal set-up to represent this thin slice of beach and investigate the interaction between water and sediments [90].

The Hele-Shaw cell was originally introduced by Henry Selby Hele-Shaw in 1898 with the aim to investigate flow through parallel plates of a narrow width and the displacement of one fluid by another [37]. It consists of two parallel plates placed within close proximity of each other, allowing the water flow to be laminar with a Poisseuille profile in the narrow direction and a small Reynolds number. Since then, the Hele-Shaw cell has had applications in coastal engineering [15, 42, 90] tissue engineering [22], biomedical engineering [95] and the oil industry [19]. It has been used to investigate viscosity-driven instabilities in two-phase flows such as viscous fingering [66, 67, 71, 84], or droplet relaxation in microfluidic flows [17]. The viscous fingering, also known as the Saffman-Taylor instability, occurs when an injected less viscous fluid displaces a more viscous fluid. Hele-Shaw cells have also been used to investigate the Kelvin-Helmholtz instability [57, 73], as well as free surface waves, either inertia-dominated thus creating Faraday waves, or viscosity-dominated to achieve reduction of the sloshing motion in a closed tank [76], [98].

In this project, the vertical Hele-Shaw cell has been employed to investigate the interaction between forced water waves and a bottom topography and how the interaction of the two leads to phenomena such as sediment transport, breaking waves and beach formation. The topography consists of zeolite particles inserted into the tank, the diameter of which is found to represent coarse gravel sediments [88]. The narrow width of the cell in the lateral direction, a property attributed to its vertical rather than horizontal position, has transformed the tank into a "slice of beach" and allows for the velocity variations in the lateral direction to be ignored. Waves are thus created, propagate in the tank and reflect once they reach the right wall of the tank. In the full case, the topography is present with the profile evolving in the cross-shore direction as a result of the incoming waves. Inertia, rather than viscosity, dominates flow in the vertical Hele-Shaw cell due to the wave-forcing attributed to the operation of the wave-pump. The gap width between the two glass plates of the Hele-Shaw cell is based on the optimised plate width in article [90], and was thus chosen carefully as it allows determination of the effects of inertia on the flow and can lead to the flow's deviation from the Poisseuille profile. The advantages of this set-up outnumber its disadvantages [15]:

1. Its experimental design is easy to grasp whilst providing decent experimental data for the validation of the mathematical models,
2. the quasi-two-dimensional dynamics significantly decrease computational efforts, and
3. turbulence is greatly reduced so that attention can lie mostly on the interaction between the waves and the beach.

The disadvantages are mostly related to the existence of momentum dissipation and surface tension due to the close proximity of the glass plates. At these small scales, surface tension effects are more likely to occur as the water molecules that make up the surface of the water tend to cling to the glass plates. Fortunately, these effects can be modelled and managed experimentally to a fair degree. For example, the momentum damping effect can be optimised by using a suitable gap width, while the surface tension can be overcome by adding alcohol to the water to minimise the water's adhesive qualities.

1.3 Sediments

The material of seabed forms and associated sediments are roughly classified into three categories: mud, sand and gravel; a classification based on their diameters and cohesion. Mud, *i.e.* clays and silts, consist of sediments with diameters between 1-60μm, while sand grain diameters usually lie between 0.06-2mm. The diameter of gravel such as granules, pebbles, cobbles and boulders fall between 2-300mm [28, 88]. Of course in nature, beaches consist of sediments with more than one diameter and even more than one type of sediment; a byproduct of the cumulative effects of wind, water waves and storm events on beaches over the decades. Some of these beaches, also known as mixed beaches [48], may have sediments with diameters ranging three orders in magnitude; for example from fine sand (0.1mm) to small boulders (300mm).

The transport of sediments in the oceans or in river beds is highly dependent on the sediment diameter. Gravel rolls, slides or jumps on a layer above the bed, always staying in contact with it [61, 83, 88]. This type of sediment transport, called bedload, reacts directly to local flow conditions and requires a substantial amount of work to trigger and maintain its motion. Bedload transport is usually initiated when a threshold value for the bed shear stress is exceeded, a condition also known as “initiation of motion”, originally described by Shields [28, 87]. As a result, the waves that manage to lift the gravel dissipate energy. Sand, unlike gravel, is generally transported by suspension: it is carried by the turbulent fluid flow and has less frequent contact with the bed. These effects are visualised in Figure 1.4.

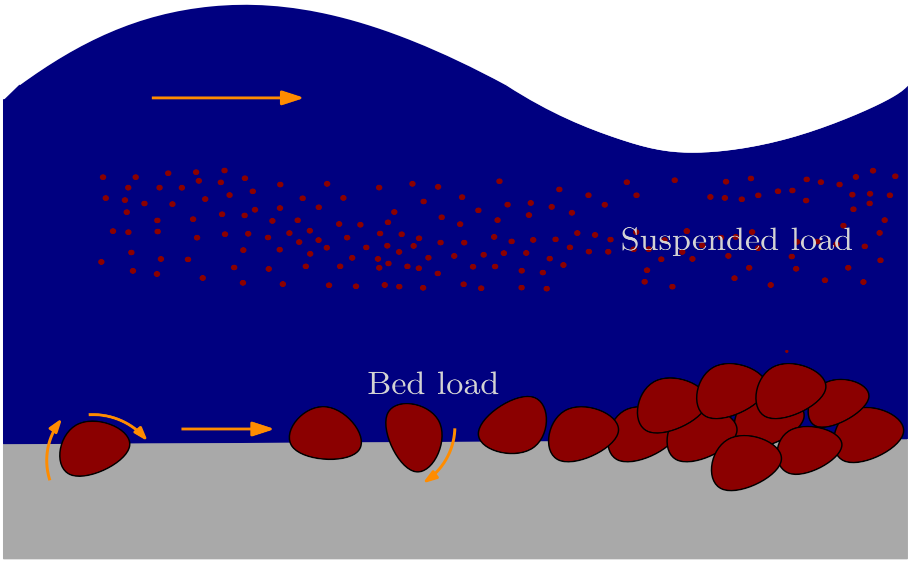


Figure 1.4: Bedload and suspended load sediment transport initiated by a wave travelling from left to right. The bedload is at the bottom with gravel sliding and rolling, while suspended load is occuring across the entire water column.

This study focuses on the formation of shingle beaches, in particular within the specific aforementioned experimental setting of the Hele-Shaw cell. Shingle is defined as water-worn rounded gravel. When dealing with gravel beaches and their profile variation over time, there are some factors that can affect the evolution and the final shape of their profile. These factors can roughly be divided into two groups: those related to the water and those related to the bed. Factors related to the water include the wave frequency, wave height, wave duration and the angle of wave attack [75, 96]. These are also called environmental influences. Factors related to the bed are the sediment size and grading, the thickness of the bed and the initial bed profile. It is thus useful when experiments allow control of these aspects as a greater understanding of their influence on beach formation can be achieved.

In nature, changes in the beach profile often require decades to manifest, even though beach dynamics dramatically changes mostly due to storm events. Having a set-up that can investigate beach formation in a matter of minutes rather than years offers an ideal opportunity to rapidly observe the dynamics of beach profile changes and the transition from one beach shape to another.

1.4 Objectives

This work was two-sided as it entailed both modelling and experimental work carried out in an effort to combine and contrast theory with reality. After all, models work well when they do justice to the phenomena they describe. The modelling aspect consisted of the following objectives:

1. Determine which mathematical models can capture the dynamics of waves in a vertical Hele-Shaw cell. This objective is addressed in Chapter 3.
2. Measure how close the chosen mathematical models are in relation to reality, and to adapt the models if necessary in the light of experimental results. This objective is addressed in Chapter 4.

The experimental aspect aimed to:

3. Perform wave-only experiments for various wave frequencies and water depths and extract useful measurements of the water depth, through a bespoke tracking algorithm. Then, insert the measurements to the mathematical models and challenge their performance. This objective is addressed in Chapters 2, 4 and 5.
4. Perform experiments with both water and particles. Vary the wave frequency, water depth and wave height so as to investigate beach formation and beach-related phenomena such as sediment transport and wave breaking. Then, collect measurements of the bed height and the water depth from the experiment by extending the tracking algorithm. This objective is addressed in Chapter 5.

1.5 Outline

The objectives mentioned in the previous section have been tackled within the thesis in the following order. Chapter 2 introduces the experimental set-up in detail and summarises the water-only experiments that were carried out for different frequencies and depths. An image-analysis algorithm, developed in Matlab with the purpose to track the water depth, is introduced and described. After the process of translating images to raw numerical data has been fully described, snapshots of the experiments compared against the tracking results follow. In this way the efficiency of the algorithm can be fully fathomed.

Chapter 3 introduces the one-dimensional nonlinear shallow-water and potential-flow shallow-water equations, extended with a momentum dissipation term due to the narrow width of the tank in the lateral direction. A time-dependent forcing term is also introduced, to account for the wave sloshing initiated from the aquarium wave-pumps, and is manifested in the boundary conditions that accompany the models. Furthermore, the finite volume and finite element methods used to spatially discretise the equations of each model are also introduced, with a particular focus on the implementation of the boundary conditions.

The validation of the models ensues in chapter 4. It entails processing the raw data, previously acquired in chapter 2, altering the boundary conditions and gradually imposing the processed data to the latter. The novelty of the validation process lies in the fact that the data enters the model through the boundary conditions, which are implemented in a different way for each numerical method. Last but not least, the results stemming from the validation of each method are presented and contrasted with each other, in order to elucidate the performance of the models and the effect that the data have on their approximate solutions.

Chapter 5 introduces the full experimental set-up and describes water-bed experiments, of varying wave frequency and depth, ultimately leading to beach formation through their constant interaction. The extended image-analysis algorithm, initially presented in chapter 2, is presented. Its novelty lies in the fact that it simultaneously tracks the changes in the water and the bed morphology throughout the duration of each experiment efficiently and precisely as part of the same algorithm. Snapshots of the experiments are overlaid against the tracking results so as to illustrate the effectiveness of the extended tracking algorithm. The successful tracking of the water and the bed allowed for further investigation of the creation of breaking waves, the measurement of the bed angles, and the shape of the final bed profiles. In the quest for the parameters that affect beach formation, the tracking results of three experiments have been used to determine the observed breaker types and their timescales, the final bed profiles and their corresponding angles of repose. The determined angles are then compared against known angles of repose for shingle beaches. To conclude, chapter 6 provides a summary of the work carried out, discusses key results and suggests directions for future work.

Chapter 2

Experimental tracking of the wave dynamics

The interest behind this chapter was on water-only experiments as the means to acquire useful experimental data. These data involved time series of the depth and the area flux of water and they were to be used as a means of validating the mathematical models. Determining the depth would then allow for the calculation of the area flux of water entering the tank. An image-analysis algorithm that uses snapshots of the experiments to track water waves and translate them into depth measurements was developed and is described below. Essentially, the input of the algorithm are images and the output are time series for the depth and the area flux. The biggest part of the processing has been spent on finding the optimal way to translate the water line from an image to its corresponding depth data.

Two rounds of experiments were carried out, each in a different wave tank. The Hele-Shaw set-up and the dimensions of each tank can be found in Section 2.1, with the experiments performed in each tank presented in Section 2.2. The developed image-analysis algorithm, its tracking methodology and the obtained experimental data are described in Section 2.3, with the conclusions following in Section 2.4.

2.1 The Hele-Shaw cell

The Hele-Shaw cell was used as the ideal set-up to investigate beach formation under the effect of waves. In the absence of a beach the set-up, illustrated in Figure 2.1, includes a tank (cell) consisting of two parallel glass plates of equal length, placed vertically at a fixed distance apart from each other. It also comprises of a set of two underwater aquarium pumps, carefully placed in a bucket hanging under the table (see Figures 2.2b and 2.2c), generating waves that propagate along the tank. These pumps are operated by an Arduino Uno motor controller and are programmed by a laptop. Essentially, a script determining signals is loaded to Arduino through a usb connection, indirectly regulating the speed of the pump's motor and the frequency and steepness of the waves created. Choosing between two tanks of different lengths and widths was possible (see Table 2.1), while the rest of the set-up remained the same. The only extra equipment required, was an average digital camera and an efficient background lighting illuminating the water line.

Table 2.1: Dimensions of the wave tanks used during the experiments. Tank one was slightly longer and wider than tank two. Tank one was used in [42].

Tanks	***Length*** (m)	***Width*** (m)	***Height*** (m)
1	0.57	0.0022	0.20
2	0.52	0.002	0.18

2.2 Experiments

All experiments were initialised in the same way; the bucket was partially filled with coloured water to a still-water depth H_0 before the pumps were switched on. Water-soluble blue food dye powder, selected mainly due to its nontoxic nature and the uniformity of the colour that it provided, turned the water blue thus helping achieve a better visualisation of the flow. Alcohol was also added in the water so as to minimise the effects of surface tension in the flow. The lights were switched on, constantly illuminating the water line and a Nikon camera D3300 was placed

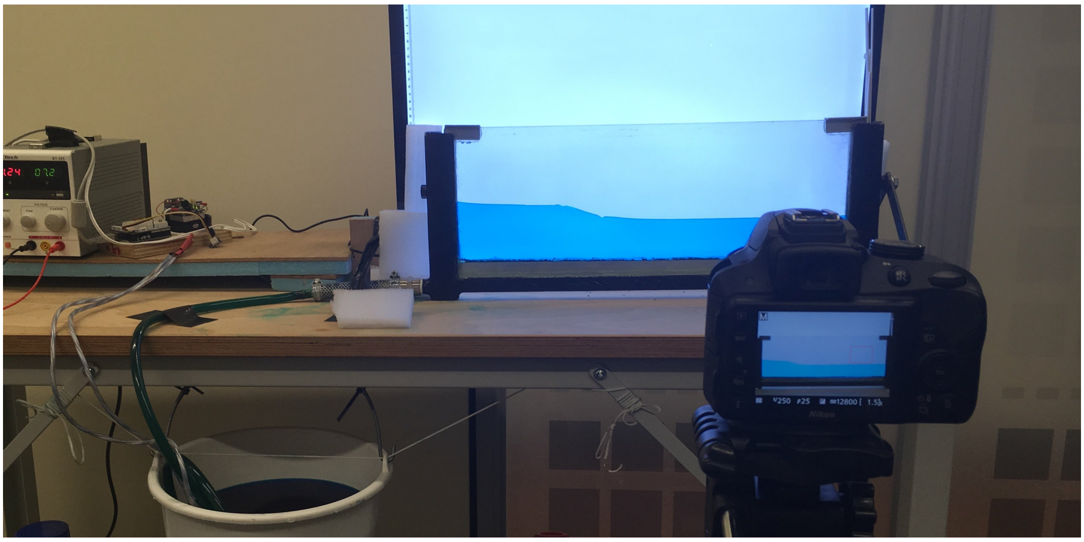


Figure 2.1: Hele-Shaw set up (tank two). Waves propagating in the cell as a result of the operation of the submersible wave-pumps. The Nikon camera was placed in front of the cell and recorded the propagating waves. Two stilling wells (black chambers) are located on the left and the right of the glass plates. A power supply of 12V (white box) is connected to the Arduino.

(a) (b) (c)

Figure 2.2: Addition of 22 mL of ethanol to the dyed water (left panel). Pair of submersible wave pumps with flow rate 12 litres per minute and water pressure 0.6 bar (middle panel). Wave pumps submerged in a bucket filled with 10 litres of dyed water (right panel).

parallel to the cell so as to record the wave propagation. The core pump operation was essentially one; they repetitively filled and emptied the tank, with water under the effect of gravity. Choosing appropriate combinations for the emptying (downward) and filling (upward) phase of the tank, as well as the time required for each phase to happen, resulted in waves of varying speeds, steepness, frequencies and wave heights. When the pumps were switched off, the tank was emptied out of all water. The wave speed and wave steepness were also influenced by the distance between the bucket and the tank; the larger the distance between the tank and the bucket, the lower the incoming speed and steepness of the incoming wave were. This effect was due to the angle of the hose, connecting the pumps with the water. Additionally, changing the voltage from the analog dials affected the minimum and maximum wave depth whilst conserving the preset wave height.

2.2.1 Tank one

During the first round of experiments, tank one was used with the lights focused on the sides of the tank (see Figure 2.3). This round of experiments was performed in an effort to determine an initial but optimal experimental setting which would assist the water tracking. Through trial and error and with the available resources in mind, the position of the camera the colour of the lights and the dye were determined. An appropriate algorithm was then developed for the tracking of the water line, subject to the corresponding limitations of the experimental set-up. Ultimately only one case was selected from tank one for further tracking and processing, and has been summarised on Table 2.2. The reason for this choice was the background lighting that allowed for the water waves to be captured in a clearer way compared to the other cases, providing improved tracking results. The data set originating from this experiment was then used in [42] to validate the mathematical models. The edge detection strategy related to this work has been described in Appendix A.1.

In spite of the polished design of the tank, the slightly larger gap width of about 0.002m which had been adjusted proportionally to the longer length of the cell, created problems during runs of the experiment where the particles were present. Plenty of particles would get stuck between the plates, thus obstructing sediment transport and beach formation. Consequently tank two was used to alleviate this problem, by only replacing the main wave tank and the background lighting while keeping the rest of the set-up the same.

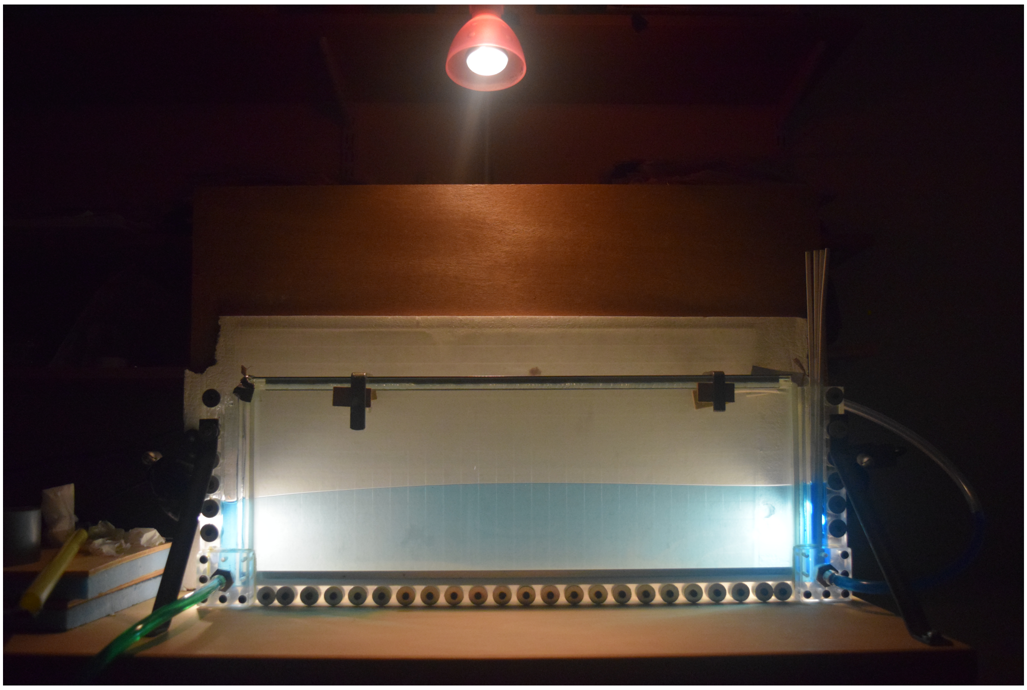


Figure 2.3: Hele-Shaw set-up (tank one). This wave-tank was slightly longer than tank two. Two individual LED lamps were placed near the left and right wall and a normal lamp at the top of the cell.

Table 2.2: Water-only experiment performed in tank one.

Video	**f** (Hz)	**Wave depth** (max)	**Duration** (min)	**Figures**
47	0.7	Very deep (9.6cm)	00:21	A.4, A.5

2.2.2 Tank two

During the second round of experiments, and with the set-up more or less determined, the focus was on improving the image tracking for tank two; not only through improving the image tracking method but by improving the experimental setting too. The location of the lights was identified as troublesome; focusing the lights on a specific region rather than having a uniform background colour, even though successful at tracking the regions near the walls, failed to help with the tracking of the area in the middle of the tank. Especially when using a tracking method such as edge detection, further explained in Appendix A.1, which largely relied on the location of the lights. Hence, uniform colour was accomplished by switching from individual LED lamps to an A2-sized LED light bank same with those seen in Figure 2.1. This LED light bank provided colour uniformity and was placed behind the tank. In this way not only the wave propagation phenomenon was much more visible compared to before, but the shadows which were previously present due to the smaller range of light from the lamps, were eliminated.

Another improvement that was put into effect, was a change in the camera settings. Increasing the aperture and decreasing the shutter speed of the camera manually before each recording began, provided snapshots with a larger contrast between the water and the background colour; an effect which proved very useful during the processing stage. The experiments performed in tank two have been summarised on Table 2.3.

Table 2.3: Water-only experiments performed in tank two with frequencies ranging from 0.33Hz to 0.67Hz. Duration refers to how long each experiment was recorded for.

Videos	f(Hz)	**Wave depth** (max)	**Duration**(min)	**Figures**
149	0.67	Medium (6.5 cm)	00:43	2.11, 2.12
154	0.48	Very deep (9.4cm)	00:45	A.7, A.8
165	0.33	Shallow (5.5cm)	00:43	2.13, 2.14

2.3 Tracking water waves

2.3.1 An introduction to digital imaging

Digital image processing is the procedure of manipulating images by developing and using suitable programs in a computer. Image analysis refers mainly to the extraction of useful information from these images [18]. In the literature, the lines are blurred between where the processing stops and where the analysis begins [33].

Image processing is an area of increasing interest; its numerous applications vary from detecting features and improving the diagnosis in biomedical imaging [23, 77], to remote sensing [82]; where satellite images depicting areas damaged by flooding [26] or earthquakes [102] to capturing water erosion [100] and monitoring crop conditions in the agricultural field [104]. The images are often processed extensively so as to provide useful information. The variety of the applications of image processing along with the increased computer capabilities have rendered image processing a crucial step towards acquiring useful information by calibrating, correcting and analysing digital images in various softwares.

Each image consists of elements with a horizontal and vertical component with a unique location and specific value, called pixels. From a mathematical point of view, pixels can be regarded as the coordinates (x, y) of a two-dimensional image array M with elements $f(x, y)$, where $x = 1, 2, ..., n$ and $y = 1, 2, ..., m$. The array is then of the form:

$$M = \begin{bmatrix} f(1,1) & f(1,2) & \dots & f(1,m) \\ f(2,1) & f(2,2) & \dots & f(2,m) \\ \vdots & & \vdots & \\ f(n,1) & f(n,2) & \dots & f(n,m) \end{bmatrix}.$$

Each entry in the image array is called intensity and measures the amount of light captured by each pixel. For example in greyscale images, which are also called monochromatic because of the lack of colour, black is 0 and white is equal to 255 with the intensity values being in the range of $[0, 255]$. The product of the vertical by the horizontal pixels, $m \times n$, is called image resolution and it refers to the total amount of pixels in an image. Thus, the more pixels in an image (more pixels per unit distance) the smallest the area of the image depicted in each pixel, hence the more accurately this image is described.

Images obtained during the experiments, with a rate of fifty frames per second, described in Section 2.2 have a resolution of 1980 × 1080 pixels or 1280 × 720 pixels. For experiments with duration less than one minute, higher resolution images were preferred. For experiments that lasted longer than one minute the volume of the images was too big, ranging from a couple to hundreds of thousands snapshots per experiment, hence the lower resolution was selected so as to avoid a potential data overload.

Useful information can be exported from these coloured snapshots of the experiment provided some processing and analysis. In this thesis, image processing consists of two stages; image pre-processing and main processing/image tracking. Image pre-processing aimed to prepare the raw images for the main processing that followed. Then the object in question was tracked in the main processing stage in image coordinates and was ultimately translated into useful data (image analysis). Matlab was the selected software, used throughout all stages of image processing. It is simple to use and favours performing tasks iteratively, especially for large data sets, while providing further support through specific toolboxes with useful preset functions, such as the Image Processing Toolbox.

2.3.2 Image pre-processing

Image pre-processing during this work entails exporting images from each video, segmenting and organising them in each folder by assigning appropriate sequential names to them. Videos consist of sequential images captured under a specific rate (frame rate). The advantage of using recorded videos to taking images lies in the fact that videos have a higher frame rate compared to images. Higher frame rates translate to more images, hence more information, per second. The output images were then saved in a portable network graphics (PNG) format. This universal format provides better compression rates, i.e the image quality does not degrade when saving or viewing the image, while it supports a full spectrum of colours [7].

Image segmentation was the last step of the image pre-processing. It was essential as it ensured that only the physical area of interest remained in each image with the unnecessary and potentially troublesome background information being discarded. Proper care was taken to adjust the image resolution of the segmented images to the physical area depicted within the images. In order for the segmenting to be optimal, the position of the camera during each experiment was fixed.

2.3.3 Main processing

Main processing refers to the steps taken to further manipulate the images and track the object in question, with the input and the output being images. Depending on the method used for the tracking the main steps are roughly classified into:

1. Loading the segmented images,
2. further processing, thresholding and tracking object(s) within each image.

In a nutshell, the image processing and analysis performed within this chapter aimed to primarily track the water line location. Each step of the main processing has been further explained below, with a particular focus on step two of the main processing; a step highly dependent on the tracking method used.

2.3.3.1 Loading segmented images

The first step of the processing was to load each image, in an iterative way, in Matlab. Each image was then translated into an image array, with a form similar to the one described in Section 2.3.1 and a number of rows and columns determined from the resolution of the segmented image.

2.3.3.2 Further image processing and tracking of the water line

When tracking the shape or the outline of a one-dimensional object, colour or edge detection methods are usually preferred. Both methods track objects through different mechanisms. During this thesis, both of these methods have been used. *Edge detection*, colour or monochrome, is a method used for locating the boundaries of objects within images by locating the discontinuities in brightness (gradient method) [18, 72]. Monochrome edge detection was used in earlier work, with the corresponding experiment carried out in tank one and its obtained data used in [42]. At the time it was the most suitable method to detect the free surface line, in view of the specific light settings. Further information on edge-detection can be found in Appendix A.1.

Colour-channel detection, uses properties of coloured images to locate whole regions of similar brightness rather than just focusing on the location where the brightness changes. This method was used for tank two after the improvement of the experimental setting had been accomplished. The change in the lights and the camera settings, paved the way for the colour-channel detection to work more precisely and effectively. The performance of each method was tested against each other for experiment 154 in Appendix A.2, with the same tank and experimental setting present. Indeed the colour-channel-detection method yielded better results and is further discussed in the following section.

Tracking the water line with colour-channel detection

Colour-channel detection tracked the water depth by optimising the information 'hidden' in the snapshots. The steps leading to the acquisition of the water line have been summarised below and are illustrated in Figure A.6 (see Appendix A.2).

(i) Selecting an appropriate colour channel

In order for colours to be adequately described, colour models with different coordinate systems have been created. In essence, each colour is described by a specific combination of colour coordinates provided for each coordinate system. Depending on which device is being used, different colour models are utilised. For example, colour monitors, TVs and video cameras use the Red Green Blue (RGB) colour model while printers use the Cyan Magenta Yellow and Black (CMYK) model [7, 33], to name a few. The work described here has been carried out using the RGB colour model.

The RGB colour model is based on a cartesian coordinate system with axes $(x = R, y = G, z = B)$. The advantage of the this model lies in the fact that by using red, green and blue as the primary colour palette any other desired colour can then be described uniquely as a combination of these three colours in the range of [0,1] (see Figure 2.4) without sacrificing the image algorithm's accuracy or the image quality [7]. Hence each image obtained from the experiments consists of three colour channels; a red, green and blue with different sensitivity to depicting light. Hence each channel contains information and the composition of all three channels results in a coloured

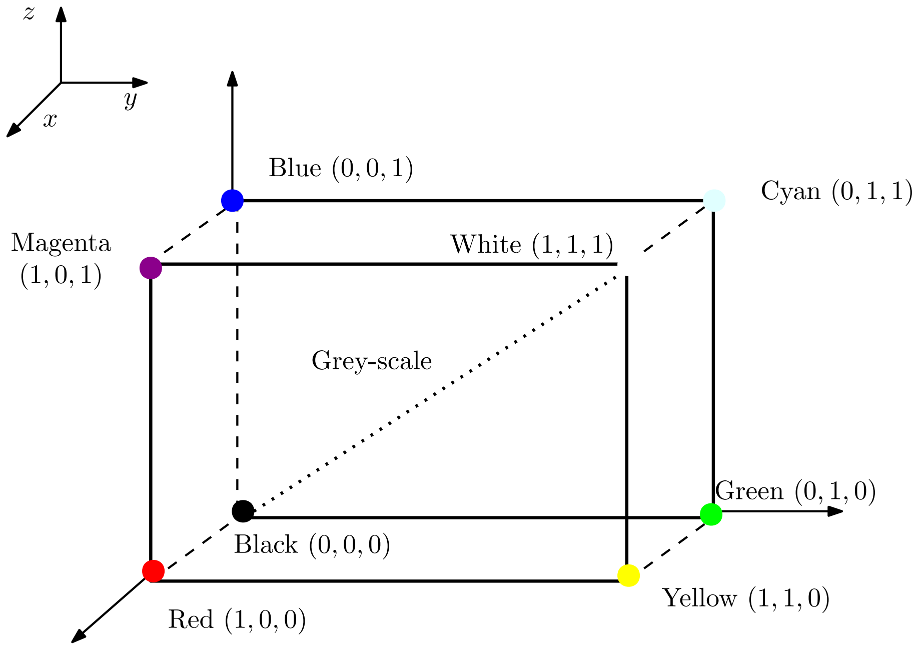


Figure 2.4: RGB cube, as determined in [18]. Colours expressed in RGB coordinates with axes $(x = R, y = G, z = B)$.

image obtained by a camera, as sketched in Figure 2.5.

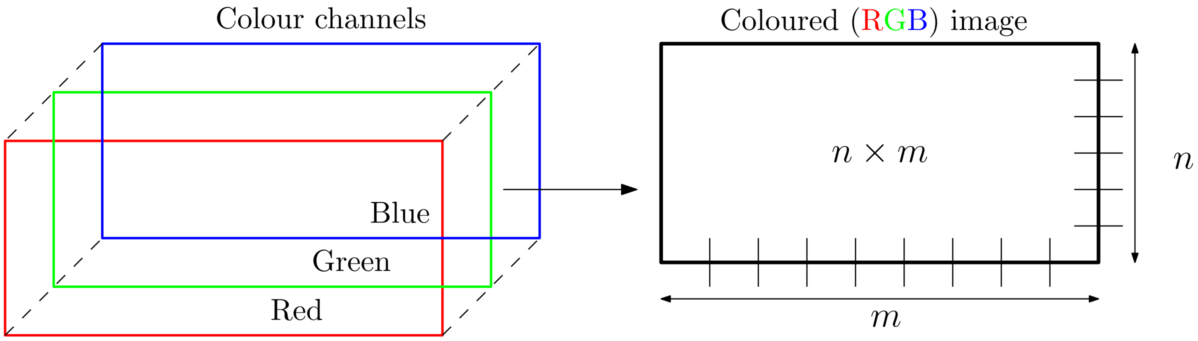
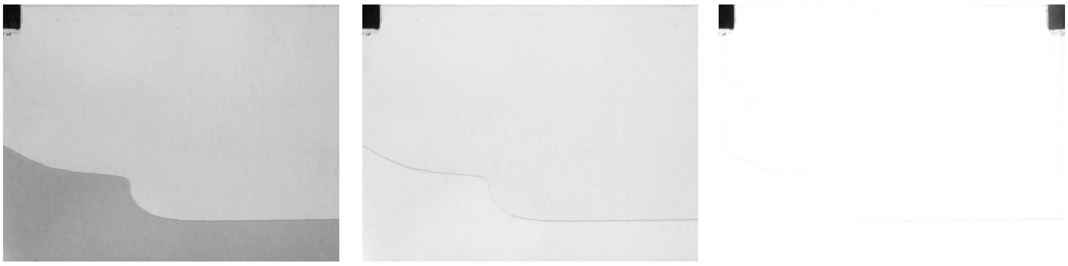


Figure 2.5: RGB colour channels (left panel) composing a coloured image of resolution $n \times m$ pixels (right panel).

Working backwards to Figure 2.5, *i.e.* starting from the coloured image and using information stored in the colour channels, can yield useful information. For example, if a coloured image has blue colour in some parts of it, when observing the blue colour channel of the image, the blue is ignored and turned to white. Hence, it is important to use the colour channel that is most appropriate for either the colour that is wished to be ignored or further intensified.

Figure 2.6: Original (cropped) image from experiments.



(a) Red (b) Green (c) Blue

Figure 2.7: Comparison of the three colour channels applied on image 2.6. The blue channel has eliminated most of the image, while the highest contrast between the background and the wave shape has been achieved by the red channel.

Working in a similar way, tracing the water line required for a specific colour channel that would make the water stand out from the background. A comparison of the three colour channels has been carried out for one of the experimental images and is illustrated in Figure 2.7. Because the water and the background colour were blue, the red channel was optimal (Figure 2.7a) as the shape of water was depicted well and was distinguishable from the background.

(ii) Histogram equalisation and thresholding

The histogram of a monochrome image depicts the intensity values found in that image, between 0 and 255, by organising them in groups of pixels with the same intensity magnitude. These groups of similar intensities, are also called classes and are separated by a valley. When the classes are nearby it translates to the image having similar intensities and hence being poorly contrasted; a setting not ideal when detecting a sub-region of the image. Tracing the water line as efficiently as possible, required for the contrast between the water and the background to increase through histogram equalisation.

Histogram equalisation is an enhancement technique applied in the spatial domain; it is based on manipulating pixels in the image, in contrast to frequency domain techniques applied at the Fourier transform of the image [72]. It transforms narrow-located classes to further spread-out classes of a more uniform shape. In other words, the intensities closer to white get whiter and the dark ones darker thus increasing the distance between the classes. The effect of histogram equalisation on one of the snapshots is visualised in Figure 2.8.

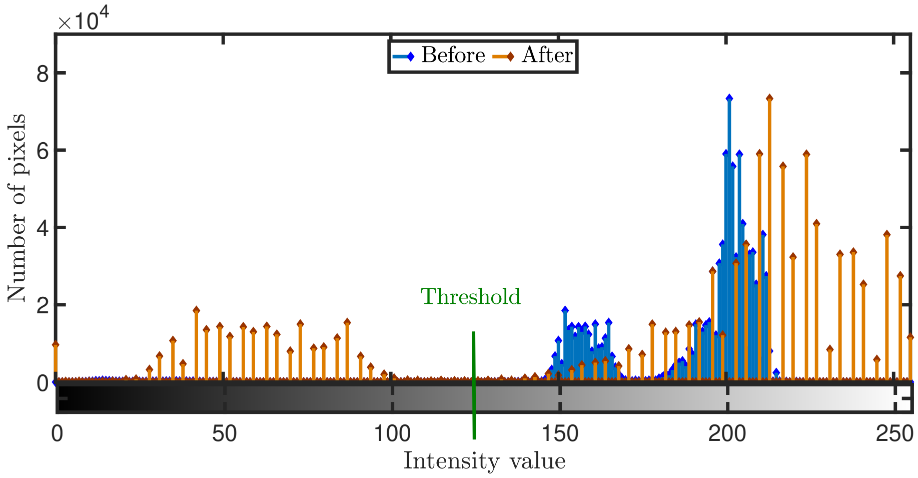


Figure 2.8: Effect of histogram equalisation on intensity values. Before histogram equalisation (blue) two classes/groups of intensities were closely located to each other in the range of 150-220. After histogram equalisation (orange), the classes moved further away from each other to the broader intensity range of [20, 255], providing a higher intensity difference between the darkest and the brightest pixels. A valley formed between the intensity classes at 135.15. The threshold value (TH) (green) is thus TH=135.15, with its value normalised to TH=0.53.

With the contrast enhanced, thresholding was optimised. Thresholding in essence occurs by selecting the intensity value that is found at the bottom of the valley in the image histogram. Using one valley to determine the threshold is called global thresholding whilst if more than one valleys/values are used it is called local thresholding. This threshold value (TH) then acts as a criterion to create a binary image, *i.e.*, a black and white image similar to Figure 2.9. If $f(x, y)$ is the image matrix before thresholding, and $g(x, y)$ is the matrix of the binary image determined by a global threshold TH then:

$$g(x,y) = \begin{cases} 0 & \text{if } f(x,y) < \text{TH} \\ 1 & \text{if } f(x,y) \geq \text{TH} \end{cases}.$$

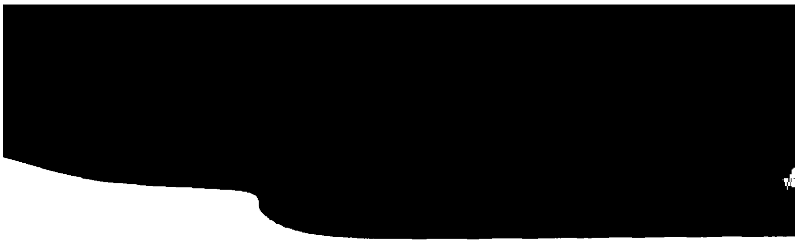


Figure 2.9: Binary image after thresholding. The area under the water line is white.

As the water was only one object that had to be tracked, and with the rest of the background being of a uniform colour, only two classes were found in the histogram of each image. Hence a single valley was enough to determine the threshold. Otsu's method [33, 63], a global thresholding method which minimises the within-class variance of intensities, allowed for the gap between the dominant classes from the histogram in Figure 2.8 to be used to calculate one normalised threshold value within the range $[0, 1]$ in an iterative way for each image. After the water area had been determined, the next step was to translate the area into measurements. This was achieved by tracking the area occupied by water and determining the coordinates of the water depth, described in the following section.

2.3.4 Image analysis: turning images to data through coordinate acquisition

Image analysis refers to translating the traced objects from image-dependent units such as pixels to useful numerical data, expressed in universal units such as metres. This transformation came true by tracing the coordinates of the moving object in question and translating them to length and depth values. Hence, the more precise the processing and the analysis, the more accurate the obtained data are.

Each snapshot was read from top to bottom and from left to right. The free surface line in the binary images, was white with the background having a black colour. Depending on the selected detection method, the area under the free surface would be either black or white, as illustrated in Figures 2.9 (colour-channel method), A.3 (edge-detection method). For either one of the tracking methods, the free surface line coordinates $c_{i,j}$ were determined at the location where the white line was, *i.e.* at the transition point from black to white. These pixel coordinates $c_{i,j}$, depicting the free surface line were thus acquired and stored.

As aforementioned, the coordinates of the tracked area were in pixel units (length l_{px}, depth $\mathrm{depth}_{\mathrm{px}}$) and had to therefore be transformed to metric units. It was thus important to first calibrate the images, which entailed determining the metre-to-pixel ratio for snapshots with the same resolution, so as for the translation of the tracked area from pixels to metres to occur successfully. For the calibration to be successful, the snapshots were first segmented such that only the wave tank remained in each of them. The reason for this was because both the tank's measurements and the snapshot's resolution were known by construction and from the segmentation stage respectively. As a result calibrating the tank, enabled by determining the metre-to-pixel ratio from the measurements and the image resolution, also calibrated the water depth because the water was in the same frame of reference with the tank. Segmenting the images reduced their resolution, therefore an ‘updated’ calibration ratio was determined separately for the length and the height of the tank. After that, the length and height coordinates of the tracked water line were translated from pixels to metres.

The following formulas were hence used to perform this transformation for the length and the height/depth by using the calibration ratios for the length $\frac{\text{length}_\text{cell}}{\text{no}_\text{col}}$ and the height/depth $\frac{\text{depth}_\text{cell}}{\text{no}_\text{rows}}$, embedded in equation (2.1). During the transformation, the results were flipped so that the length and depth could be interpreted as being read from bottom to top. The transformation relationships are:

$$l_\text{m} = l_\text{px} \times \frac{\text{length}_\text{cell}}{\text{no}_\text{col}} \times 10^{-2}, \tag{2.1a}$$

$$h_\text{m} = \text{depth}_\text{cell} \times (1 - \frac{\text{depth}_\text{px}}{\text{no}_\text{rows}}) \times 10^{-2}, \tag{2.1b}$$

where l_m, h_m are the length and depth coordinates in metres, $\text{length}_\text{cell}$ and depth_cell are the dimensions of the cell enclosed in the snapshot in centimetres and $\text{no}_\text{rows} \times \text{no}_\text{col}$ is the total resolution of the snapshot. In the case that the snapshots had been uniformly segmented, $\text{length}_\text{cell}$, depth_cell as well as no_rows and no_col were still constant but adapted to the smaller resolution of the cropped snapshots.

Using equations (2.1a) and (2.1b) provided sets of coordinates $c_{i,j}$=$(\ell_i, max_i(h_j))$ measured in metres, for every image element c on the free surface with coordinates: $i = 1, 2, ..., no_\text{col}$ for the length (horizontal axis) and $j = 1, 2, ..., no_\text{rows}$ for the depth (vertical axis). It was, however, possible to have multiple depth coordinates for one length coordinate. To this end, $\max_i(h_j)$ was introduced; thus always selecting the largest depth value out of all to locate the free surface. Figure 2.10 illustrates how the tracking works.

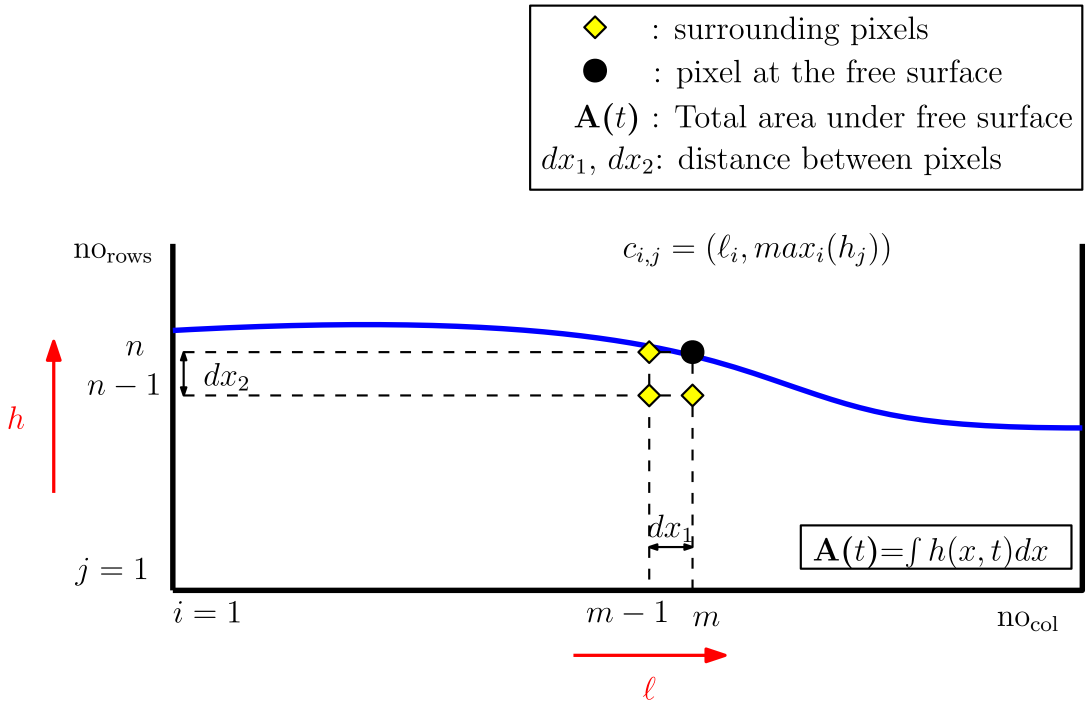
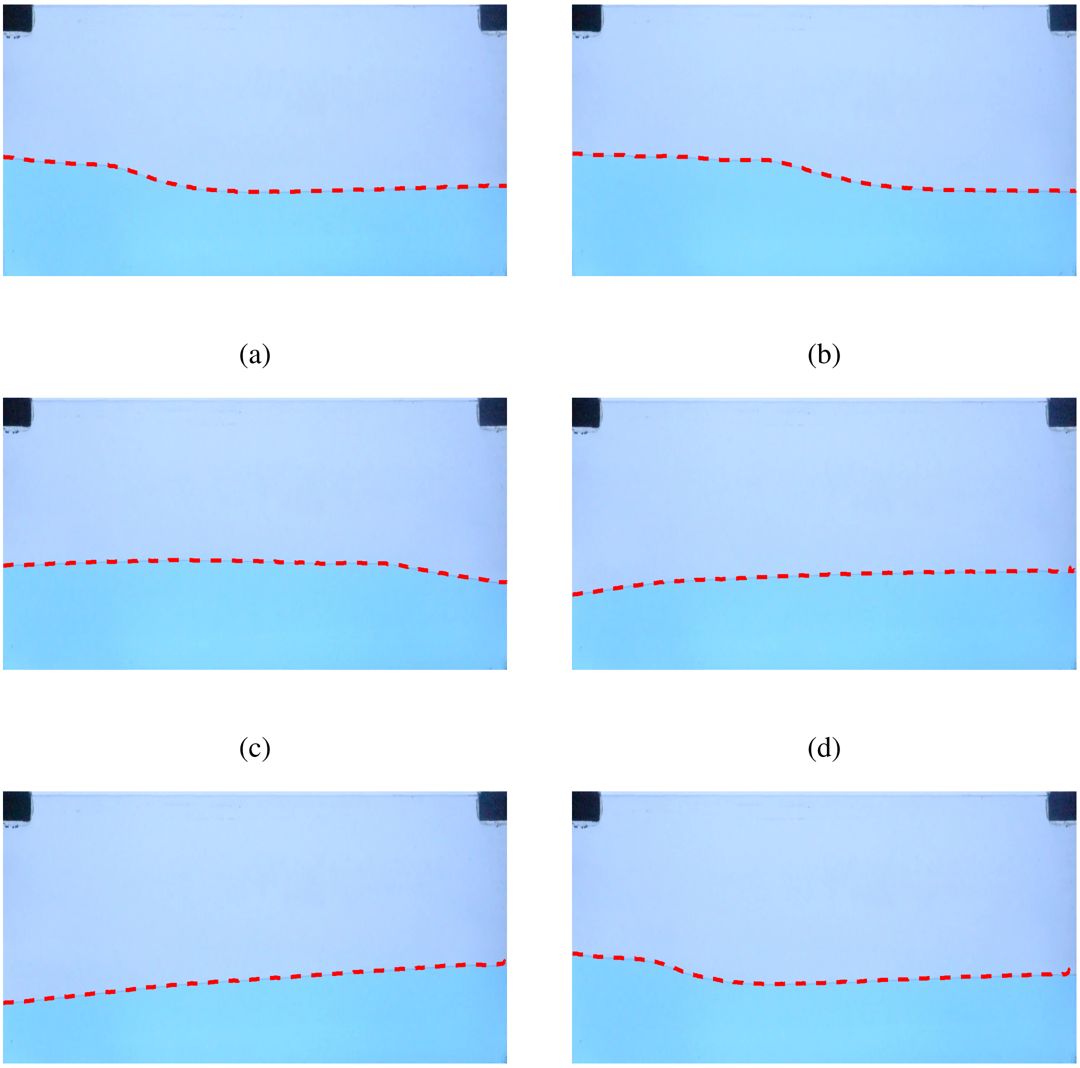


Figure 2.10: Coordinate acquisition of the free surface line. Pairs (l_i, h_j) of equal length l_i are used for the computation of the area under the free surface line, with the free surface coordinates given by $c_{i,j}$=$(\ell_i, max_i(h_j))$. In this graph $i = m$ and $j = n$ are the coordinate indices at the free surface.

Provided that $\ell_i, max_i(h_j)$ were turned into vectors of the same length, the area under the free surface with coordinates $c_{i,j}$ could be then computed for each snapshot. Having the depth coordinates in metres allowed for the computation of the area occupied by water and located under the free surface line for each moment in time:

$$A(t) = \int h(x,t)\ dx. \tag{2.2}$$

In general, for each case the output of the image-analysis algorithm were depth $h(x, t)$ and area $A(t)$ time series. In the following figures, area $A(t)$ and depth at the left wall $h(0, t)$ time series have accompanied snapshots of cases 149, 165 overlaid against the tracked depth. In this way the tracked water area is illustrated in an image as well as in a numerical form.

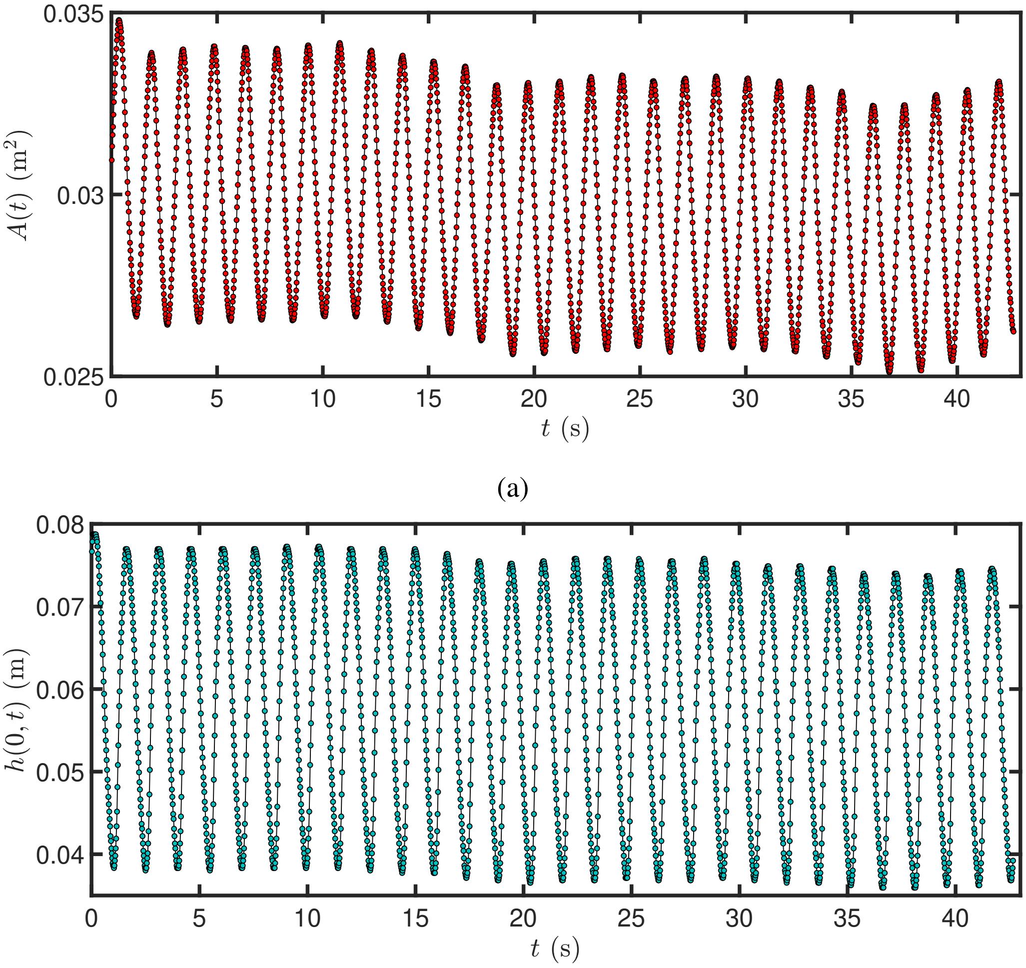


(a) (b)

(c) (d)

(e) (f)

Figure 2.11: Tracking the water line at medium depths for video 149 during times $t = 0.02\text{s}, 0.2\text{s}, 0.48\text{s}, 0.76\text{s}, 1.06\text{s}, 1.44\text{s}$. Wave period is $T = 1.42\text{s}$. The fully tracked video of the experiment can be found here: `https://www.youtube.com/watch?v=WXTVx-RDT5o`.

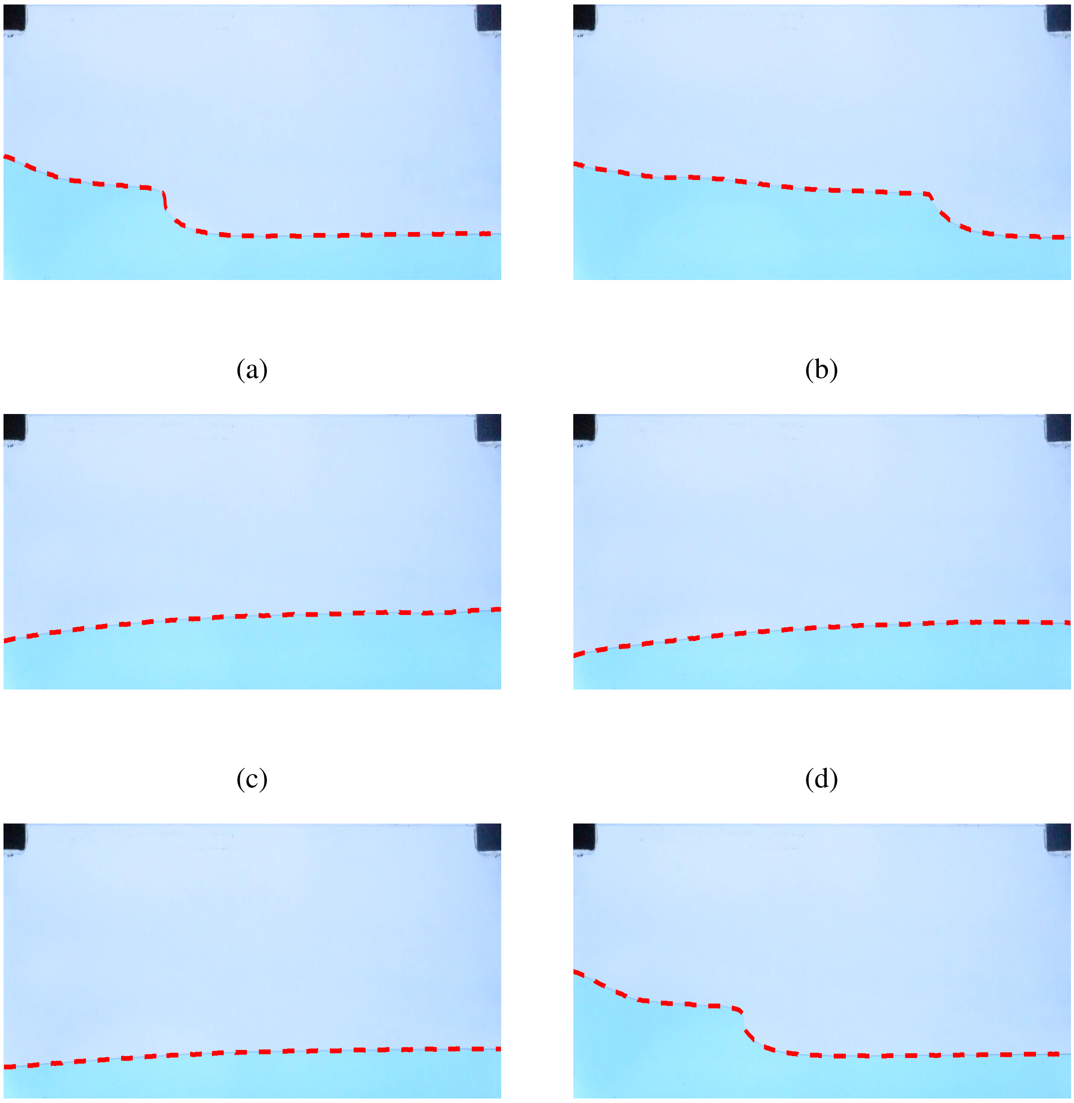


(a)

(b)

Figure 2.12: Raw data for the area (top panel) and the depth at the left wall (bottom panel). These data originated from video 149 (tank two).

It can be seen that the data in Figure 2.12 have a sinusoidal shape. The variations on the minimum and maximum area are attributed to Arduino. Even though Arduino was already in operation prior to the recording, the electrical current controlling the pumps was stabilised much later on with some variations on the water depth still visible over the course of the experiment. As a result, the minimum and maximum water depth were relatively stabilised during the second half of the recording of the experiment (20s-40s).

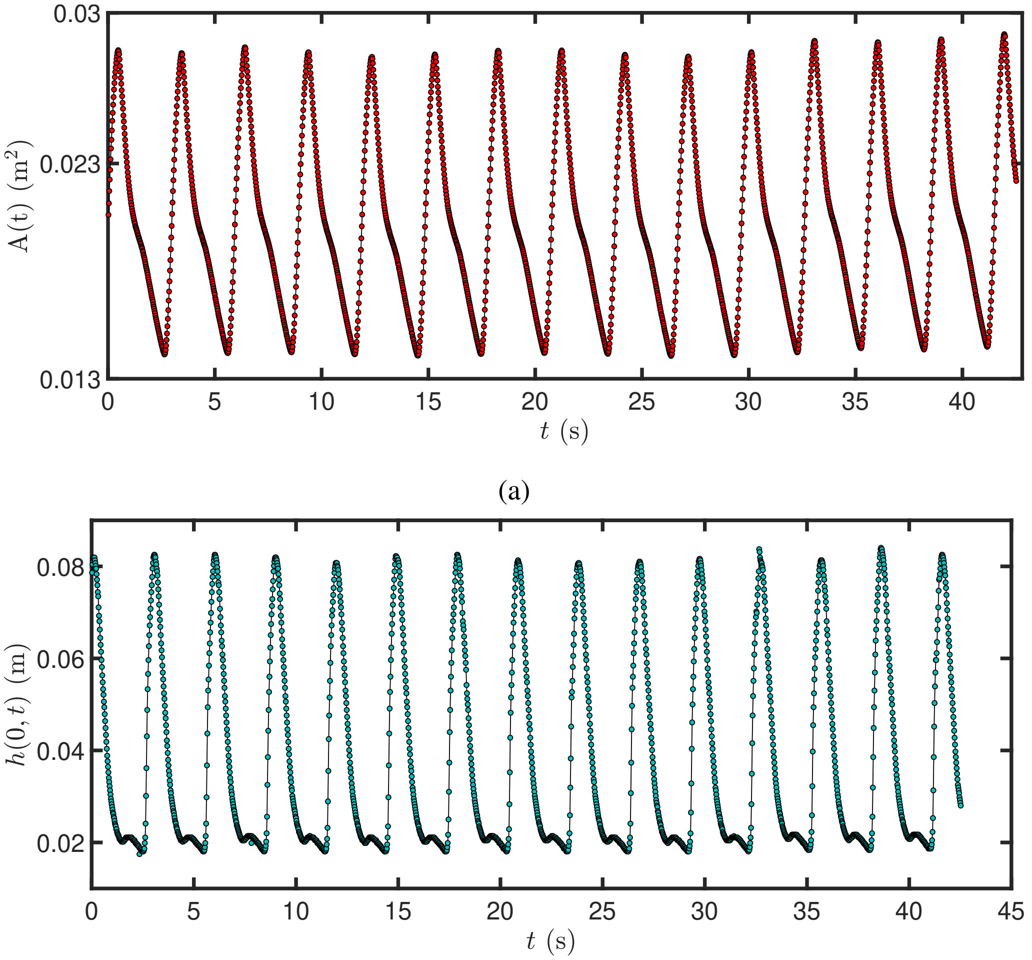


(a) (b)

(c) (d)

(e) (f)

Figure 2.13: Tracking the water line at shallow depths for video 165 at times $t = 0.02\text{s}, 0.32\text{s}, 0.48\text{s}, 0.92\text{s}, 1.68\text{s}, 2.6\text{s}, 3\text{s}$. Wave period is $T = 3\text{s}$. The fully tracked video of the experiment can be found here: `https://www.youtube.com/watch?v=fwtB55BSU4o`.



(a)

(b)

Figure 2.14: Raw data for the area (top panel) and the depth at the left wall (bottom panel). These data originated from video 165 (tank two).

It can be seen in Figures 2.11 and 2.13 that the detection of the free surface line has been computed accurately for various frequency and wave steepness values. It should be noted that the shape of the depth data at the left wall $h(0,t)$ of case 165 is different than of case 149. Thus for different Arduino settings, *i.e* frequency and water depth, no deviations in the minimum and maximum wave height were observed in the above data. Consequently the assumption that Arduino's operation influenced the output data, for those specific settings, was further enforced.

2.4 Conclusions

The colour-channel-detection method has been selected as the most suitable, precise and reliable tracking method for detecting the water line and translating it into useful measurements. It successfully tracked waves of various frequencies and steepness. The advantages of this method are that the main processing algorithm works iteratively, adapting to the threshold values of each snapshot, while it remains simple to use. In addition an option to choose the times of interest that the processing should occur has been provided, thus offering flexibility to the user while saving time from running the algorithm for all images. The main disadvantage is that the code has been developed in Matlab using the Image Processing Toolbox, which is not an open source software. For cases such as this, where a large number of snapshots has to be processed iteratively, OpenCV could alternatively be used by translating the Matlab commands to Python commands accordingly.

Chapter 3

Modelling water waves

In the quest for the models that capture the water dynamics in the Hele-Shaw cell the best, two one-dimensional nonlinear shallow-water models were selected and have been presented in this chapter. A shallow-water and a potential-flow shallow-water model[1], later validated against experiments in Chapter 4. Both (shallow-water) models have been extended to include terms responsible for momentum damping and forcing at the left boundary, effects directly related to the narrow width of the tank and the operation of the wave pump.

The structure of the chapter is as follows: the developed mathematical models are presented in Section 3.1 and the numerical discretisation of the models ensues in Section 3.2. Lastly, the conclusions follow in Section 3.3.

3.1 Mathematical models

Two one-dimensional shallow-water models, one using the velocity u and one using the velocity potential ϕ (such that $u = \nabla\phi$), have been used to model the hydrodynamics in the vertical Hele-Shaw cell. Taking into consideration the narrow width of the tank, any contributions to the velocity

---

[1] initially developed and partly validated in [42]

profile from the lateral dimension can be encapsulated by width-averaging the three-dimensional Navier-Stokes equations.

The shallow-water equations comply with the shallow-water approximation; there is no wave dispersion and the wavelength must be roughly twenty times larger than the water depth, following the long-wave approximation. As a result, any variations of the velocity profile outside the boundary layer are very small and can be neglected by depth-averaging. Even though the long-wave approximation is only satisfied to leading order (in a suitable aspect ratio) in our experiment, the shallow-water approximation forms a useful first step towards approaching the modelling. In [90] the one-dimensional shallow-water model was recovered by width- and depth- averaging the three-dimensional Navier-Stokes equations and by omitting the rest of the terms. However, there is evidence from the results found in Chapter 4 that future work can benefit from the inclusion of the dispersible terms omitted in shallow-water theory. The potential-flow shallow-water equations have been derived from Luke's (extended) variational principle, with the dynamics considered only at the free surface.

Experimentally, higher frequencies combined with shallower depths or the presence of a bottom topography, are the factors responsible for creating steeper waves or even wave-breaking, with the latter investigated in Chapter 5. Wave breaking has thus been accounted for in the shallow-water model by using the theory of hydraulic jumps and bores [101] and is approximated by a local line or point discontinuity, which violates the long-wave approximation. Even though this discontinuity violates the long-wave assumption, the hydraulic bore model has proven to be a very powerful to model wave breaking. This is because after wave-breaking wave motions become quasi-steady they become analogous to a spilling breaker or a bore, if only the macroscopic-scale effects are of interest [12, 69]. In the case of the potential-flow approximation, wave breaking violates the irrotational assumption and thus cannot be handled without extra parameterisations.

3.1.1 Shallow water model

An incompressible fluid has been considered in a one-dimensional domain $x \in [0, L]$ with water depth $h = h(x,t)$ and velocity $u = u(x,t)$ as functions of space and time, incoming area flux

$Q(t)$ at $x = 0$ and a hard wall at $x = L$. The fluid is bounded from below by a bottom topography, which is flat in this instance, and from above by a free surface. Dispersion is absent and the dynamics have been reduced to the free surface only by providing solutions for the depth at the free surface and the velocity.

Momentum damping $-\gamma u$ is present as a result of the close proximity of the glass plates, with a damping factor $\gamma = \frac{3\nu}{l^2}$ defined as the ratio of the kinematic viscosity ν and the tank's half-width $l = 0.001\mathrm{m}$. The momentum damping term was originally derived in [90], as a result of width- and depth-averaging the three-dimensional Navier-Stokes equations. As the water viscosity is small ($\nu = 10^{-6}\,\mathrm{m}^2/\mathrm{s}$) its effect on the fluid's velocity in the lateral direction is confined to a thin boundary layer between the fluid and the boundary walls (effectively the gap width), where it is dominant. The effect of viscosity on the flow's momentum was encapsulated by width-averaging the Navier-Stokes equations, thus revealing the momentum damping term.

The gap width for tank two was not chosen randomly. In [90] simulations of the system (3.1) were carried out, prior to the design of the tanks, with the aim to investigate the dissipative effect of various gaps thickness on the momentum of generated bores. It was found that for a frequency range of 0.5-1Hz and gap widths $2l < 1.5\mathrm{mm}$ the fluid's momentum dissipated too quickly and the bores never reached the end, for a chosen tank length of 0.7-1m. On the contrary, above the width threshold of 1.5mm, the generated bores travelled across the tank. Consequently, the tank was built with a gap width equal to $2l = 2\mathrm{mm}$ (see Chapter 2) with aluminium oxide particles of a diameter $d = 1.75 \pm 0.05\mathrm{mm}$ used to represent a gravel beach in water-beach experiments (see Chapter 5).

In the simulations of the experiment the momentum damping factor γ was determined from the half-width l of the wave tank, which was fixed by construction, and was thus constant for each wave tank. In addition, as water was the only fluid used during the experiments meant that the kinematic viscosity remained constant and was equal to $\nu = 10^{-6}\,\mathrm{m}^2/\mathrm{s}$ for all simulations.

Given that ∂_x and ∂_t represent the partial derivatives with respect to space and time and g is the acceleration due to the effect of gravity, the one-dimensional nonlinear shallow-water equations read:

$$\partial_t h + \partial_x (hu) = 0, \tag{3.1a}$$

$$\partial_t u + u\partial_x u + g\partial_x h = -\gamma u, \tag{3.1b}$$

with an inflow boundary condition at the left wall due to the pump action resulting in an in- or outflow of water:

$$Q(t) = hu|_{x=0}. \tag{3.1c}$$

Throughout the simulations in tank two, the momentum dissipation factor was $\gamma = 3\mathrm{s}^{-1}$, with $\nu = 10^{-6}\,\mathrm{m}^2/\mathrm{s}$ and $2l = 0.002\mathrm{m}$. For tank one, it was equal to $\gamma = 2.479\mathrm{s}^{-1}$, with the kinematic viscosity same as before and the gap width equal to $2l = 0.0022\mathrm{m}$.

3.1.1.1 Non-dimensionalisation

Provided that L_x, H_0 are the horizontal and vertical length scales and the non-dimensional variables are denoted by the superscript stars, the scales used to non-dimensionalise the equations are: $x = L_x x^*$, $h = H_0 h^*$, $t = \frac{L_x}{U_0} t^*$, $Q = H_0\sqrt{gH_0}Q^*$, $u = U_0 u^*$, with $U_0^2 = gH_0$ resulting in $g^* = 1$. Substituting the scales, dropping the stars, rearranging the equations and taking $U_0 = \sqrt{gH_0}$, equations (3.1) become:

$$\partial_t h + \partial_x (hu) = 0, \tag{3.2a}$$

$$\partial_t (hu) + \partial_x (hu^2 + \frac{h^2}{2}) = -C_d hu \tag{3.2b}$$

$$Q = hu|_{x=0}, \tag{3.2c}$$

$$0 = u|_{x=L}, \tag{3.2d}$$

with scaled damping $C_d = \frac{\gamma L_x}{U_0}$. Details on the discretisation of equations (3.2) can be found in Section 3.2.1.

3.1.2 Potential-flow shallow-water model

The potential-flow shallow-water model was originally developed in [42] where it was partly validated against obtained experimental data. Following this line of work the model was fully validated against new and improved experimental data, with further details on the validation process found in Chapter 4 of this thesis. The model is therefore presented in this section for closure and, similarly to the shallow-water model, describes the water dynamics in the Hele-Shaw cell while enhanced with the effects of momentum dissipation and inflow of water.

The variational principle offers the advantage that the whole system is described by a single functional. The dimensionless equations for the potential-flow shallow-water model can be thus recovered by determining a variational principle for the system. The variational principle also leads to energy conservation through Hamiltonian dynamics. In particular, Miles' variational principle [58] uses the Hamiltonian dynamics and is dynamically equivalent to Luke's variational principle, with the latter presented below. Luke's (dimensional) variational principle [51], visualised in equation (3.3) for the two-dimensional case assuming irrotational flow, has been used as a building block for the variational principle of the mathematical system:

$$0 = \delta \int_0^T \int_0^L \int_0^{h(x,t)} -\rho(\phi_t + \frac{1}{2}(\nabla\phi)^2 + g(z - h_0)) \; dz\, dx\, dt, \tag{3.3}$$

with ρ the density, $0 \leq z \leq h(x,t)$ the depth between the bottom and the free surface $h(x,t)$ and ϕ the velocity potential such that $u = \partial_x \phi$. The lateral direction has been ignored, following the quasi two-dimensional nature of this vertical Hele-Shaw set-up, which is why the free surface does not depend on the lateral direction y either. The velocity potential is two–dimensional in space $\phi = \phi(x,z,t)$ with h_0 the depth of the water at rest. Equation (3.3) is essentially equal to Bernoulli's principle for an unsteady and irrotational flow. Due to the incompressibility assumption, the density ρ is constant and has thus been simplified from the following equations.

The variational operator (δ) is applied on the action functional $S = \int_0^T \mathscr{L}[h,\phi] \; dt$, with $\mathscr{L}$ the Lagrangian (difference between potential and kinetic energy [51]), and is defined as follows [52]:

$$0 = \delta S = \delta \int_0^T \mathscr{L}[h,\phi] \; dt \equiv \lim_{\epsilon \to 0} \int_0^T \frac{\mathscr{L}[h + \epsilon\delta h, \phi + \epsilon\delta\phi] - \mathscr{L}[h,\phi]}{\epsilon} \; dt. \tag{3.4}$$

The first variation δS, *i.e.* the change in the action functional S, is zero if its arguments functions h, ϕ change by an infinitesimal amount.

In the case of the potential-flow shallow-water model, the dynamics only at the free surface are considered hence resulting in $\phi = \phi(x, z = h, t)$. Consequently, the velocity potential becomes one-dimensional in space and thus depends only on the horizontal dimension and time ($\phi = \phi(x,t)$). With ϕ defined at the free surface, the integral with respect to z can be computed for equation (3.3) and integrating the first term by parts for t yields:

$$0 = \delta \int_0^T \int_0^L \phi \partial_t h - \frac{1}{2} h (\partial_x \phi)^2 - \frac{g}{2}(h - h_0)^2 dx \ dt. \tag{3.5}$$

The effect of the momentum damping due to the close proximity of the glass plates has to be included through the term $e^{\tilde{\gamma} t}$ following formulations for damped oscillators [64], with $\tilde{\gamma}$ the nondimensional momentum damping factor equal to C_d from system (3.2). In addition the existence of forcing at the left wall, a direct consequence of the operation of the pumps, has to be accounted for at the left boundary with the inclusion of the term $Q(t)\phi(0,t)$.

Non-dimensionalising equation (3.5) with the same scaling as in Section 3.1.1.1 was possible, with the velocity scale transformed accordingly for the velocity potential: $\phi = L_x \sqrt{g H_0} \phi^*$. Adding the influx term and multiplying by $e^{\tilde{\gamma} t}$ results in the relevant variational principle for the damped shallow-water flow in the Hele-Shaw cell:

$$0 = \delta \int_0^T \int_0^L [\phi \partial_t h - \frac{1}{2} h (\partial_x \phi)^2 - \frac{1}{2}(h - h_0)^2 dx \ - Q(t)\phi(0,t)] e^{\tilde{\gamma} t} dt. \tag{3.6}$$

Integrating by parts for the first term, for reasons explained later on (cf. Section 3.2.2.2), yields:

$$\begin{aligned} 0 = \delta \int_0^T \int_0^L & h \partial_t (\phi \mathrm{e}^{\tilde{\gamma} t}) + \left(\frac{1}{2} h (\partial_x \phi)^2 + \frac{1}{2}(h - h_0)^2 \right) \mathrm{e}^{\tilde{\gamma} t} \ dx \\ & + Q(t) \phi(0,t) \mathrm{e}^{\tilde{\gamma} t} \ dt. \end{aligned} \tag{3.7}$$

Taking the variation of (3.7), in the same way as in equation (3.4) gives:

$$\begin{aligned} 0 = \int_0^T \int_0^L & [\delta h \partial_t (\phi e^{\tilde{\gamma} t}) + \underline{h \partial_t (\delta \phi e^{\tilde{\gamma} t})} + \frac{1}{2} \delta h (\partial_x \phi)^2 e^{\tilde{\gamma} t} + \underline{h (\partial_x \phi)(\partial_x \delta \phi) e^{\tilde{\gamma} t}} \\ & + \delta h (h - h_0) e^{\tilde{\gamma} t}] dx + Q(t) \delta \phi(0,t) \mathrm{e}^{\tilde{\gamma} t} \ dt. \end{aligned} \tag{3.8}$$

The second and fourth terms in equation (3.8) are integrated by parts in time and space, respectively. Then, some of the terms that arise are eliminated by considering a solid wall boundary condition at the right wall $\partial_x\phi|_{x=L} = 0$ and applying suitable end point conditions $\delta\phi = 0$ and $\delta h = 0$ at $t = 0, T$. Then equation (3.8) becomes:

$$\begin{aligned}0 = \int_0^T \int_0^L \Big(\big(\partial_t(\phi e^{\tilde{\gamma}t}) + \frac{1}{2}(\partial_x\phi)^2\, e^{\tilde{\gamma}t} + (h - h_0)e^{\tilde{\gamma}t}\big)\delta h - \big(\partial_t h + \partial_x(h\partial_x\phi)\big) \\ e^{\tilde{\gamma}t}\delta\phi\Big)\, dx + \big(Q(t) - h\partial_x\phi\big)\big|_{x=0}\, \delta\phi(0,t)e^{\tilde{\gamma}t}\ dt. \qquad (3.9)\end{aligned}$$

It can be seen that the influx term is balanced by the left boundary term ($h\partial_x\phi|_{x=0}$) acquired from the integration by parts. Since the variations $\delta\phi$ and δh are arbitrary, the potential-flow shallow-water equations are obtained:

$$\begin{aligned}\delta h:&\quad \partial_t(\phi e^{\tilde{\gamma}t}) + \frac{1}{2}(\partial_x\phi)^2\, e^{\tilde{\gamma}t} + (h - h_0)\, e^{\tilde{\gamma}t} = 0 \\ &\quad \Leftrightarrow \partial_t\phi + \tilde{\gamma}\phi + \frac{1}{2}(\partial_x\phi)^2 + (h - h_0) = 0, && (3.10a)\\ \delta\phi:&\quad \partial_t h + \partial_x(h\partial_x\phi) = 0, && (3.10b)\\ \delta\phi(0,t):&\quad Q(t) = h\partial_x\phi \qquad \text{at} \qquad x = 0, && (3.10c)\\ \delta\phi(L,t):&\quad 0 = h\partial_x\phi \qquad \text{at} \qquad x = L. && (3.10d)\end{aligned}$$

Equations (3.10b), (3.10c), (3.10d) can be directly obtained from (3.2a), (3.2c), (3.2d) by setting $u = \partial_x\phi$ everywhere. Similarly, substituting (3.2a) in equation (3.2b) leads to the same result as differentiating (3.10a) by x and multiplying it by h, while setting the scaled momentum damping $C_d = \tilde{\gamma}$ in (3.2b). The scaled acceleration due to gravity remains $g^* = 1$.

3.2 Numerical discretisation

Numerical methods are used so as to compute approximations of the solutions to the sets of differential equations of physical variables. This is accomplished by replacing the continuous system, that is described from the partial differential equations in each model, with a set of discrete and finite values for space and time; a process also known as discretisation process. Each numerical method then approximates the solution of the discretised equations at those allocated times and locations, based on a specific strategy determined for each numerical method.

The shallow-water and the potential-flow shallow-water models have been discretised in space using either the finite-volume method or the finite-element method and in time using either the Forward Euler or the Störmer–Verlet method. This section is going to provide some background on how these methods work.

3.2.1 Finite-volume method

The $1D$ shallow-water equations have been discretised in space and time using the finite-volume and Forward-Euler method, respectively. The finite-volume method used is a first-order method and has been chosen due to its advantage to handle discontinuities arising from hydraulic bores. The shallow-water equations (system (3.2)) are a system of conservation laws which can be rewritten as follows:

$$\partial_t U + \partial_x F(U) = S, \tag{3.11}$$

where $U = (h, hu)$ and the flux vector $F(U) = (hu, hu^2 + g\frac{h^2}{2})$ is a function of the variables involved. In addition, the vector $S = (0, -C_d hu)$ is the source term, here including the damping term. For non-flat topographies, the (dimensionless) topography term is also located in the source term ($S = (0, -C_d hu - h\partial_x b)$).

The shallow-water equations belong in the family of hyperbolic problems, whose Jacobian (J) has real and distinct eigenvalues. Turning equation (3.11) to its quasi-linear form in equation (3.12), in which we set $S = 0$ for simplicity, reads:

$$\partial_t U + J\partial_x U = 0, \text{ with J} = \begin{bmatrix} 0 & 1 \\ -u^2 + gh & 2u \end{bmatrix}. \tag{3.12}$$

Following from [47], the eigenspeeds of the Jacobian J are: $\lambda\pm = u \pm \sqrt{gh}$. An estimate of the time step required to ensure numerical stability of the FV method can be made by considering the eigenvalues, characteristics and grid size. A time step criterion, also known as Courant, Friedrichs and Lewy (CFL) condition [21], can yield an estimate provided a finite time step Δt and a grid size Δ_k: $\Delta t \leq \text{cfl} \frac{min(\Delta_k)}{\max(\lambda_-, \lambda_+)}$. In this formula for Δt a constant value for the cfl has to be selected. Theory suggests that for the Godunov method a CFL value satisfying $0.5 < \mathrm{CFL} < 1$ is appropriate [54]. Our numerical simulations suggested a value of 0.6, which was suitable and was adopted in all simulations that used this method and are found in Chapter 4.

Going back to the vectorial conservation laws found in equation (3.11) with U, $F(U)$ same as before, one can discretise the one-dimensional spatial domain x of length $[0, L]$ into N intervals/volumes and determine n number of times $[0, n]$ at which the solutions U are to be computed. The Godunov method [46, 93] has been selected in order to compute these solutions. The basis of the Godunov method, after a space-time mesh has been introduced, is to integrate the conservation law over one space-time cell $x_{k-1/2} < x < x_{k+1/2}$ and $t^n < t < t^{n+1}$. The vector of cell-averaged quantities or means $U_1, U_2, ..., U_N$, visualised in Figure 3.1 and separated by a one step discontinuity, comprises the unknowns of the problem that are determined by the numerical method. The cell-averaged quantities are updated in time with the help of the flux, evaluated at the intercell edges also known as faces/nodes. After the resulting flux at the nodes is replaced by a numerical flux, this immediately yields the FV scheme. Defining the numerical flux, which will allow for the solution to be updated in time, is key.

A Riemann problem [46, 92] is an initial value problem and in the field of hyperbolic problems consists of the equation of the conservation law and an initial state for the system expressed by piecewise constant data with a single discontinuity (see Figure 3.2). The Godunov method uses the solution of each "local" Riemann problem in order to determine the flux at the interface at the discontinuity, using the solution/states U of the surrounding volumes of the system. Here the Harten, Lax and van Leer (HLL) flux has been used, also known as the approximate HLL Riemann-type solver [47, 92], and has been further described in Appendix B.2.

Provided that subscripts and superscripts denote the space and time discretisation, equation (3.13) reads:

$$U_k^{n+1} = U_k^n - \frac{\Delta t}{|\Delta x_k|}(F_{k+1/2} - F_{k-1/2}) + \Delta t \, S_k^n, \tag{3.13}$$

U_k^n is the space- and time- cell average of the solution for volume k at time n with source term $S_k^n = (0, -C_d h_k^n u_k^n)$, grid size Δx_k and the HLL flux F which depends only on U_k and U_{k+1}. The origin of equation (3.13) has been further described in Appendix B.1. In the case that the

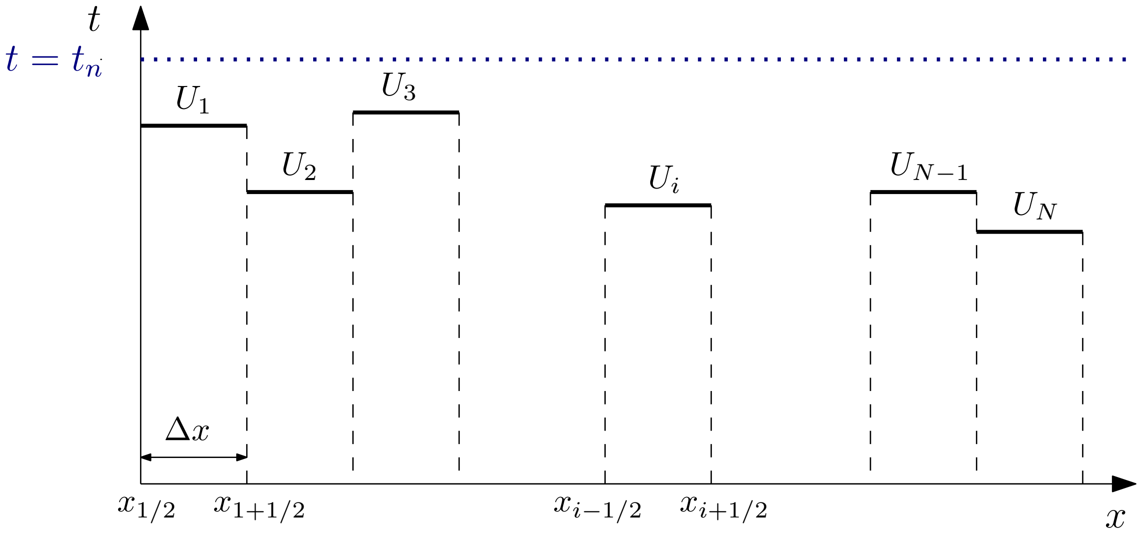


Figure 3.1: The computational domain has been divided into $k = 1, 2, ..., N$ volumes each of length Δx. At each volume, a piecewise constant value for the solution U_k is assigned at the centre of the cell, contained within edges $[x_{k-1/2}, x_{k+1/2}]$. Pairs of states are separated by a discontinuity at the edges. The vector of all these piecewise constant solutions forms the numerical solution of the problem at time t_n.

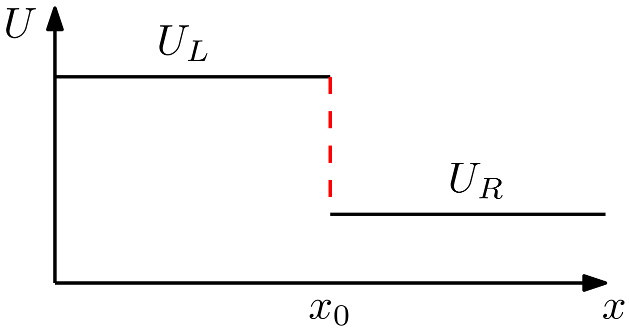


Figure 3.2: Riemann problem. Piecewise constant states U_L, U_R separated by a single discontinuity (red dash).

bottom topography is not flat, the topography term is implemented in the numerical shallow-water model with the help of well-balanced schemes [3].

In the following section, the boundary conditions as well as their implementation are presented. It should be noted that the implementation of the boundary conditions was crucial; the latter were the entrance through which the experimental data entered the model, which ultimately led to its validation in Chapter 4.

3.2.1.1 Boundary conditions and ghost cells

For the shallow-water equations (system (3.2)), a hard-wall condition was imposed on the right wall $u_n|_{x=L} = u|_{x=L} \cdot \hat{n} = 0$, with $\hat{n}$ the normal unit vector and $u|_{x=L} = 0$, such that no mass of water goes through the solid impermeable wall and the fluid is reflected when it reaches the right wall. On the left wall (at x=0), the measured mass flux and depth were imposed to the model through an in- and out-flow boundary condition (equation (3.2c)). The boundary conditions have been sketched in Figure 3.3.

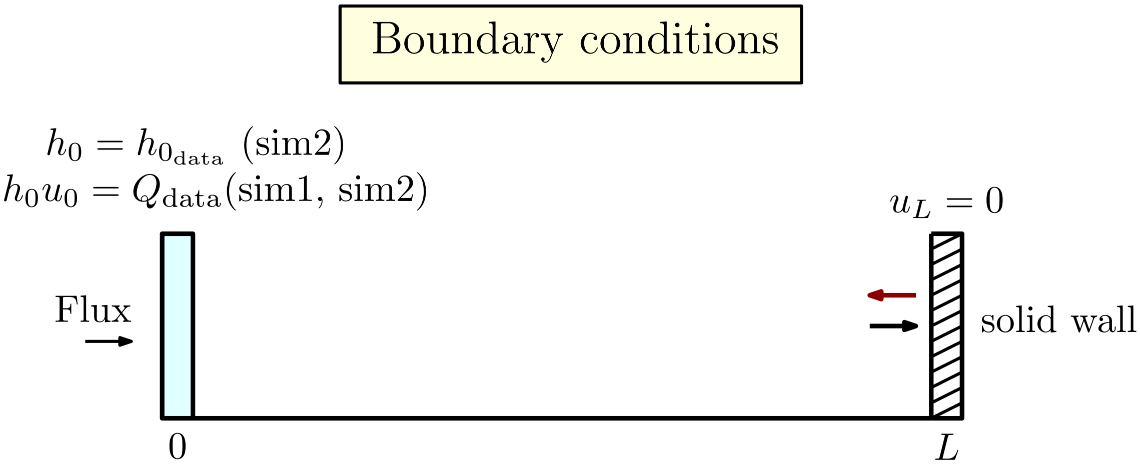


Figure 3.3: Boundary conditions for the Hele-Shaw cell. Inflow and outflow (flux) boundary condition imposed on the left wall to account for the existence of the pump. Solid wall assigned on the right wall, responsible for the water reflection (black, red arrows). The implemented form of the boundary conditions is found in Section 4.2, and is closely linked to the validation of the model through simulation one (sim1) and simulation two (sim2). In sim1 only the inflow of water was prescribed through a function, which was superimposed on the data, Q_{fitted}. In sim2, both the depth at the left wall and the inflow of water were 'directly' prescribed from the data $h0_{\text{final}}$, Q_{final} and by employing linear interpolation.

In the finite-volume method, fluxes determine the communication between cells. Similarly, using the ghost cells boundary treatment [47], [93] sketched in Figure 3.4, the boundary conditions were not imposed directly but indirectly as values that entered the numerical flux at $x = 0$ and $x = L$, together with the information on the left of x=0 (ghost cell) and the right of x=L (ghost cell) within the domain. The addition of the ghost cells thus required the extension of the computational domain so as to accommodate for these two extra cells. Hence the domain was a union of the true interior volumes, where the solution was calculated, and the ghost cells that were only used for the computation of the fluxes at the boundaries. The HLL flux, using approximations of the characteristics of the hyperbolic system, then determined the extent at which the boundary information was used.

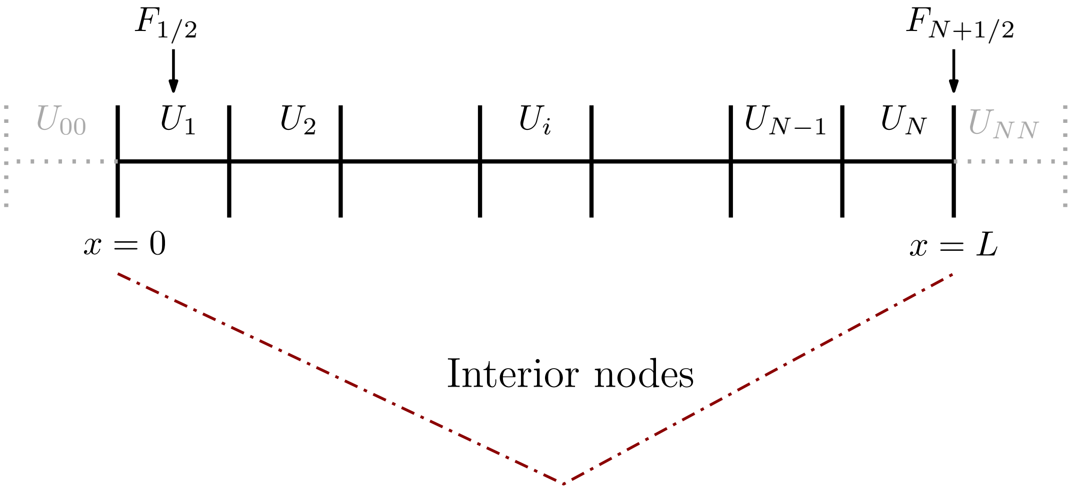


Figure 3.4: Addition of the ghost cells U_{00}, U_{NN} (grey) to the main computational domain consisting of cells $U_1, U_2, .., U_N$ enclosed at $x = 0, L$. Intercell fluxes $F_{1/2}$ ($x = 0$), $F_{N+1/2}$ ($x = L$) lead to solutions that satisfy the boundary conditions.

Provided an initial condition at $t = t_n$ for U_k one can generalise the process for all times, as follows from equation (3.13) and is illustrated in Figure 3.5. Calculating the intercell fluxes by using the previous time information of the state U_k^n, allows for the computation of an updated solution U_k^{n+1}.

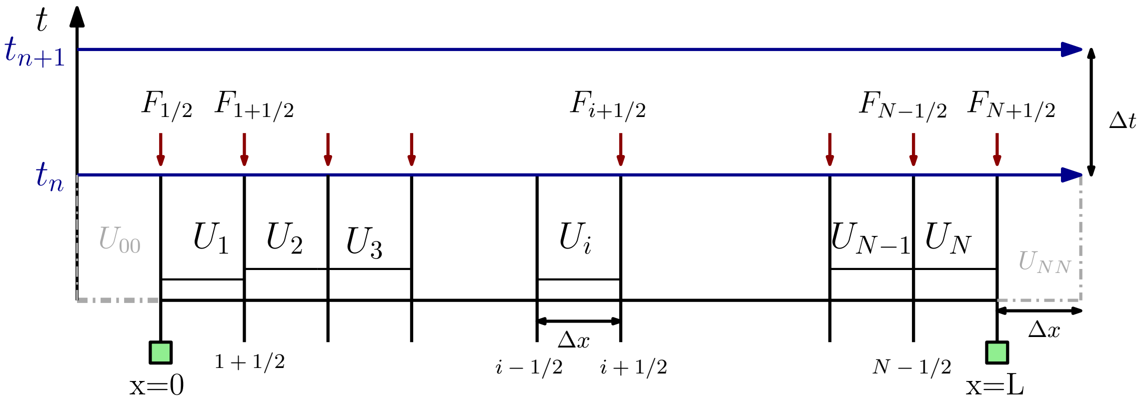


Figure 3.5: The computational domain has been divided into N cells of length Δx_k, and is bounded by the green squares. The ghost cells, U_{00}, U_{NN} have the same length Δx and have been added to the left and right of the computational domain to facilitate the intercell flux computation at the boundaries $F_{1/2}$, $F_{N+1/2}$ at time t_n. The updated solution can be then computed at time $t = t_n + \Delta t$ through equation (3.13).

3.2.2 Finite-element method

The potential-flow shallow-water equations were implemented in Firedrake; an automated system that uses the finite-element method to solve the equations spatially [79]. A summary of the implementation process can be found in Figure 3.8, after the explanation of the steps has been completed. The finite-element method (FE) works similarly to the finite-volume method, in the sense that it aims to approximate the solution of a system of partial differential equations by breaking down the domain into a discrete and finite number of sub-regions/elements and computing approximations of the solutions on each of these intervals. Depending on the geometry and the number of spatial dimensions required to simulate the problem at hand, the elements can be lines (1D), triangles or quadrilaterals (2D) and tetrahedra or hexaedra (3D) [78]. FE can deal with complex geometries in the fields of engineering and fluid dynamics, by allowing for its elements to be adapted to the geometry of interest, and that is its great advantage. For example, when dealing with a curved geometry one can use curved elements during the discretisation; depending on the dimensions of the problem it could be a curved line element, or a triangle with curved sides or a hexahedron with curved sides.

Apart from the selection of the type of element to discretise the mesh, functions are required to approximate the variables on the mesh. The function can be a polynomial of any degree with different continuity requirements; thus not only facilitating the integration and differentiation procedures but also providing improved accuracy of the numerical solution as the polynomial degree increases. At this point it is useful to point out that if a zero-degree polynomial was to be selected for each element, the approximate solution would be computed by a piecewise constant function, similarly to the finite-volume method.

The first step is to bring the system of equations into a weak form. Each equation is multiplied by an arbitrary, yet suitable, test function. A suitable function space must be defined for these test functions, which depends on the differential equation at hand. Then, integrating by parts and applying appropriate boundary conditions reduces the spatial derivatives and leads to the weak form. In Section 3.1.2, Luke's variational principle was modified and the potential-flow for shallow-water equations were acquired. As the variational principle has already brought the system of equations in an integral form, the calculation of the weak form for the implementation stage is not necessary. Instead, starting directly from the variational principle in equation (3.7), discretising it using an appropriate mesh, basis functions and first-order polynomials to approximate the solutions, taking the variations, integrating by parts and applying Dirichlet boundary conditions at the two walls results in the implemented equations.

3.2.2.1 Mesh, basis functions and approximate solutions

The one-dimensional potential-flow equations originating from equation (3.7) have been discretised by using a one–dimensional mesh consisting of line elements. As depicted in Figure 3.6, the domain has been partitioned into $N+1$– nodes such that each line element is located between each set of nodes. Hence for nodes $k = 1, 2, ..., N+1$ each line element can be defined as $E_k = \{x | x \in (x_k, x_{k+1})\}$. The union of these line elements results in a tessellation, with the line elements not overlapping with each other.

Approximations to the solutions h, ϕ of equation (3.7) are then computed at each line element. The variables are approximated using Ritz-Galerkin finite-element expansion with C^0-continuity

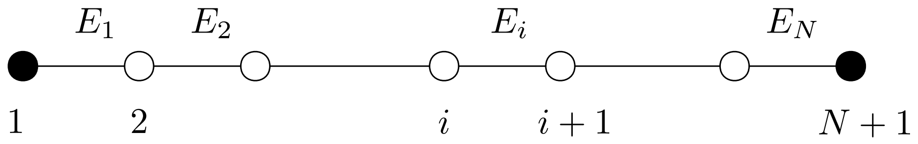


Figure 3.6: One–dimensional computational domain for the finite-element method. The domain has been partitioned into N elements E_k with $k = 1, 2, ..., N + 1$ nodes. Within each pair of nodes, a line element is enclosed.

as determined in equation (3.14). The basis functions $\varphi_k(x)$, $\varphi_l(x)$ determine the order of the polynomial approximating the solution and affect the quality of the approximations. In this work, one–dimensional continuous linear Lagrange polynomials [2] have been used in the following approximations:

$$h(x,t) \approx h_h(x,t) = \sum_{k=1}^{N+1} h_k(t)\varphi_k(x), \qquad \phi(x,t) \approx \phi_h(x,t) = \sum_{l=1}^{N+1} \phi_l(t)\varphi_l(x), \tag{3.14}$$

where u_h, h_h are the approximations to the solutions, with respective coefficients $h_k(t)$ and $\phi_l(t)$. The basis functions have compact nodal support, being unity at the centre node and zero at neighbouring nodes. For example, the basis function at the first element, also visualised in Figure 3.7, will be:

$$\varphi_1(x) = \begin{cases} 1 & \text{if } k = 1 \\ 0 & \text{if } k \neq 1 \end{cases}.$$

Substituting the approximations in equation (3.7) and taking the variations results in the discretised form of equation (3.8):

$$\begin{aligned} 0 = \int_0^T \int_0^L & [\delta h_h \partial_t(\phi_h e^{\gamma t}) + \underline{h_h \partial_t(\delta\phi_h e^{\gamma t})} + \frac{1}{2}\delta h_h (\partial_x \phi_h)^2 e^{\gamma t} + h_h(\partial_x\phi_h)(\partial_x \delta\phi_h)e^{\gamma t} \\ & + \delta h_h (h_h - H_0)e^{\gamma t}]dx + Q(t)\delta\phi_h(0,t)\mathrm{e}^{\gamma t}\ dt. \end{aligned} \tag{3.15}$$

In view of the equations being solved in Firedrake, equation (3.15) was modified by integrating the second term by parts with respect to time, using end-point condition and leaving the fourth

---

[2] $L_k(x) = \Pi_{m=1}^{n+1} \frac{x-x_k}{x_k-x_m}$, with k the number of nodes in the domain and n the order of the polynomial used to approximate the numerical solution.

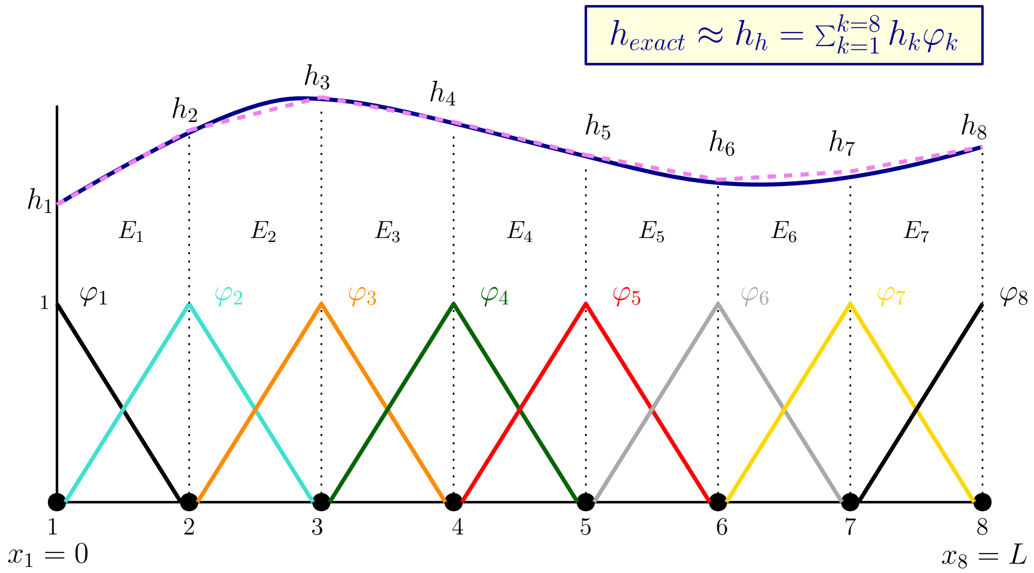


Figure 3.7: Approximation of the exact solution h_{exact} (blue) with h_h (magenta dash). The computational domain has been divided into seven line elements and eight nodes (circles). The approximate solution is computed at each line element with the help of basis functions consisting of continuous linear Lagrange polynomials φ_k, with $k = 1, 2, ..., 8$. Compact support has been enforced at each of the nodes.

term as it was. Recall that the equations were acquired by integrating the second and fourth term of equation (3.8) by parts. Rearranging equation (3.15) yielded:

$$\begin{aligned} 0 = \int_0^T \int_0^L \delta h_h \big[\partial_t(\phi_h e^{\gamma t}) + \frac{1}{2}(\partial_x \phi_h)^2 e^{\gamma t} + (h_h - H_0) e^{\gamma t}\big] \\ - \big[\partial_t h_h \delta\phi_h - h_h(\partial_x \phi_h)(\partial_x \delta\phi_h)\big] e^{\gamma t}\, dx + Q(t)\delta\phi_h(0,t)\mathrm{e}^{\gamma t}\ dt. \end{aligned} \tag{3.16}$$

In usual software such as Matlab, FE discretisation would occur as follows. The approximations of the solutions would be used in equation (3.7) and would be replaced with their expansion. This replacement would result in a discretised-in-space variational principle whilst continuous in time. Spatial matrices would then arise from the discretised variational principle, usually referred to as mass and stiffness/Laplace matrices in FE terminology, as a part of equations that need to be discretised in time. Once the time discretisation is completed, the equations and these matrices would be implemented and solved for ϕ_h, h_h. The presence of nonlinearity can complicate the expressions for the matrices and an iterative method is needed to reach to a solution. To this end, Firedrake has been able to tackle this complication as it provides a user-friendly interface and can solve nonlinear problems with the help of PETSc [10, 11]. It does not require for the matrices to be specified by the user during the space discretisation; it essentially does it behind the scene. It only requires for a time-discrete weak form, a mesh and a function space. Consequently, discretising equation (3.16) in time was the only step remaining before the equations were implemented on Firedrake. The time discretisation of the potential-flow shallow-water equations has been carried out by using the second order Störmer–Verlet scheme.

3.2.2.2 Non-autonomous system

The system has explicit time-dependence due to the existence of the exponential dissipation term $e^{\gamma t}$ and the influx term $Q(t)$. This turned the system from autonomous to non-autonomous which ultimately makes the system more complicated, prone to numerical instabilities with the energy not longer being conserved. Hence work done for non-autonomous systems in [14, 30] was employed.

Starting from non-autonomous Hamiltonian dynamics casted in the following variational principle:

$$0 = \delta \int_0^T L(q,p,t)\,dt = \delta \int_0^T p\frac{dq}{dt} - H(q,p,t)\,dt, \tag{3.17}$$

with a time coordinate t, a conjugate momentum vector $p = p(t)$, a generalised coordinate vector $q = q(t)$ and a time-dependent Hamiltonian $H(q,p,t)$. In general, the non-autonomous variational principle in equation (3.17) can be brought to an autonomous form similar to equation (3.18). This is accomplished by transforming the variables p, q, and by considering time to be an auxiliary variable with a new time coordinate $\tau(t) = t$ as well as introducing a new variable $\tilde{p}$ conjugate to τ. The energy is thus written in a Kamiltonian form [14], [42], defined as $K(Q,P,\tau) \equiv H(Q,P,\tau)+\tilde{p}$. Following the aforementioned steps leads to the autonomous variational principle:

$$0 = \delta \int_0^T P\frac{d\tilde{Q}}{dt} + \tilde{p}\frac{d\tau}{dt} - K(\tilde{Q},P,\tau)\,dt. \tag{3.18}$$

As mentioned earlier, the explicit time-dependence creeps in through the damping term $e^{\gamma t}$ and the forcing. As the latter is applied on the boundary, its effect has not been included in the Kamiltonian. Considering the effect of the momentum dissipation, the non-autonomous variational principle of the mathematical problem is equation (3.6). Equation (3.6) has been compared with (3.18) in an effort to determine the variables $P, \tilde{Q}$ which can turn the variational principle to autonomous. Thus the goal is to determine a transformation for the original variables ϕ, h which can "absorb" the exponential term and result in an autonomous variational principle for the Hele-Shaw problem in the form of equation (3.18). A suitable transformation is the following:

$$P = \phi e^{\tilde{\gamma} t},\ \tilde{Q} = h, \tag{3.19}$$

with $P, \tilde{Q}$ time- and space-dependent functions. Substituting the transformations from equation (3.19) in equation (3.6) leads to:

$$0 = \delta \int_0^T \int_0^L P\frac{d\tilde{Q}}{dt} - \Big(\frac{1}{2}\tilde{Q}(\partial_x P)^2 e^{-\tilde{\gamma} t} + \frac{1}{2}(\tilde{Q} - h_0)^2 e^{\tilde{\gamma} t}\Big) dx \ - Q(t)P(0)dt, \tag{3.20}$$

with the term inside the parenthesis, integrated over space, being the transformed Hamiltonian. Considering time as an auxiliary variable with a new time coordinate $\tau = \tau(t)$, such that $\frac{d\tau}{dt} = 1$ and $\tau(0) = 0$, and introducing a conjugate variable $\tilde{p}$ results in the autonomous variational principle:

$$\begin{aligned} 0 = \delta \int_0^T \int_0^L P\frac{d\tilde{Q}}{dt} + \tilde{p}\frac{d\tau}{dt} - \underline{\Big(\frac{1}{2}\tilde{Q}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} + \frac{1}{2}(\tilde{Q} - h_0)^2 e^{\tilde{\gamma}\tau} + \tilde{p}\Big)} dx \\ - Q(t)P(0)dt, \end{aligned} \tag{3.21}$$

with the underlined term being the Kamiltonian. The variation of equation (3.21) yields:

$$\begin{aligned} 0 = \int_0^T \int_0^L &\delta P\frac{d\tilde{Q}}{dt} - \frac{dP}{dt}\delta\tilde{Q} + \delta\tilde{p}\frac{d\tau}{dt} - \frac{d\tilde{p}}{dt}\delta\tau - \delta P\big(\tilde{Q}(\partial_x P)(\partial_x \delta P)e^{-\tilde{\gamma}\tau}\big) \\ &- \delta\tilde{Q}\big(\frac{1}{2}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} + (\tilde{Q} - h_0)e^{\tilde{\gamma}\tau}\big) - \delta\tau\big(- \frac{\tilde{\gamma}}{2}\tilde{Q}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} + \frac{\tilde{\gamma}}{2}(\tilde{Q} - h_0)^2 e^{\tilde{\gamma}\tau}\big) \\ &- \delta\tilde{p} + P\delta\tilde{Q}\big|_{t=0}^{T} + \tilde{p}\delta\tau\big|_{t=0}^{T} dx - Q(t)\delta P(0)\, dt. \end{aligned} \tag{3.22}$$

Taking end-point conditions $\delta\tilde{Q}(0) = \delta\tilde{Q}(T) = 0$ and $\delta\tau(0) = \delta\tau(T) = 0$ and rearranging the terms leads to:

$$\begin{aligned} 0 = &\underline{\int_0^T \int_0^L \delta P\frac{d\tilde{Q}}{dt} - \tilde{Q}(\partial_x P)(\partial_x \delta P)e^{-\tilde{\gamma}\tau}} - \underline{\delta\tilde{Q}\bigg(\frac{dP}{dt} + \frac{1}{2}(\partial_x P)^2 e^{-\tilde{\gamma}\tau}} \\ &\underline{+ (\tilde{Q} - h_0)e^{\tilde{\gamma}\tau}\bigg)} - \delta\tau\bigg(\frac{d\tilde{p}}{dt} - \frac{\tilde{\gamma}}{2}\bigg(\tilde{Q}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} - (\tilde{Q} - h_0)^2 e^{\tilde{\gamma}\tau}\bigg)\bigg) \\ &+ \delta\tilde{p}\bigg(\frac{d\tau}{dt} - 1\bigg)\, dx \ - Q(t)\delta P(0)dt. \end{aligned} \tag{3.23}$$

If one integrates the expression for $\delta\tau$ in time, an expression for $\tilde{p}(t)$ can be recovered of the form: $\tilde{p}(t) = \int_0^T \frac{\gamma}{2}(\tilde{Q}(\partial_x P)^2 e^{\tilde{\gamma} t} - (\tilde{Q} - h_0)^2 e^{\tilde{\gamma} t})\, dt$, with $d\tau = dt$ yielding $t = \tau$. The underlined terms in equation (3.23) were discretised in time using the second-order Störmer–Verlet method (equation (3.24)) [14, 35]. Symplectic integrators [35] such as this help the discretised system to avoid numerical instabilities and maintain energy conservation, as is the case for the original system. Provided that $P, \tilde{Q}$ are the same as before and that superscripts denote time, the time-discretisation reads:

$$P^{n+1/2} = P^n - \frac{\Delta t}{2} \frac{\partial H(\tilde{Q}^n, P^{n+1/2}, t^{n+1/2})}{\partial \tilde{Q}^n}, \tag{3.24a}$$

$$\tilde{Q}^{n+1} = \tilde{Q}^n + \frac{\Delta t}{2}\Big(\frac{\partial H(\tilde{Q}^n, P^{n+1/2}, t^{n+1/2})}{\partial P^{n+1/2}} + \frac{\partial H(\tilde{Q}^{n+1}, P^{n+1/2}, t^{n+1/2})}{\partial P^{n+1/2}} \Big), \tag{3.24b}$$

$$P^{n+1} = P^{n+1/2} - \frac{\Delta t}{2} \frac{\partial H(\tilde{Q}^{n+1}, P^{n+1/2}, t^{n+1/2})}{\partial \tilde{Q}^{n+1}}. \tag{3.24c}$$

Transforming back to the original variables (ϕ, h), the following weak forms for the respective variations are recovered, with the exception of the $\delta\tau$ equation which is decoupled from the rest. Provided that superscript notation corresponds to time levels the time- and space-discretised potential-flow shallow-water equations read:

First, a half update of ϕ_h:

$$\begin{aligned} 0 = \int_0^L \Big(&\frac{(\phi_h^{n+1/2} \mathrm{e}^{\gamma t^{n+1/2}} - \phi_h^n \mathrm{e}^{\gamma t^n})}{\Delta t/2} + \frac{1}{2}(\partial_x \phi_h^{n+1/2})^2 \mathrm{e}^{\gamma t^{n+1/2}} \\ &+ (h_h^n - h_0)\mathrm{e}^{\gamma t^{n+1/2}} \Big)\, \delta h_h \; dx \end{aligned} \tag{3.25a}$$

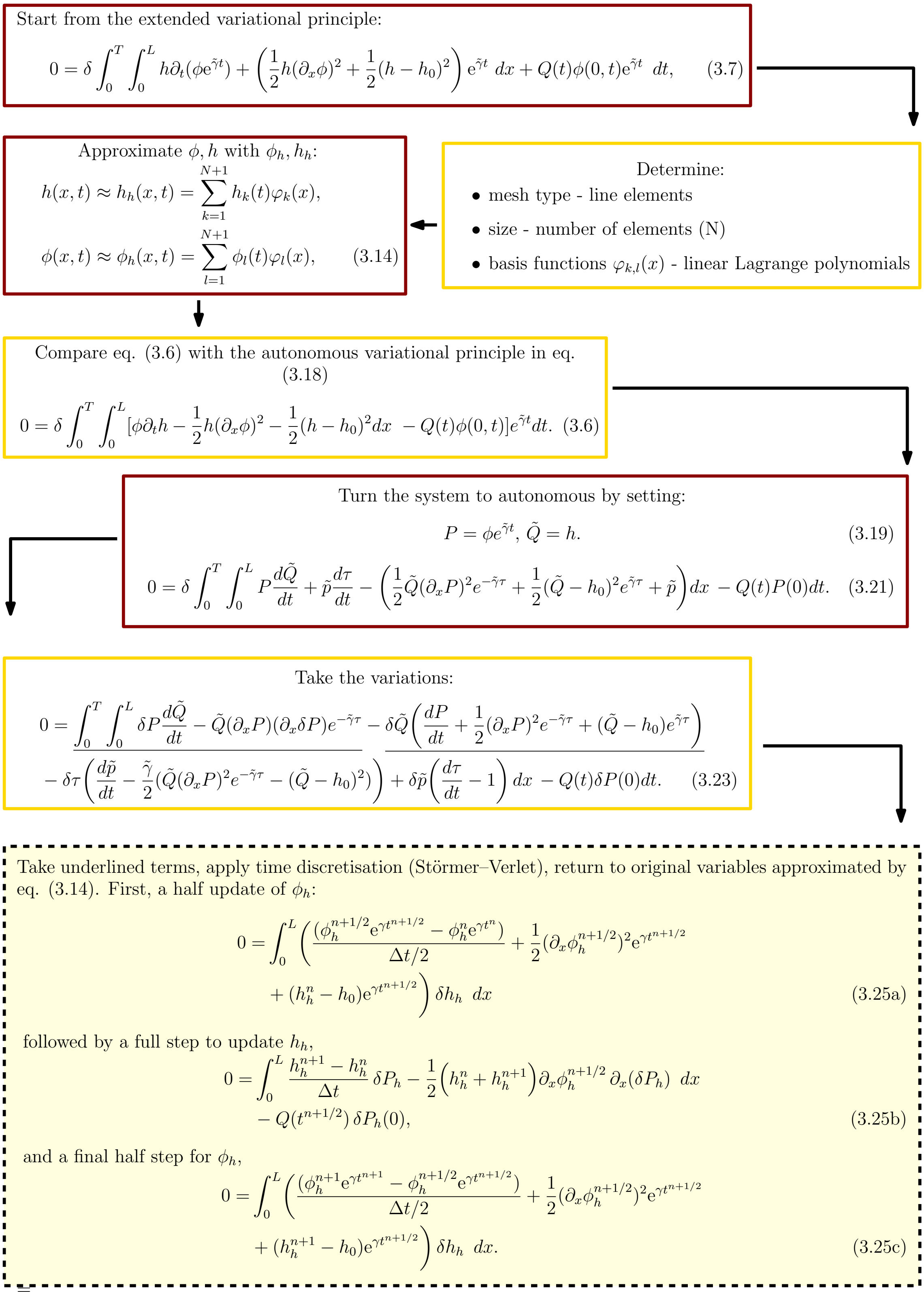
followed by a full step to update h_h,

$$\begin{aligned} 0 = \int_0^L &\frac{h_h^{n+1} - h_h^n}{\Delta t}\, \delta P_h - \frac{1}{2}\Big(h_h^n + h_h^{n+1}\Big) \partial_x \phi_h^{n+1/2}\, \partial_x(\delta P_h) \; dx \\ &- Q(t^{n+1/2})\, \delta P_h(0), \end{aligned} \tag{3.25b}$$

and a final half step for ϕ_h,

$$\begin{aligned} 0 = \int_0^L \Big(&\frac{(\phi_h^{n+1} \mathrm{e}^{\gamma t^{n+1}} - \phi_h^{n+1/2} \mathrm{e}^{\gamma t^{n+1/2}})}{\Delta t/2} + \frac{1}{2}(\partial_x \phi_h^{n+1/2})^2 \mathrm{e}^{\gamma t^{n+1/2}} \\ &+ (h_h^{n+1} - h_0)\mathrm{e}^{\gamma t^{n+1/2}} \Big)\, \delta h_h \; dx. \end{aligned} \tag{3.25c}$$

Equations (3.25) were directly implemented in Firedrake and solved with the nonlinear solver. A snippet of the code illustrates the Firedrake implementation in Chapter 4 Figure 4.3. A sketch summarising the steps leading to the implemented equations is illustrated below.



Start from the extended variational principle:

$$0 = \delta \int_0^T \int_0^L h\partial_t(\phi \mathrm{e}^{\tilde{\gamma}t}) + \left(\frac{1}{2}h(\partial_x\phi)^2 + \frac{1}{2}(h-h_0)^2\right)\mathrm{e}^{\tilde{\gamma}t}\, dx + Q(t)\phi(0,t)\mathrm{e}^{\tilde{\gamma}t}\ dt, \tag{3.7}$$

Determine:

- mesh type - line elements
- size - number of elements (N)
- basis functions $\varphi_{k,l}(x)$ - linear Lagrange polynomials

Approximate ϕ, h with ϕ_h, h_h:

$$h(x,t) \approx h_h(x,t) = \sum_{k=1}^{N+1} h_k(t)\varphi_k(x),$$
$$\phi(x,t) \approx \phi_h(x,t) = \sum_{l=1}^{N+1} \phi_l(t)\varphi_l(x), \tag{3.14}$$

Compare eq. (3.6) with the autonomous variational principle in eq. (3.18)

$$0 = \delta \int_0^T \int_0^L [\phi\partial_t h - \frac{1}{2}h(\partial_x\phi)^2 - \frac{1}{2}(h-h_0)^2 dx\ - Q(t)\phi(0,t)]e^{\tilde{\gamma}t}dt. \tag{3.6}$$

Turn the system to autonomous by setting:

$$P = \phi e^{\tilde{\gamma}t},\ \tilde{Q} = h. \tag{3.19}$$

$$0 = \delta \int_0^T \int_0^L P\frac{d\tilde{Q}}{dt} + \tilde{p}\frac{d\tau}{dt} - \left(\frac{1}{2}\tilde{Q}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} + \frac{1}{2}(\tilde{Q}-h_0)^2 e^{\tilde{\gamma}\tau} + \tilde{p}\right)dx\ - Q(t)P(0)dt. \tag{3.21}$$

Take the variations:

$$\begin{aligned} 0 = &\underline{\int_0^T \int_0^L \delta P\frac{d\tilde{Q}}{dt} - \tilde{Q}(\partial_x P)(\partial_x \delta P)e^{-\tilde{\gamma}\tau}} - \underline{\delta\tilde{Q}\left(\frac{dP}{dt} + \frac{1}{2}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} + (\tilde{Q}-h_0)e^{\tilde{\gamma}\tau}\right)} \\ &- \delta\tau\left(\frac{d\tilde{p}}{dt} - \frac{\tilde{\gamma}}{2}(\tilde{Q}(\partial_x P)^2 e^{-\tilde{\gamma}\tau} - (\tilde{Q}-h_0)^2)\right) + \delta\tilde{p}\left(\frac{d\tau}{dt} - 1\right)dx\ - Q(t)\delta P(0)dt. \end{aligned} \tag{3.23}$$

Take underlined terms, apply time discretisation (Störmer–Verlet), return to original variables approximated by eq. (3.14). First, a half update of ϕ_h:

$$\begin{aligned} 0 = \int_0^L \Bigg(&\frac{(\phi_h^{n+1/2}\mathrm{e}^{\gamma t^{n+1/2}} - \phi_h^n \mathrm{e}^{\gamma t^n})}{\Delta t/2} + \frac{1}{2}(\partial_x\phi_h^{n+1/2})^2\mathrm{e}^{\gamma t^{n+1/2}} \\ &+ (h_h^n - h_0)\mathrm{e}^{\gamma t^{n+1/2}}\Bigg)\,\delta h_h\ dx \end{aligned} \tag{3.25a}$$

followed by a full step to update h_h,

$$\begin{aligned} 0 = \int_0^L &\frac{h_h^{n+1} - h_h^n}{\Delta t}\,\delta P_h - \frac{1}{2}\left(h_h^n + h_h^{n+1}\right)\partial_x\phi_h^{n+1/2}\,\partial_x(\delta P_h)\ dx \\ &- Q(t^{n+1/2})\,\delta P_h(0), \end{aligned} \tag{3.25b}$$

and a final half step for ϕ_h,

$$\begin{aligned} 0 = \int_0^L \Bigg(&\frac{(\phi_h^{n+1}\mathrm{e}^{\gamma t^{n+1}} - \phi_h^{n+1/2}\mathrm{e}^{\gamma t^{n+1/2}})}{\Delta t/2} + \frac{1}{2}(\partial_x\phi_h^{n+1/2})^2\mathrm{e}^{\gamma t^{n+1/2}} \\ &+ (h_h^{n+1} - h_0)\mathrm{e}^{\gamma t^{n+1/2}}\Bigg)\,\delta h_h\ dx. \end{aligned} \tag{3.25c}$$

Figure 3.8: Summary of the steps leading from Luke's extended variational principle to the fully discretised potential-flow shallow-water equations implemented in Firedrake.

3.3 Conclusions

Two one–dimensional models a shallow-water and a potential-flow shallow-water model, used to model water dynamics in a Hele-Shaw cell, have been presented in this chapter. The presence of a wave-pump to generate waves as well as an inherent narrow width between the glass plates, required for the models to be extended so as to include the effects of the above on the flow. An inflow boundary term has thus been included, accounting for the in- and out-flow of water caused by the operation of the pump. A momentum dissipation term linked to the narrow width of the tank and the viscosity of the fluid was also added.

The extended models were solved numerically, with the implementation of each model following its mathematical description. Namely, the shallow-water model was discretised in space and time using a first-order finite-volume and a Forward-Euler method, respectively. The potential-flow shallow-water model was discretised in space and time using a second-order finite-element and a Störmer–Verlet method. The spatial discretisation of the latter model was implemented in Firedrake; an automated software environment that essentially computes the numerical solution under the user's surface, and only requires for a mesh, a weak form, test functions and a time-discretisation. Special attention was paid to the implementation of the boundary conditions (inflow at $x = 0$ and solid wall at $x = L$) of both numerical models. The boundary conditions in the finite-volume method were indirectly applied by introducing ghost cells to the numerical domain. In the finite-element method the inflow boundary condition was added directly in the weak form thus being part of the implemented equations. The validation of the numerical models ensues in the following chapter.

Chapter 4

Numerical simulations for hydrodynamics

In this chapter the goal was to validate the models against the experiments described in Chapter 2. In essence, simulations of each model were performed with the obtained experimental data as an input. In this way, each model was tuned into the experiment thus allowing for its behaviour to be further investigated under the conditions that the experimental data determined. Inserting the data into each model was two-sided. First, the experimental data were processed further prior to their inclusion in each model. Once the data had been fully processed, each numerical model was amended to receive these data as an input.

The data imposed in the numerical models were the influx/area flux $Q(t)$ and the depth at the left wall $h(0,t)$. Processing these data series thus entailed calculating $Q(t)$ from the time-series for the area $A(t)$, smoothing $Q(t)$, $h(0,t)$ to avoid any peaks in the data and fitting a function Q_{fitted} to the influx time series.

The boundary conditions of each numerical model, presented in Chapter 3, were modified to include the experimental data. The inclusion of the data in the modified numerical models was done in two separate simulations for each numerical model, summarised in Table 4.1. In simulation one (sim1), also used in [42], only the influx data were imposed and reproduced by the fitted function. This intermediate step allowed the introduction of the data in the model without any potentially troublesome peaks in the data, left from the smoothing. Then in simulation two (sim2), the smoothed data for both $Q(t)$, $h(0,t)$ entered the code 'directly', in the sense that no fitted function was used and linear interpolation was used to recover data at missing times.

Table 4.1: Summary of the validation stages. Sim1 used only a fitted function for the influx, while $h(0,t)$ was determined internally in the numerical algorithm without any data being used (free). In sim2 the $Q(t)$, $h(0,t)$ smoothed data were imposed directly to the boundary conditions.

Simulation	$Q(t)$	$h(0,t)$
One	Q_{fitted}	free
Two	Q_{final}	$h0_{\text{final}}$

The summary of this chapter is as follows: the processing of the data is presented in Section 4.1, while the changes in the finite-volume model and the finite-element model are described in Sections 4.2 and 4.3, respectively. The validation results from the simulations are presented in Section 4.4 accompanied by the conclusions in Section 4.5.

4.1 Stage one: Data processing and analysis

Running the image-analysis algorithm in Chapter 2 yielded data consisting of two arrays, one for the depth at the left wall $h(0,t)$ and one for the area under the free surface $A(t)$, with both data sets being functions of time. Given that $Q = \frac{dA}{dt}$, $Q(t)$ can be calculated and approximated also as a function of time. Unfortunately, calculating the area influx Q directly from $A(t)$ resulted in very peaky flux data. Hence, smoothing techniques were employed so as to minimise the peaks, which also helped the function fitting required for preparatory stage one. A moving average method was selected for the smoothing: it uses the average of seven neighbouring data points (span) to the current one to readjust its location closer to the span, thus reducing the noise/peaks. The reason behind this choice was that it provided the least peaky results for data originating from tank one when compared against other smoothing methods such as the Savitzky-Golay method and the local regression method.

The smoothing process for Q is summarised in the following sketch. Initially, the area data were smoothed, whereafter an intermediate Q_{smooth} was determined by applying a numerical time derivative (finite difference in time) to these smoothed area data. Subsequently, the final Q (Q_{final}) was determined by smoothing this intermediate Q_{smooth} once more.

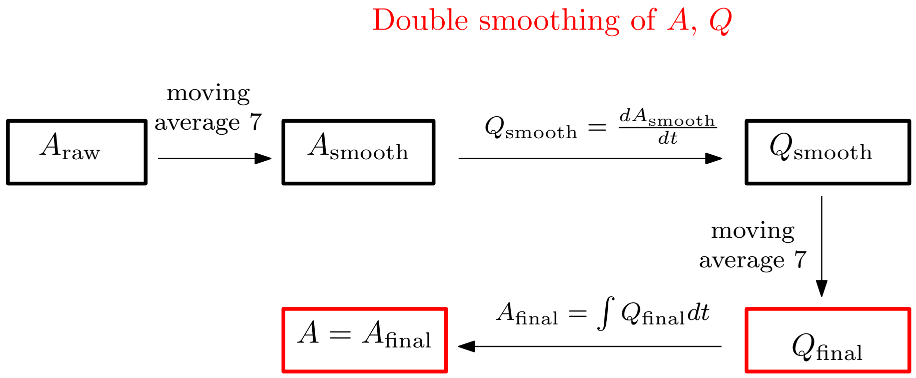
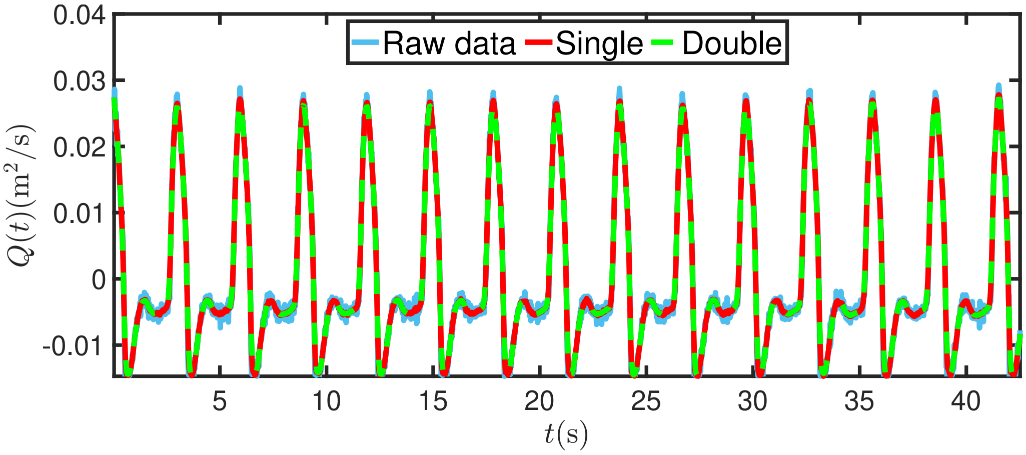


Figure 4.1: Smoothing process for $A(t)$ and $Q(t)$ ultimately leading to Q_{final}, later imposed to the numerical models. One can also recover the double smoothed area data (A_{final}) at the end of the smoothing process.

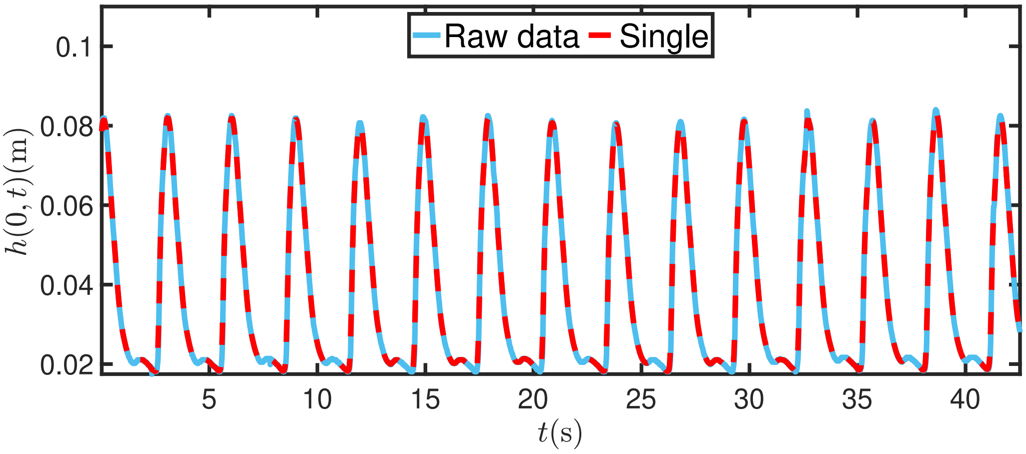
The smoothing for the depth at the left wall $h(0,t)$ was also performed with a moving average method, which was applied once or twice, ultimately yielding $h0_{\text{final}}(t)$. For example in the case of tank one, where the image processing provided relatively accurate results, double smoothing was preferred to minimise the peaks as much as possible. On the contrary, tank two provided very accurate results for the depth such that a second round of smoothing was not required.

4.1.1 Comparison of initial against smoothed data

To visually assess the effect of the smoothing, a comparison between the raw and the smoothed data for $Q(t)$, $h(0,t)$ for case 165 has been visualised in the following figure. Comparison plots for the rest of the cases can be found in Appendix D.1.



(a) Inflow $Q(t)$.



(b) Depth at the left wall $h(0,t)$.

Figure 4.2: Comparison between double (green dash), single (red) smoothed and raw data (light blue) for $Q(t)$ (top panel), $h(0,t)$ (bottom panel), case 165 (tank two).

It can be seen that the smoothing had an influence on the amplitude of $Q(t)$ and no influence on the amplitude of $h(0,t)$, for case 165. This was the case for all tank-two data and was somewhat expected as the colour-channel-detection method provided less erroneous and peaky data. The influence on Q is also confirmed by looking at the data from tank one (Figure D.14); the edge-detection method yielded very peaky data for Q. From the above, it is concluded that double smoothing for $h0_{\text{final}}(t)$ was only required in case 47, where the data were the most spiky. After the smoothing had been completed, the next step was to fit a function to Q.

4.1.2 Step two: Function fitting to area flux data

Superimposing a sinusoidal function to the influx data was essential for simulation one; the goal was to insert this function in the numerical model thus reproducing the (smoothed) experimental data at various times. The pump action was assumed to lead to a straightforward sinusoidal wave parametrised as follows:

$$Q_{\text{fitted}} = Q_a \sin(2\pi f t_n + c), \tag{4.1}$$

with f the wave frequency, Q_a the influx amplitude and c the phase of the fitted function; all determined from the smoothed influx data. The wave frequency had been calculated during the experiments, and it was thus known. Q_a was determined by $Q_a = \frac{\max(Q_{\text{final}})-\min(Q_{\text{final}})}{2}$, while c was the angle that brought the fitted function in phase with the data. Essentially, each experiment provided a unique combination of f, Q_a and c, summarised in Table 4.2. At this point it should be pointed out that smoothing preceded the fitting because it ensured that any peaks in the data would disappear hence providing a "finalised" value for Q_a.

Table 4.2: Values of the flux amplitude (Q_a), wave frequency (f) and phase (c) that determine the fitted inflow of equation (4.1) for each case.

Cases	$\mathbf{Q_a}$(m^2/s)	f(Hz)	c(radians)	**Figures**
47	-0.0188	0.7042	0	4.3
149	0.01569	0.675	$\pi/2$	D.17
154	0.0136	0.48	$\pi/6$	D.18

By construction the tank could not hold the water inside unless the pump was switched on. This resulted in an initial condition that was not a state of rest ($u = 0$, $h = H_0$), hence affecting the data. As the exponential damping indicates, the system looses memory of its initial state and momentum will thus decay at a rate of $1/\gamma$. Thus starting the simulation from the recorded data, produced after the pumps had been switched on, and allowing for a couple of wave periods for the system to sync to the experiment resolved the initialisation problem. The smoothed Q data as well as the functions fitted to them for each case, apart from case 165, can be visualised in Figure 4.3 and in Figures D.17 and D.18.

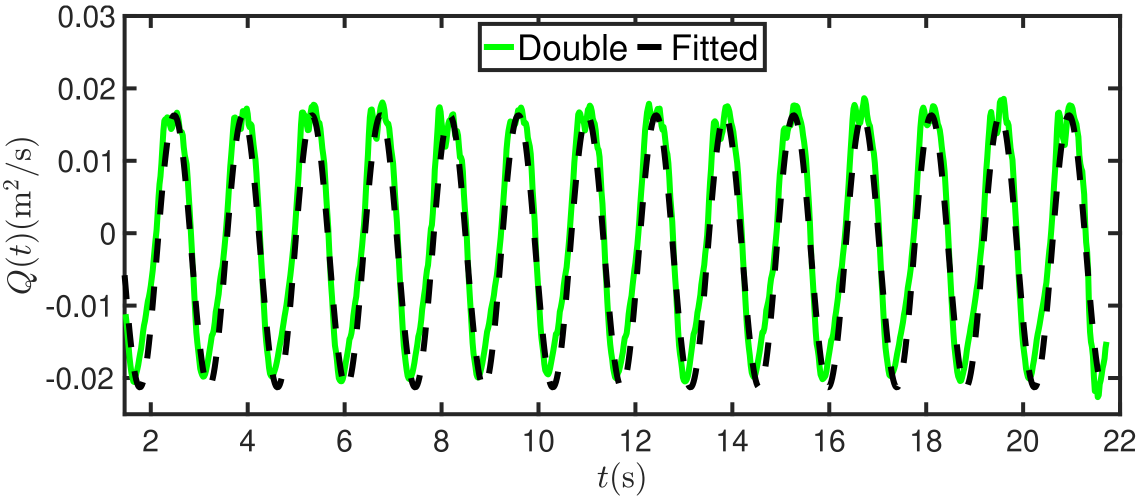


Figure 4.3: Case 47: Double smoothed data for $Q(t)$ (green) compared against the fitted function $Q_{\text{fitted}}(t) = -0.0188\sin(2\pi * 0.7042t)$ (black). This plot was used in [42].

After Q_{fitted} had been determined, the next step was to modify the boundary conditions where the fitted function for each experiment would be imposed on.

4.2 Alterations to the finite-volume model

In the finite-volume method, the ghost cells are introduced to help calculate the flux at the boundaries which then lead to the satisfaction of the boundary conditions. For simulation one (sim1), the value of the depth at the left ghost cell (h_L) of the domain was set to be the same as the value of the depth at the right cell (h_R). In addition, the value of $h_L u_L$ at the ghost cell was set equal to Q_{fitted}, while h_R, u_R were determined from the values of the numerical solution evaluated at the first cell.

In simulation two (sim2), both the depth and the inflow of water were determined by the data. Thus the measurements $h0_{\text{final}}$, Q_{final}, originating from the experiments were imposed directly. As mentioned earlier, linear interpolation was applied to recover values of Q at unknown times from known (neighboring) data. The interpolation process was optimised by ensuring that the known time entries were tracked within the data set and were as close as possible to time t_n. The modified boundary conditions at the left wall, for each simulation, are the following:

(i) Simulation one:

$$\begin{bmatrix} h_L \\ h_L u_L \end{bmatrix} = \begin{bmatrix} h_R \\ Q_{\text{fitted}} \end{bmatrix},$$

such that $\partial_x h|_{x=0} = 0$ and $hu|_{x=0} = Q_{\text{fitted}}$.

(ii) and for simulation two:

$$\begin{bmatrix} h_L \\ h_L u_L \end{bmatrix} = \begin{bmatrix} h0_{\text{final}} \\ Q_{\text{final}} \end{bmatrix},$$

such that $h|_{x=0} = h0_{\text{final}}$ and $hu|_{x=0} = Q_{\text{final}}$. The right solid-wall boundary condition stayed the same for both simulations, and was imposed at $x = L$ by setting:

$$\begin{bmatrix} h_R \\ h_R u_R \end{bmatrix} = \begin{bmatrix} h_L \\ -h_L u_L \end{bmatrix},$$

which implies that $\partial_x h|_{x=L} = 0$ and mimics $u(L,t) = 0$ by taking an equal but opposite jet (recall Figure 3.3). The total number of boundary conditions remained the same for either simulation, with the depth at the left ghost cell determined either internally for sim1 or from the data for sim2. Then, the HLL flux would 'decide' which information to use, based on the eigenspeeds. This means that the boundary condition, which is applied in each simulation by allocating the influx and depth data, does not necessarily imply that the HLL flux will use the data provided (see Chapter 4, Figure 4.18).

4.3 Alterations to the finite-element model

The finite-element model was validated in a similar way to the finite-volume model; with some minor changes to inserting the data. Linear Lagrange polynomials were used for the basis functions and the boundary conditions for each simulation were similar to the ones in the finite-volume method. For the finite-element code and with a second-order time discretisation scheme in effect, the influx at half-time step $Q(t_{n+1/2})$ $\big(i.e., Q_{05}\big)$, had to be determined.

Similarly to the FV method, the influx during simulation one was determined from the fitted function, while for simulation two it was directly determined from the data. In contrast to the FV method, the boundary condition was not implemented via the numerical flux but was introduced after integration by parts in the weak formulation equation (3.25b). Hence the solver, which was where the boundary condition came in effect, utilised the updated value of $Q(t_{n+1/2})$ and prescribed it at $x = 0$. For simulation two $h0_{\text{final}}(t)$ was not imposed via the weak formulation but was nonetheless imposed by taking out the degree of freedom at $x = 0$; essentially by applying a Dirichlet boundary condition through a built-in command directly rather than manually after the update of the solution. In the following figure, snippets of the implementation in Firedrake have been illustrated.

Firedrake implementation:

```
#----------------------------------Q----------------------------------#
if use_data=='True':
        # Sim2-Interpolation for Q
        Qint=Q_interpolation(t, Qdata, h0data)
        Q_t_expr.Q=Qint # Updating the value of Q
else:
        # Sim1-Use fitted function
        Q_t_expr = Expression("-Ampl*sin(2*pi*F*t+c)")
Q_t0_5.interpolate(Q_t_expr)
```

```
#————————————————————————————————————————h(0,t)————————————————————————————————————————#
# Sim2—Depth interpolation
h0d=find_depth(t, Qdata, h0data);
# Define the dirichlet boundary
bcval1.assign(h0d)
bc=DirichletBC(V,bcval1,1)
```

Figure 4.3: Computation of the influx $Q(t_{n+1/2})$ (top panel) and allocation of $h(0,t)$ (bottom panel). For sim1, Q was computed by the superimposed function evaluated at $t = t_{n+1/2} = t + 0.5dt$ and $h(0,t)$ was not determined from the data. For sim2 $Q_{\text{final}}(t)$, $h0_{\text{final}}(t)$ entered the model directly and linear interpolation was employed. For each time t_n in sim2, a function bcval was used to store the interpolated depth, which was then passed on the Dirichlet boundary condition (bc). Lastly, $Q(t_{n+1/2}), \mathrm{bc}$ were called in the solvers where they came into effect (see following figure).

```
#————————————————————————————————————————Solvers————————————————————————————————————————#

# Variational problem for phi equation (called at first/third step) #
Fphi = ( v*(phi1*exp(gamma*t1)—phi0*exp(gamma*t0))/(0.5*dt) + (v*(h0—
    h_0) +0.5*inner(grad(phi0_5),grad(phi0_5))*v )*exp(gamma*t0_5) )*dx

phi_problem = NonlinearVariationalProblem(Fphi, phi1)
phi_solver = NonlinearVariationalSolver(phi_problem)
return phi_solver;
```

```
# Variational problem for h equation (called at second step) #
Fh = ( v*(h1-h0)/dt - 0.5*(h0+h0_5)*inner(grad(v),grad(phi0_5)) )*dx -
    Q_t0_5*v*ds(1)

if use_data == 'True':
        #Apply Dirichlet to the left boundary
        h_problem = NonlinearVariationalProblem(Fh, h1, bcs=[bc])
else:
        h_problem = NonlinearVariationalProblem(Fh, h1)
        h_solver=NonlinearVariationalSolver(h_problem)
return h_solver;
```

Figure 4.3: Implementation of equations (3.25). Steps one and three in equations (3.25) are performed by solver_phi, while step two from solver_h. The test functions from system (3.25) have been implemented with their name changed: $v \equiv \delta h_h$ and $v \equiv \delta P_h$. The data entered the model through the boundary conditions (underlined terms).

4.4 Numerical validation

In this section the numerical results, the output of simulations from the two numerical models are compared with each other and the experimental data. Two scenarios were investigated with a summary of the cases made in Figure 4.4. Firstly, a comparison of the two simulations (one and two) under the finite-volume and the finite-element method is carried out in Sections 4.4.1 and 4.4.2. Secondly, comparisons of the finite-volume and the finite-element method for simulation one and two are carried out in Section 4.4.3. In other words, either the numerical method is kept constant and the improvement between the two simulations within that method is investigated or alternatively, the simulation is kept the same and the performance of the numerical methods for that specific simulation is investigated.

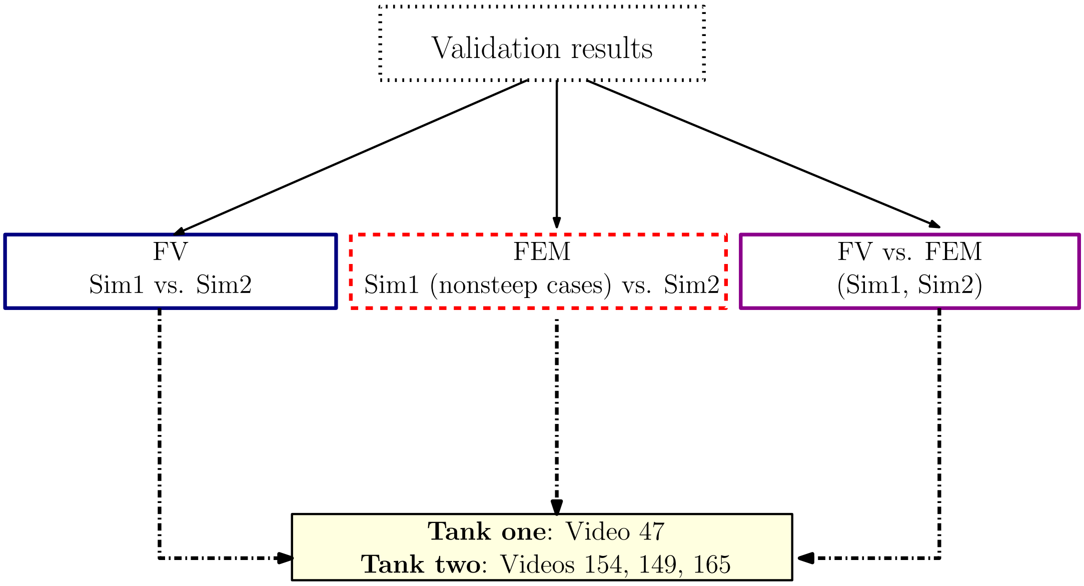


Figure 4.4: Diagram of the comparison of the results. The results have been divided into two categories. Comparison between simulations (blue, red) or comparison between numerical methods (purple). For each case, experimental data originating from two tanks have been used (tank one and tank two).

The relative error for the depth between simulations and measurements, quantified by equation (4.2) and defined as the quotient of the L_2-norm of the difference between two quantities over the

L_2– norm of the most precise one, was computed. For example the numerical depth computed from the finite-element method, was compared with the measured depth as follows:

$$\mathrm{RE} = \frac{||\mathrm{h}_{\mathrm{numerical}} - \mathrm{h}_{\mathrm{measured}}||_2}{||\mathrm{h}_{\mathrm{measured}}||_2}. \tag{4.2}$$

As mentioned earlier, the boundary conditions in the two numerical methods were imposed in a different way. It is therefore of interest to assess the numerical models' ability to adapt to the imposed data and the effect the latter had on the solution approximations. To this end, a similar formula to equation (4.2) was used to assess how close the depth approximations from each numerical method were to each other:

$$\mathrm{RE} = \frac{||\mathrm{h}_{\mathrm{FV}} - \mathrm{h}_{\mathrm{FE}}||_2}{||\mathrm{h}_{\mathrm{FE}}||_2}. \tag{4.3}$$

It should be pointed out that the L_2 - norm takes the square root of the sum of the squares of the differences between the depth values in comparison. It thus provides an indication on how close the two numerical methods are to each other. However, it doesn't indicate the method closest to reality. In what follows, the results from the simulations will be displayed over the analysed experimental data overlaid on snapshots of the experiments, in order to facilitate a direct visual comparison. A summary of the examined cases, accompanied by their respective parameters and associated figures, is given in the following table.

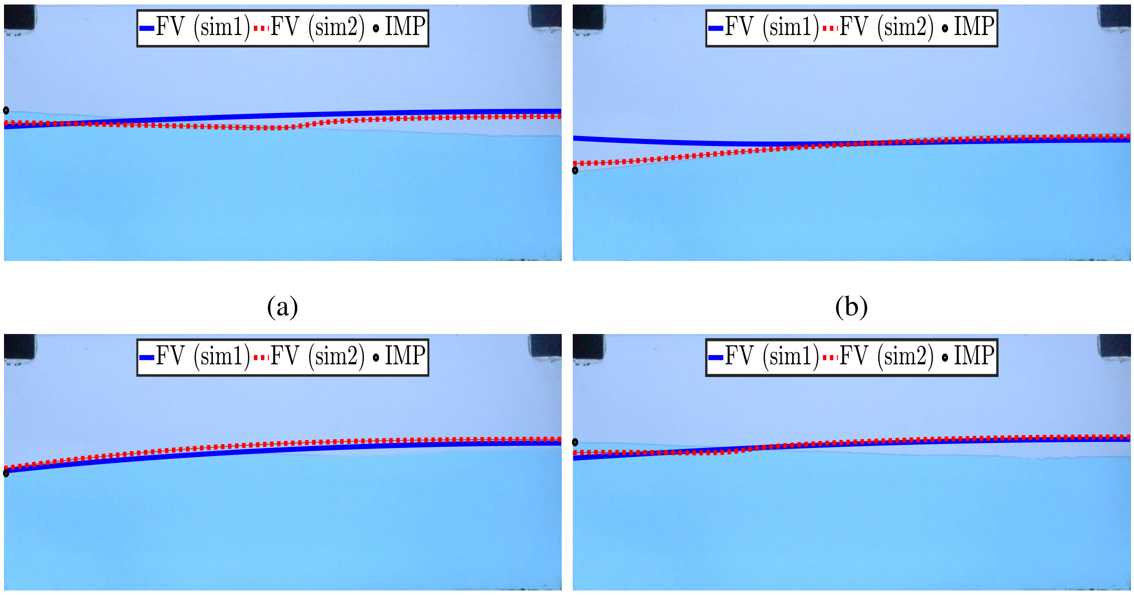
Table 4.3: Summary of the experimental cases. Each experiment had a frequency f, influx amplitude Q_a and approximated depth at rest H_0. The figures accompanying the analysed cases have been provided.

Cases	$\mathbf{f}(\mathrm{Hz})$	$\mathbf{Q_a}(\mathrm{m}^2/\mathrm{s})$	$\mathbf{H_0}(\mathrm{cm})$	**Figures**
47	0.704	0.0188	8.82	D.19
154	0.48	0.0136	8.2	4.5, 4.6, 4.10, 4.11, 4.12
149	0.675	0.0157	5.74	4.7, 4.13, 4.14
165	0.338	-	4.28	4.15, 4.16

4.4.1 Finite-volume results

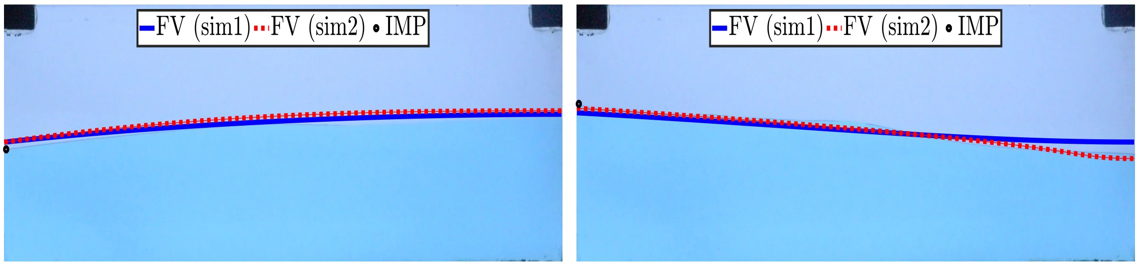
In this section, the finite-volume model was used to approximate the hydrodynamics in the Hele-Shaw cell using the (boundary) data. Recall that simulation one requires fitted functions for Q while simulation two uses the smoothed data for Q, h_0 directly.

The finite-volume method is equipped to deal with discontinuities, hence the steepest wave simulations have been carried out and are analysed below. The comparison between the two simulations has been performed in an effort to visualise and quantify the positive effect of the inclusion of the experimental data on the boundary conditions and on the overall approximation of the water depth. Case 154 has been presented first as the measurements were obtained in deeper water depth with the wave steepness being small. It is thus a good initial test case as it does not challenge the numerical methods.



(a) (b)

(c) (d)



(e) (f)

Figure 4.5: Comparison between the results of sim1 and sim2 for the FV method, case 154. Sim1 (blue) has only the influx fitted function imposed. Sim2 (red dash) has the data for $Q_{\text{final}}(t)$, $h0_{\text{final}}(t)$ at $x = 0$ imposed. Results have been plotted together with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 5\text{s}, 6\text{s}, 15.8\text{s}, 23.8\text{s}, 30.4\text{s}, 42\text{s}$. The momentum damping factor is equal to $\gamma = 3\ \text{s}^{-1}$. 800 elements have been used. A video of the above comparison can be found here: `https://youtu.be/hBHZzQb9Fp4`.

It can be seen in Figure 4.5 that sim1 overestimates the inflow of water entering the cell, compared to sim2, ultimately overestimating the whole free surface. The biggest difference between the two simulations is particularly observed when the water enters the cell ($t = 6\text{s}$) or when the waves propagate in the tank ($t = 23\text{s}$). Sim2 is thus more accurate than sim1, a fact also confirmed 'by eye' in Figure 4.6, where the error of the overall depth between the simulations is shown.

It can be seen in Figure 4.6 that the largest error is observed when the water enters the cell, an event highly dependent on the boundary conditions of each simulation, while both simulations are the closest to the data at the right wall. Since the tank's right boundary was not quite but almost a solid-wall boundary, less reflection occurred at the right-hand side region in reality. It is thus expected that all simulations are likely to overestimate the depth at the area of reflection. In spite of the more precise computed depths and the near-wall (numerical) depth value getting closer to the real value (black dot) in sim2, there are times that a difference is still particularly visible. The origin of this difference will be discussed further in the next sections.

The same conclusions that hold for case 154 hold for case 149.

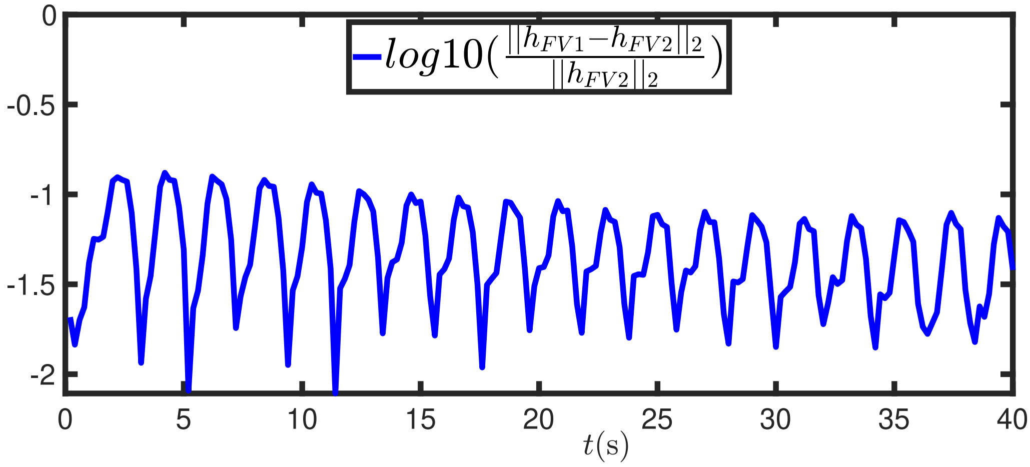
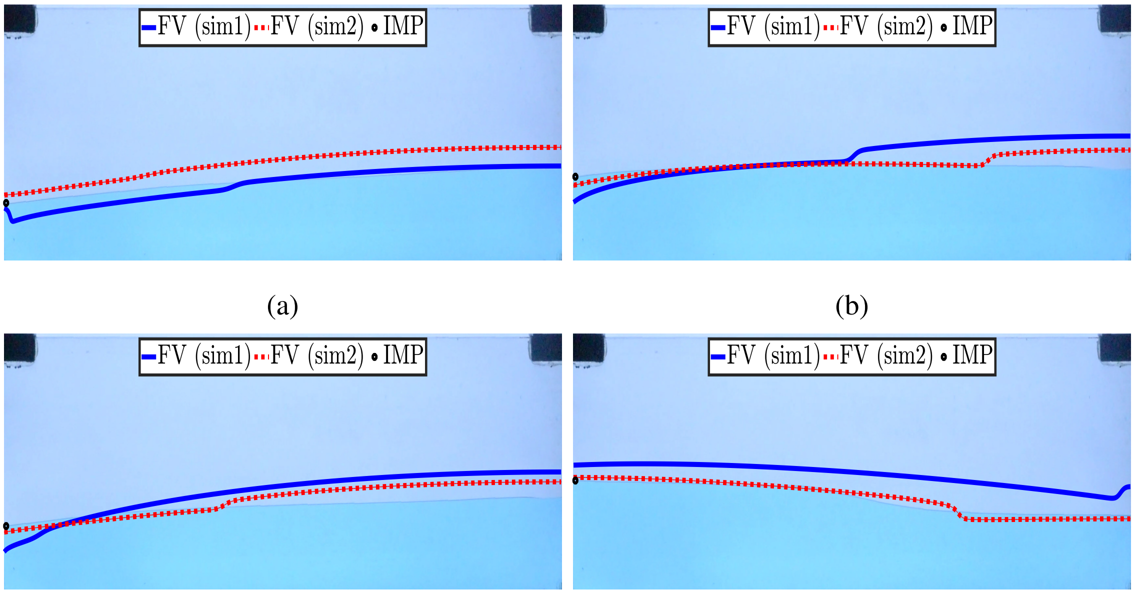
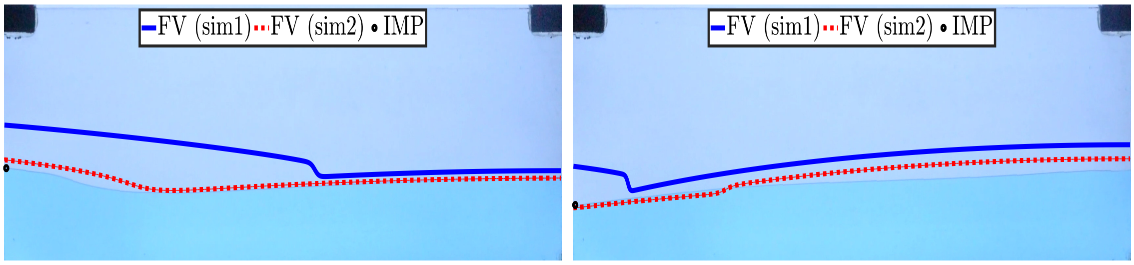


Figure 4.6: Logarithm of the relative error for the depth as computed in sim1 and sim2 from the FV method, case 154. The two methods are the closest during the reflection of the water from the right wall. By inspecting individual snapshots it is found that the largest error is observed when the water enters the cell ($t = 6$s) and the smallest one is right after reflection ($t = 23.8$s).



(a) (b)

(c) (d)



(e) (f)

Figure 4.7: Comparison between the results of sim1 (blue) and sim2 (red dots) for the FV method, case 149. Results have been plotted together with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 1\text{s}, 11\text{s}, 14.2\text{s}, 21\text{s}, 34\text{s}, 41\text{s}$. The momentum damping factor is equal to $\gamma = 3\ \text{s}^{-1}$. 800 elements have been used. A video of the above comparison can be found here: `https://youtu.be/JCKPNVETmgM`.

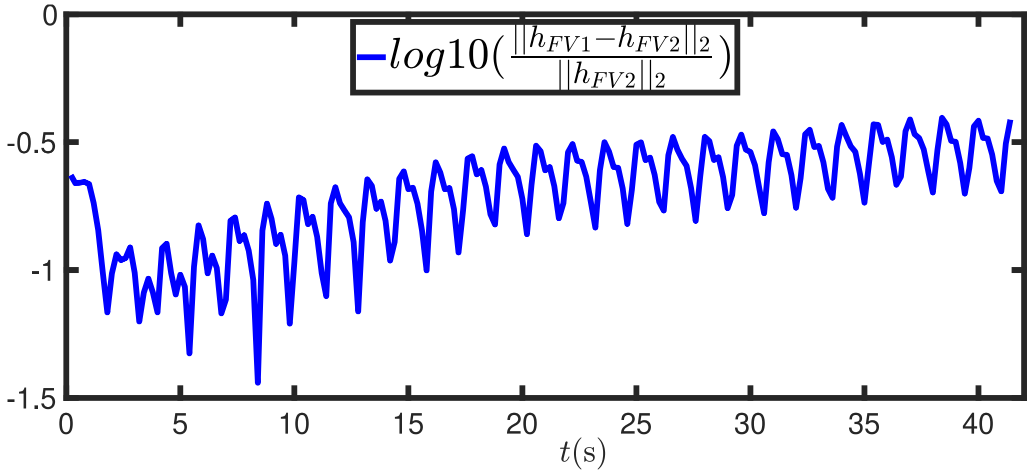
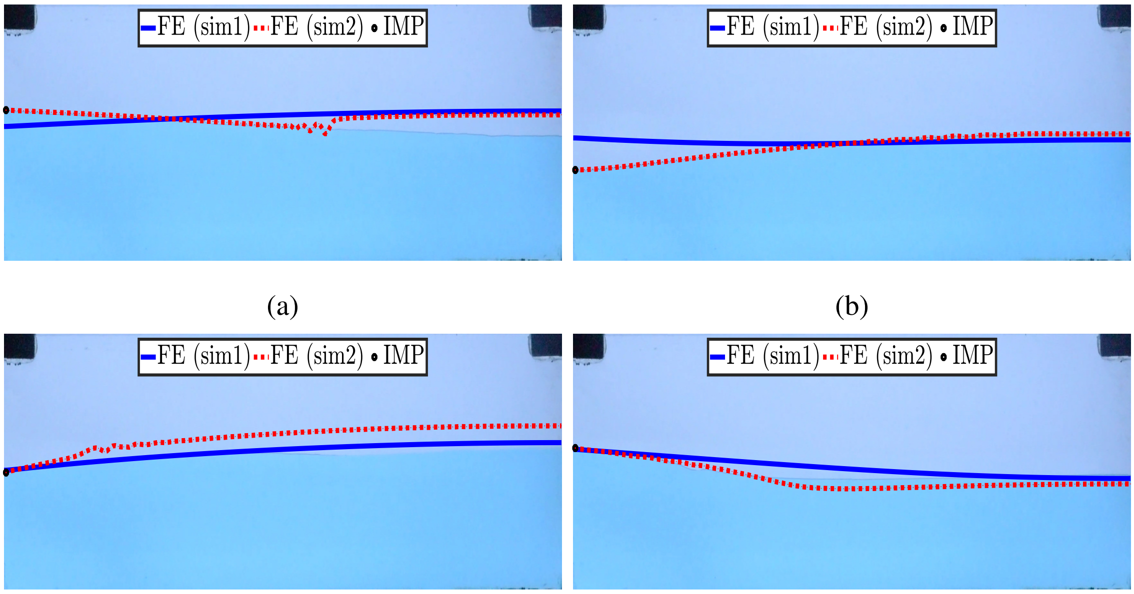


Figure 4.8: Logarithm of the relative error for the depth as computed in sim1 and sim2 from the FV method, case 149. Maximum error is found when the water enters the cell ($t = 34$s) in Figure 4.7. Minimum error is achieved when the tank is being emptied out of the water ($t = 14.2$s). The error between the simulations has increased as inflow amplitude and steepness increased.

The logarithm of the relative error for the depth between the two simulations for cases 154, 149 (see Figures 4.6, 4.8) was higher for case 149 hence revealing a larger disagreement between sim1 and sim2 for case 149. This increasing difference between the two simulations was also observed 'by eye' in the overlaid snapshots (see Figures 4.5, 4.7), where sim2 was a better fit for the captured free surface line while sim1 overestimated it particularly during case 149. Recalling that case 154 has deeper and less steep water waves while case 149 has shallower and steeper waves, it is thus concluded that steeper and shallower waves are approximated better from sim2 than from sim1.

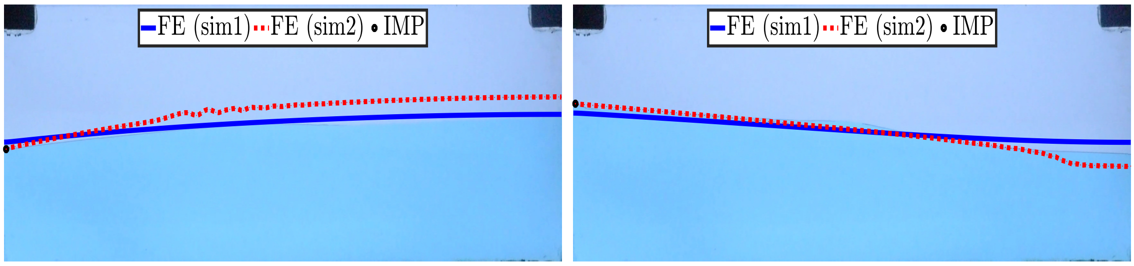
4.4.2 Finite-element results

As the wave dynamics were too steep for sim1 in the FE method resulting in the non-convergence of the FE method (case 149) only case 154 has been illustrated.



(a) (b)

(c) (d)



(e) (f)

Figure 4.9: Comparison between the results of sim1 and sim2 for the FE method, case 154. Results have been plotted together with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 5\text{s}, 6\text{s}, 15.8\text{s}, 23\text{s}, 30.4\text{s}, 42\text{s}$. The momentum damping factor is equal to $\gamma = 3\ \text{s}^{-1}$. 200 elements have been used. The video of the above comparison can be found here: `https://youtu.be/_94wBY1shhs`.

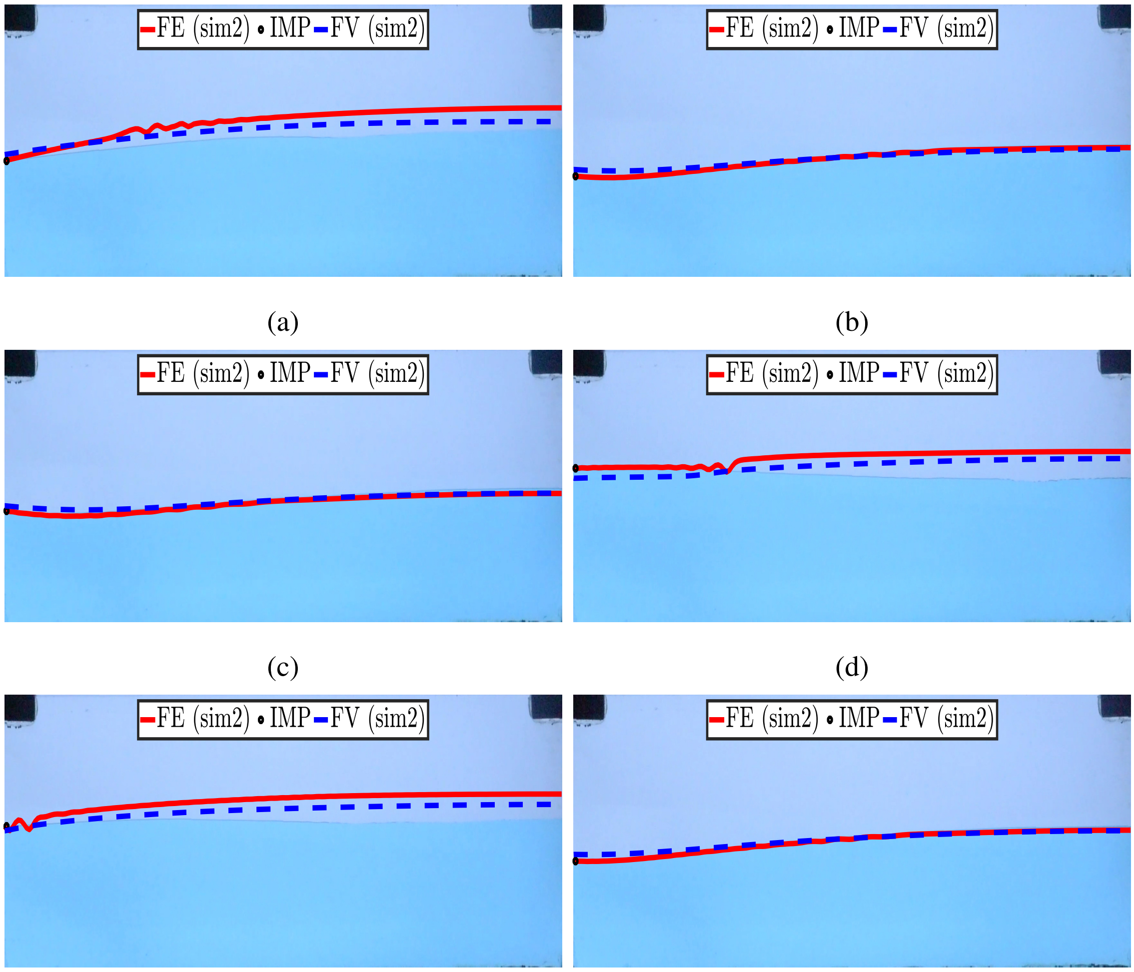
Paradoxically, in the previous figure for case 154 instabilities appeared during sim2, instead of sim1. It is understood that imposing shallower depths through the data, is the reason for these instabilities. Consequently, the plot of the relevant error between the numerical depths has not been provided.

4.4.3 Finite-volume versus finite-element results

Cases 154, 149 and 165, involving waves of increasing steepness, were approximated from each numerical method for sim2. Only the FV results have been provided for cases 149 and 165 as the FE method became highly unstable due to reaching the limit of the irrotational assumption.

Prescribing the depth was not always effective for the FV method, a fact confirmed by the data $h(0,t)$ (black dot) not coinciding with the FV depth (blue line). The origin of this difference is investigated later on in Section 4.4.4.

In contrast to the FV method, the FE method picked up the imposed depth but not without consequences. Instabilities appeared after reflection, in spite of the waves not being steep. The same instabilities appeared for a finer resolution (400 elements) and confirm that the flow is in



(a) (b)

(c) (d)

(e) (f)

Figure 4.10: Comparison between the results of sim2 for FV (blue dashed) and the FE (red), case 154. Results have been plotted together with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 3.4\text{s}, 4\text{s}, 24.8\text{s}, 28\text{s}, 30.2\text{s}, 43.4\text{s}$. The momentum damping factor is equal to $\gamma = 3\ \text{s}^{-1}$. 800 (FV) and 200 (FE) elements have been used. A video of the above approximation can be found here: `https://www.youtube.com/watch?v=s74vjT3aiUE`.

the limit of becoming irrotational. Lastly, similarly to the earlier cases, the numerical models overestimated the reflected depth.

The relative errors between the numerical methods and the numerical methods and the measurements have been visualised in the following figure, ensued by the values of $h(0,t)$ used in the FV and FE method in Figure 4.12.

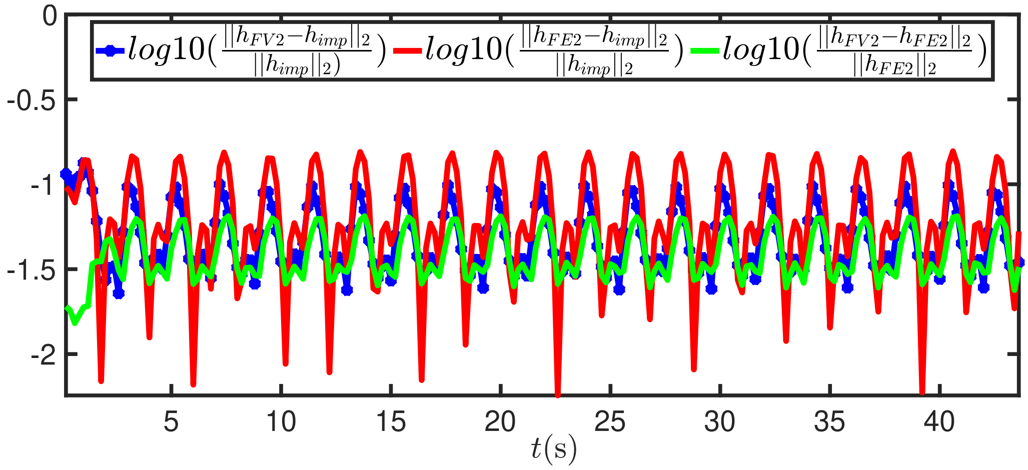


Figure 4.11: Logarithm of the relative errors for the computational depth, obtained during sim2 case 154, between the numerical models (green) and each model and the measured depth value (blue, red). Inspecting the snapshots revealed that the maximum error is achieved right after reflection ($t = 3.4\text{s}, 28\text{s}, 30.2\text{s}$), while the error is minimised once the water is on its minimum depth ($t = 4\text{s}, 24.8\text{s}, 43.4\text{s}$).

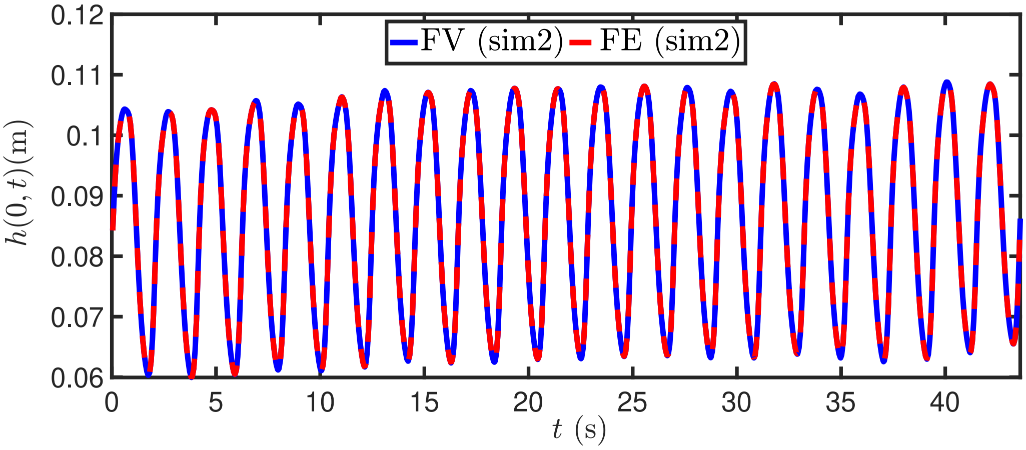
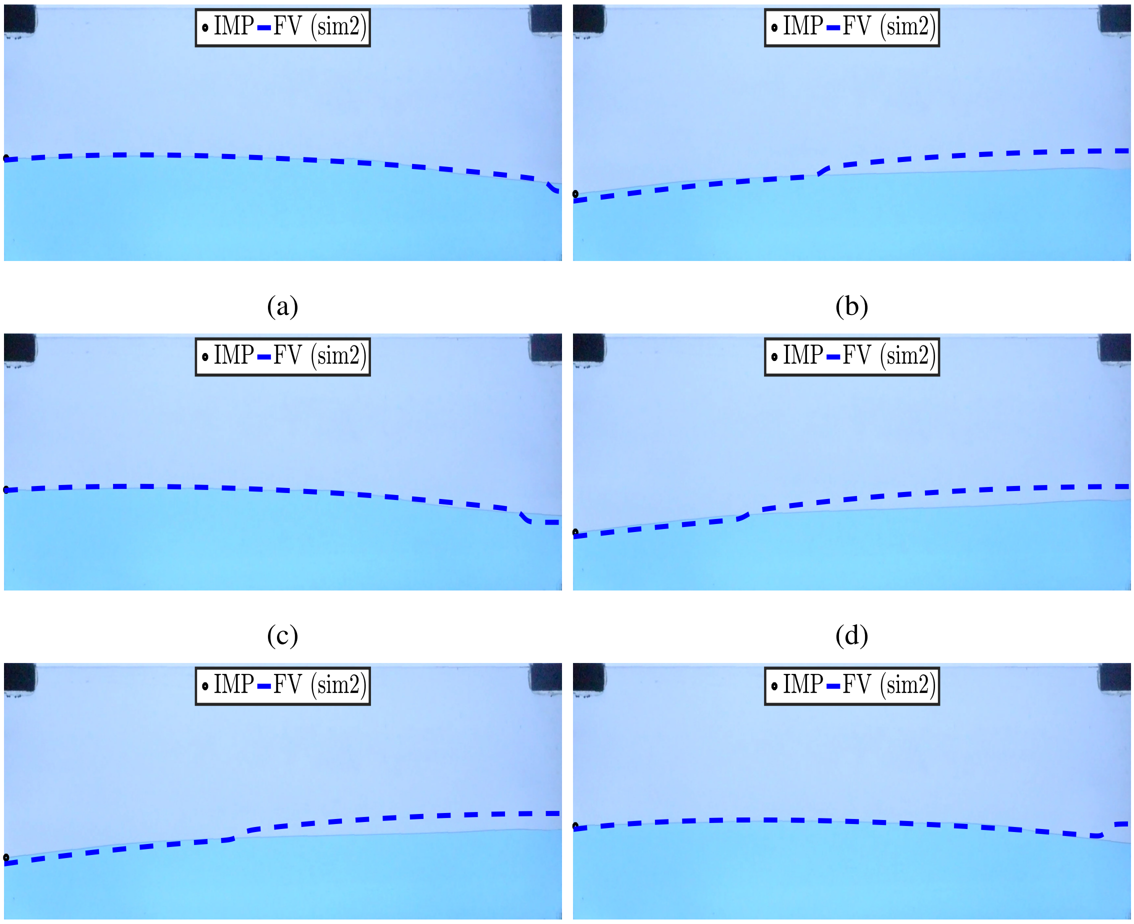


Figure 4.12: Comparison of the $h(0,t)$ data between the FE (red dashed) and the FV method (blue) for sim2, case 154. The difference between the two is quantified to be in the range of $(O(-6), O(-4))$ orders of magnitude.

The same observations made for case 154 apply for case 149 that ensues.



(a) (b)

(c) (d)

(e) (f)

Figure 4.13: Results of sim2 for the FV method (blue dash) for sim2, case 149. The numerical depth has been plotted with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 0.6\text{s}, 1\text{s}, 11\text{s}, 21\text{s}, 34\text{s}, 41\text{s}$. The momentum damping factor is equal to $\gamma = 3\ \text{s}^{-1}$. 800 elements have been used. A video of the above approximation can be found here: `https://youtu.be/3p2qiW8Y4Qw`.

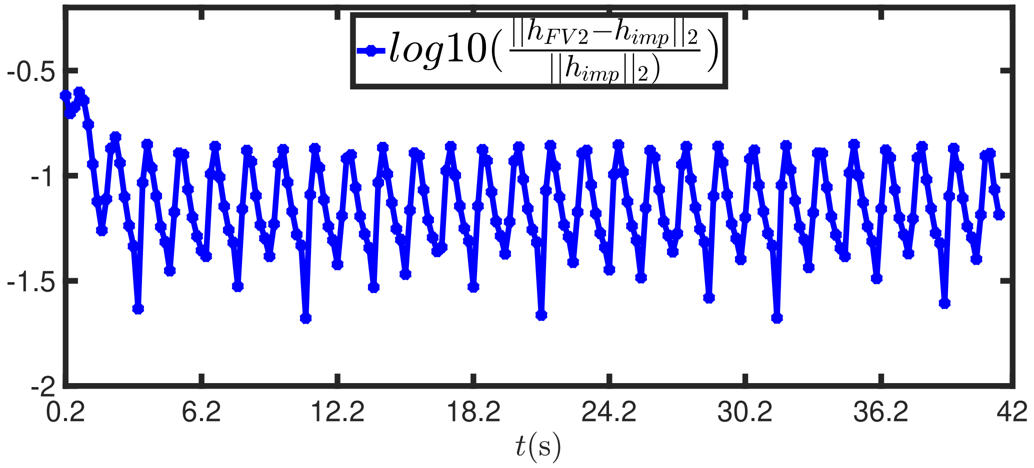
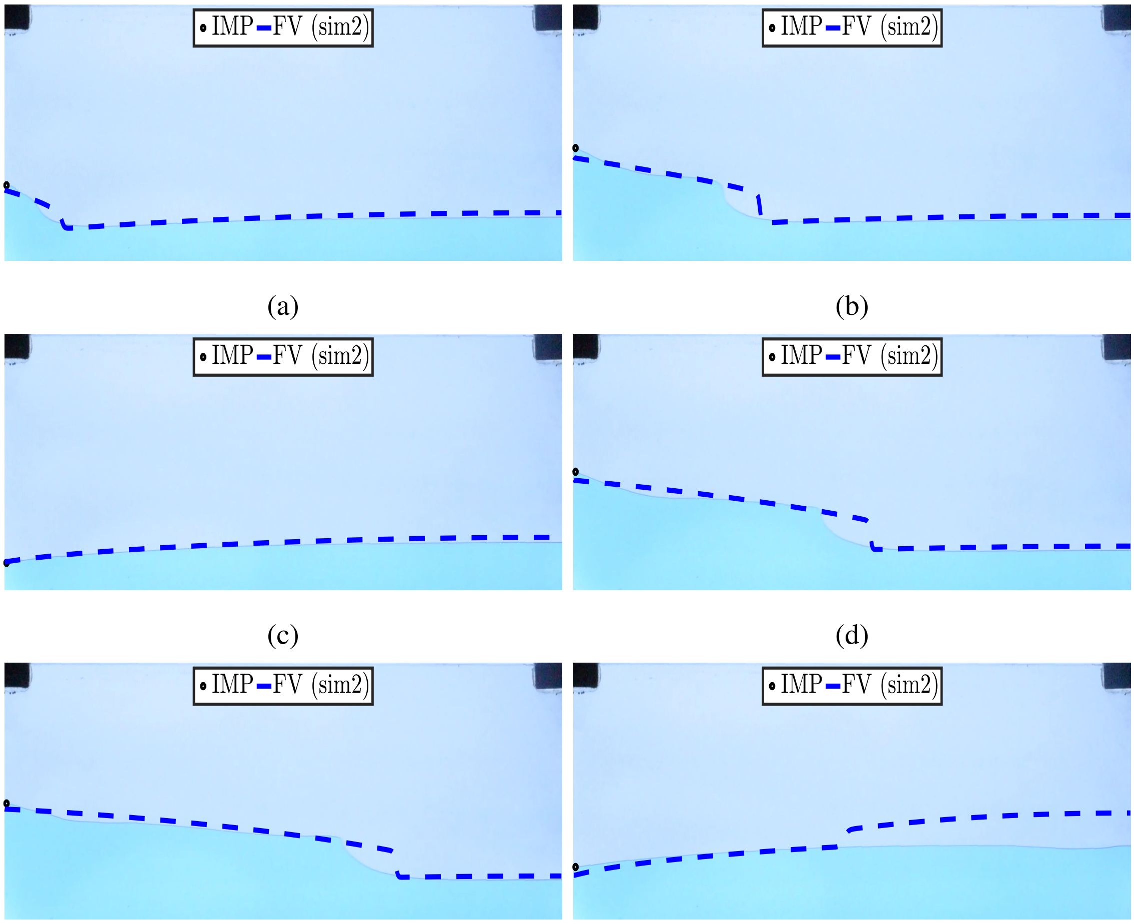


Figure 4.14: Logarithm of the relative error for the depth obtained from the FV method (blue) for sim2, experiment 149. Similarly to case 154, maximum error occurs at wave reflection ($t = 11.2\mathrm{s}, 20.2\mathrm{s}, 32\mathrm{s}$) and minimum error when the minimum water depth is in the tank ($t = 10.8\mathrm{s}, 18.2\mathrm{s}, 34.6\mathrm{s}$). Errors stabilise around $t = 3\mathrm{s}$ possibly due to the decay of momentum, following the exponential decay rule with a rate of $\frac{1}{\gamma} \approx 0.33\mathrm{s}$.

The final case studied, 165, concerns the steepest waves, almost overturning on shallow water. It also has the shallowest water depth while the pump action is very strong thus exceeding an influx threshold of 0.02 m^2/s (recall Figure 4.2a). The same observations made for cases 154,149 apply for case 165. The only difference is that the error between the measured and the numerical depth is larger in magnitude (see Figure 4.16).

It can be seen in Figure 4.15 that the numerical depth is very close to the real depth. The numerical wave front slightly overestimates the real wave front. This difference may be a byproduct of the HLL flux not using the prescribed data exactly or the result of surface tension. Even though alcohol was added in the water to minimise surface tension effects, the food dye increased the surface tension of the flow. This change was manifested with the water not flowing as easily as before and rather getting "stuck" between the glass plates forming little ripples (see Figure 5.28 $t = 3\mathrm{min}30\mathrm{s}$).



(a) (b)

(c) (d)

(e) (f)

Figure 4.15: Results of sim2 for the FV method (blue dashed) for sim2, case 165. The numerical depth has been plotted with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 14.6\text{s}, 14.8\text{s}, 23.2\text{s}, 26.8\text{s}, 35.8\text{s}, 36.4\text{s}$. The momentum damping factor is equal to $\gamma = 3$ s^{-1}. 800 elements have been used. A video of the approximation can be found here: `https://www.youtube.com/watch?v=PvhaJb_AWfk`.

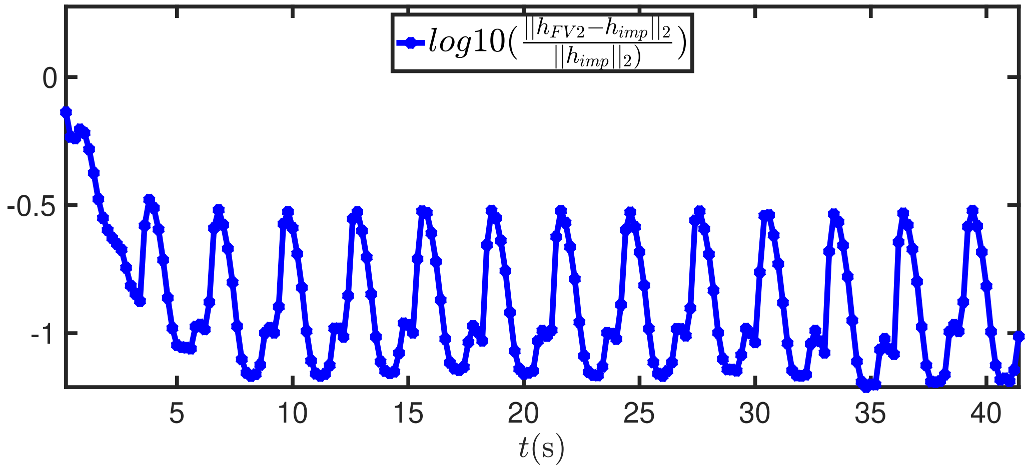


Figure 4.16: Logarithm of the relative error for the computational depth obtained from the FV method in sim2, case 165. Maximum error reached at reflection ($t = 36.4$s), minimum error reached when water is on its minimum depth ($t = 23.2$s).

Comparing the magnitude of the finite-element instabilities on the overlaid snapshots for cases 47, 149, 154 it was found that as the influx approached this threshold ($0.02\mathrm{m}^2/\mathrm{s}$) the depth became shallower, and more instabilities arose. In the absence of inflow of water in the tank, the instabilities naturally disappeared with the relative errors between the numerical and measured depths at their minimum.

Even though the finite-volume method handled waves of increasing steepness successfully, it also provided larger errors between the numerical and the measured depths. Overestimated reflection highly contributed to this increasing errors. However, as this effect was uniform for all examined cases, the source of the error may originate from the model.

4.4.4 Effect of imposed depth data on wave speeds

Recalling Sections 3.2.1, 4.2 and B.2, the boundary conditions were implemented in the finite-volume method by creating ghost cells, which in turn helped determine S_L, S_R and ultimately led to the computation of the HLL flux at the boundaries $x = 0, L$. At the left boundary ($x = 0$) in particular, the $Q(t), h(0,t)$ measurements were imposed on the left ghost cell, as sketched below.

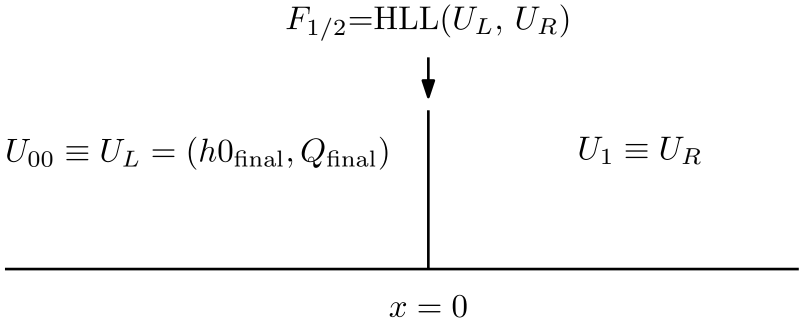
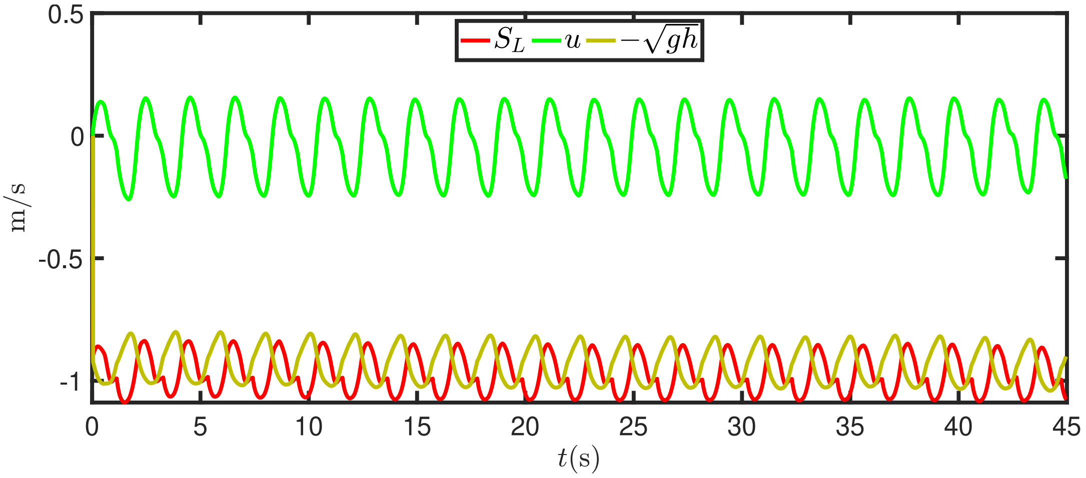
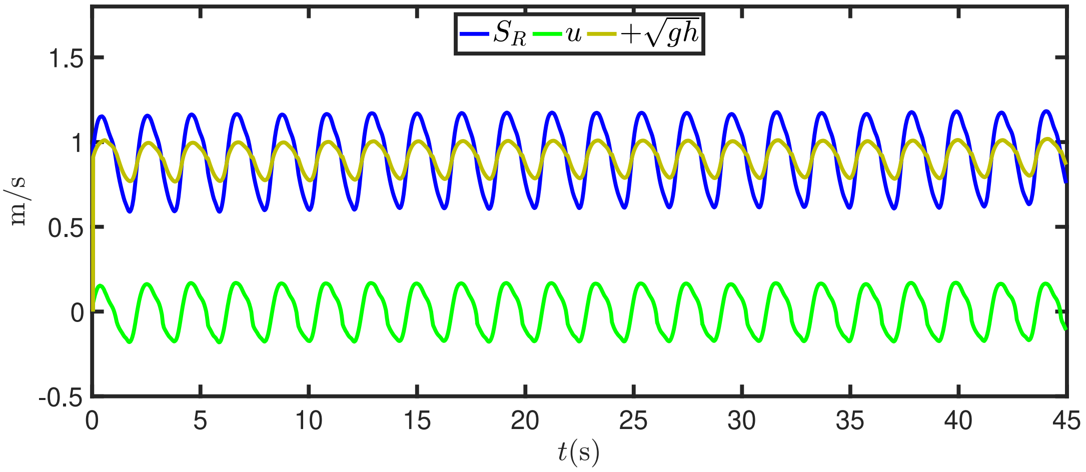


Figure 4.17: Left state U_L and right state U_R surrounding the left boundary at $x = 0$. The flux at the boundary $F_{1/2}(U_L, U_R)$, is determined from the HLL approximate Riemann solver (see section B.2) by utilising the information from the two states.

In order for $Q_{\text{final}}, h0_{\text{final}}$ to be imposed at the left wall, the HLL Riemann-type solver has to use the information originating from the left ghost cell. Hence the information from the left wall is either fully used, if the wave is supersonic on the right ($S_L > 0$), or partially if the wave is subsonic ($S_L \leq 0 \leq S_R$). In the case of a supersonic wave on the left ($S_R < 0$), the measurements are not used. Consequently, checking the values of S_L, S_R can provide further insight on the types of waves created and the wave regime(s) prevalent during simulation two for each case. The minimum and maximum wave speeds S_L, S_R, originating from the cells that surround the left boundary $x = 0$, have been computed for case 154 and are illustrated in the following figure.



(a) Minimum wave speed $S_L = \min\left(u_L - \sqrt{gh_L}, u_R - \sqrt{gh_R}\right)$ over time.



(b) Maximum wave speed $S_R = \max\left(u_L + \sqrt{gh_L}, u_R + \sqrt{gh_R}\right)$ over time.

Figure 4.18: Minimum and maximum wave speeds S_L, S_R computed for case 154. It can be seen that throughout the experiment the flow is subsonic/subcritical ($S_L \leq 0 \leq S_R$), as Fr < 1.

It can be seen that the values belong in the range of $S_L \leq 0 \leq S_R$, where the flow is subsonic. This range of values thus confirms that the measurements are partly used in the HLL flux computation. This condition therefore explains why the measured depth was not always picked up by the FV method, hence resulting to errors in the order of magnitude of O(-6)-O(-4) reported in earlier sections. The wave speeds for cases 149, 165 also indicate a subsonic flow and are found in Appendix D.4.

4.5 Conclusions

The numerical models, presented in Chapter 3, were validated against experimental data, obtained using image-analysis techniques to videos of the experiments in Chapter 2. The validation of the models involved processing the data, modifying the boundary conditions and gradually imposing the experimental data to the numerical models.

Data processing involved smoothing the data to discard troublesome peaks and fitting sinusoidal functions to the (smoothed) influx data. The boundary conditions were modified, accordingly for each numerical method, to accept the measurements as an input. Two simulations were used to gradually impose the measurements to the numerical models; simulation one and simulation two. Fitted functions for the influx were imposed in sim1, which was quickly discarded as it led to inferior results overestimating the waves' steepness. Sim2 ensued with the smoothed data for the influx and the depth at $x = 0$ imposed to the numerical models.

It was found that for sim2 the finite-element method became highly unstable for waves with an influx amplitude reaching $0.02\ \mathrm{m}^2/\mathrm{s}$. It is thus suggested that the finite-element method is either used as is with weaker forcing or for surface tension to be included in the model to offset nonlinearities by introducing dispersion. The finite-volume method, in contrast to the finite-element method, performed better in sim2 for waves of various steepness and forcing magnitudes. This is directly linked to its inherent ability to handle discontinuities arising from steep waves. The HLL flux partly used the prescribed boundary data, resulting in subsonic flow and small differences between the numerical and the measured depth at the left wall in the order of O(-6) to O(-4) decimal points. In addition, the bore profile of the numerical wave was different to the experimental one. This is attributed either to the flow's changed surface tension as a result of the addition of the food dye. Even though reflection has been overestimated causing the largest errors for the finite-volume method, one can argue that its negative influence has been uniform throughout all cases thus indicating that the source of the error may lay elsewhere. A missing term, omitted from the width- and depth-averaging during the shallow-water derivation, may be the reason behind the increasing error between the numerical and the measured depth for waves of increasing steepness and inflow amplitude.

Chapter 5

Experimental tracking and analysis of the wave-beach dynamics

In this chapter the focus was on beach formation as a result of the interaction of the incoming waves with the bottom topography. An investigation on this interaction was carried out through water-bed experiments. Data were obtained from these experiments, with the help of an extended image-analysis algorithm, and were further analysed. In particular, they were used to understand the effect of wave period and water depth on the time and shape of the beach at the final quasi-equilibrium, *i.e.* the state at which the beach profile is almost formed but is still subjected to the influence of the incident waves with the dynamics of the two, *i.e.* the waves and the bed, in balance. The data were also used to relate the formed beaches to real-life beaches by computing the angles of repose and comparing them against values reported in the literature as well as identifying the breaker types observed in the experiments.

The summary of the chapter is as follows: Section 5.1 presents the water-bed experiments carried out as well as the extension of the image-analysis algorithm, initially presented in Chapter 2, aimed to track the water and the bed simultaneously. In Section 5.2 the obtained experimental data are further analysed to understand beach formation. Finally the conclusions of the chapter ensue in Section 5.3.

5.1 Experiments and tracking the waves and bed

5.1.1 The Hele-Shaw beach set-up

The addition of aluminium oxide particles of diameter 1.75mm to the Hele-Shaw cell was aimed to accommodate the study of the coupled water-wave and bed-form dynamics; an interaction leading to a laboratory and mathematical modelling proxi of gravel beach formation and destruction. The diameter of the particles used corresponds to gravel bed when the tank is upscaled to real beach sizes ($42\text{m} \times 1.5\text{m} \times 1.4\text{m}$). As the particles have a largely similar diameter, enforced by sieving them before each experiment, the bed behaves as a purely gravel beach. Consequently during the experiments, bedload sediment transport was solely present.

Sieving was undertaken the day before each experiment using a 1.7mm and a 2mm sieve, respectively. This process ensured that the particles in the tank would be in a set diameter range, consequently ensuring similar sediment grading and minimising the number of particles trapped between the glass plates. On the day of the experiment, the particles would be gradually poured into the empty tank so as to form a uniform bed. After a uniform horizontal profile of the dry bed had been reached by levelling the sediment bed to the required initial thickness, which was around 4cm (see Figure 5.13a), the wave pumps would then allow water to enter the cell. In this way, all experiments had the same initial condition for the bed and variations in the wave frequency and wave duration could be further investigated, variables identified as important to the development of the beach profile by Powell [75].

The addition of the particles increased the duration of the experiments as a result of the longer timescales required for the formation of a quasi-steady beach shape, *i.e.* a final shape formed as a result of the wave-bed system reaching an equilibrium, under a set wave action established by the periodic action of the pumps. It should be noted that in this thesis, the terms quasi-steady state and final quasi-equilibrium both refer to acquiring a fairly constant final beach profile, with the wave-bed dynamics in balance when viewed over a few wave periods. While wave-only experiments lasted around 30-50s, bed evolution to a quasi-steady final bed form took place in matters of minutes to an hour. These longer timescales, combined with the porous nature of the particles, somewhat restricted the cases investigated experimentally from a day-to-day basis; once the particles were wet the experiment could not stop and only a particular case was to be investigated. To this end, questions arose such as: should the experimental set-up change in light of the particles and which cases could be of interest to investigate when running the experiments?

Having two elements present in the cell (water and particles), affected the analysis too as it created the need for a reliable and adaptive tracking algorithm that could deal with the extra complexity of two interfaces: the water-air and water-bed ones. It was important to determine whether the same algorithm could track the water and the bed efficiently, given the vast number of snapshots obtained by capturing fifty of them every second. The above aspects will be discussed in Sections 5.1.2 to 5.1.4.

5.1.2 Framework for the experiments

The aim of this study was to obtain experimental data for the depth of the water and the moving bed. These data could then be used to further understand beach formation and to accurately translate the observed physical phenomena into valid experimental data.

Investigating the effects of wave frequency, depth and wave height on the final shape and slope of each formed beach through the experiments, determined the range of frequencies and wave heights that could be achieved in this set-up. In order for these effects to become apparent, the initial bed depth and shape was kept the same as much as possible by using a ruler. The set-up (tank, lights, dyed water) was initially kept the same and it was thus left to the tracking to determine whether any changes would be required to improve the physical conditions of the recorded experiments. A summary of the experiments carried out is can be found in Table 5.1.

Tracking the water and the moving bed is a concept different from the particle tracking velocimetry (PTV) and particle image velocimetry methods (PIV). PIV and PTV in principle determine the flow's velocity by measuring the distance travelled by tiny buoyant particles which are tracked from one image to the next. These particles therefore work as markers of the fluid under a Lagrangian or Eulerian framework, respectively [8]. Experimentally, these methods require a laser and a high-speed camera which illuminate and track the movement of these tiny particles.

Table 5.1: Experiments performed investigating the water-bed interaction, in view of various frequencies and wave depths. Wave frequencies were in the range of 0.3 to 0.7 Hz. Maximum depth values have been provided by computing the mean depth for each experiment (see Figure 5.14). Duration denotes the length of time that the recording of each experiment lasted and it is larger or equal to the time it took for the wave-bed dynamics in each experiment to reach a quasi-equilibrium state, *i.e.* the time required for the uniform horizontal bed to transition to a formed beach shape under the continuous wave influence.

Videos	f(Hz)	**Water depth**(max)	**Duration**(min)	**Figures**
219-223	0.41	Deep (7cm)	70:05	5.7, 5.24, 5.25
232-237	0.67	Medium (5.7cm)	103:05	5.8 , 5.26, 5.27
248-256	0.5	Shallow (4.9cm)	81:53	5.9, 5.28, 5.29

Though popular within the scientific granular flow community, these methods were neither available nor suitable for the present experiments. The interest was on tracking the moving bed as a whole instead of tracking individual particles. In addition, the corresponding timescales for beach formation were relatively long (minutes to an hour) and were therefore restrictive for recording with high-speed cameras, particularly as they provide optimal image resolution but sacrifice the duration of the recordings to just a couple of tens of seconds.

ImageJ/Fiji is a popular image processing software, based on Java, widely used because of the variety of imaging operations that it offers; from visualising, processing and analysing images to tracking particles [18, 86]. Unfortunately, the large number of images that needed to be processed and analysed required a lot of memory thus rendering the use of ImageJ impossible. Alternatively, a robust algorithm was required to work recursively in a precise manner so as to analyse each image and extract the information in question.

5.1.3 Tracking the bed

The first step towards tracking the moving bed alone was to investigate to what extent the current image-analysis algorithm was adequate. The water and the background colour were in shades of blue, an effect accomplished by the use of the food dye and appropriate aperture and shutter speed values. The colour uniformity of the water and the background translated into similar threshold values. As a result, these regions could be ignored simultaneously, provided an appropriate colour channel and threshold value could be determined. Indeed, taking the blue channel removed the blue from each snapshot as seen in Figure 5.1.

(a) Original image.

(b) Blue channel.

Figure 5.1: Comparison between the original (segmented) image with the water and the bed (left panel) and its corresponding blue colour channel (right panel). The water has been ignored successfully and only the bed is depicted in the image.

It can be seen in Figure 5.1 that the use of the blue colour channel allowed the elimination of the water and the background colour successfully and only the sediments have remained. The binary version of Figure 5.1b, illustrated in Figure 5.2, depicts the bed area that remained after the colour channel selection and thresholding had been completed. The same area was then tracked by the extended image-analysis algorithm, resulting in Figure 5.3.

Some isolated particles, located at the top of the bed, can be identified in Figures 5.1 and 5.2. These particles did not remain in the image after the application of the colour channel. It is believed that their tiny size, combined with the fact that they were further away from the rest of the bed, led to the filter identifying them as part of the water rather than a part of the bed. Contrastingly, in highly

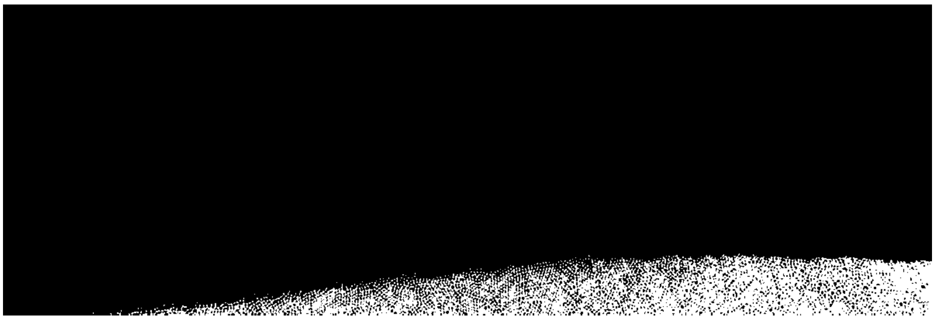
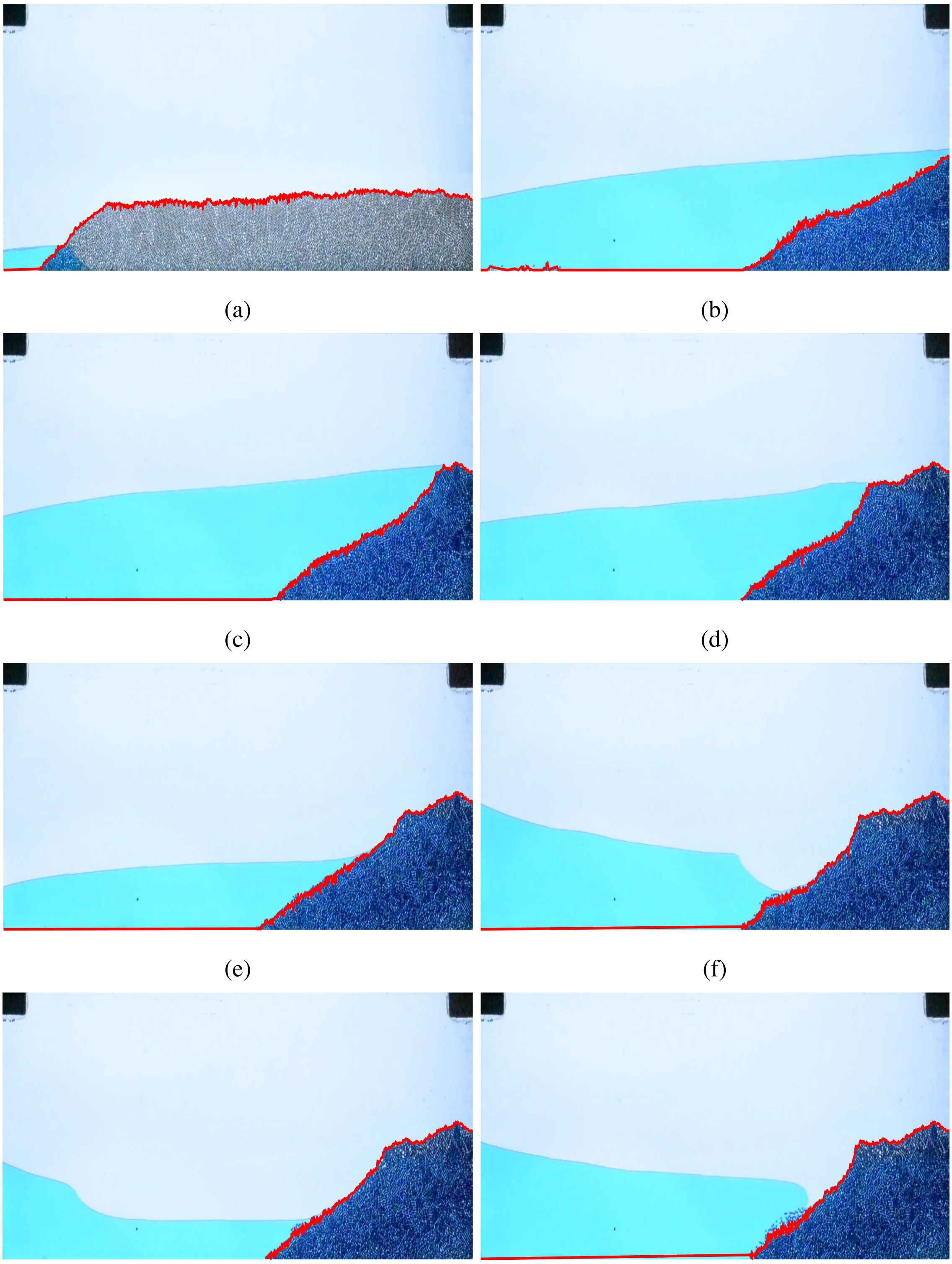


Figure 5.2: Binary bed image obtained after the intensity threshold has been applied.

particle-populated areas the colour filter correctly neglected the water and the background effects.

Figure 5.3: Tracked bed (red) overlaid against the initial snapshot.

The green channel was another option when tracking the moving bed individually; one that yielded slightly better results. Keeping in mind that the final goal was to track the bed and the water as accurately as possible, the blue and green channels were selected accordingly during each experiment with respect to the one that provided optimal tracking for the bed without interfering with the water tracking. Ultimately, the use of blue and green channels was done in a ratio of two to one, respectively. Further snapshots, overlaid against the bed tracking results, can be found in Figure 5.4.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 5.4: Tracked bed overlaid with snapshots from experiment 219-223 captured at $t = 0\text{min}\ 2\text{s}, 10\text{min}\ 2\text{s}, 20\text{min}\ 2\text{s}, 40\text{min}\ 2\text{s}, 45\text{min}\ 2\text{s}, 50\text{min}\ 2\text{s}, 60\text{min}\ 2\text{s}, 70\text{min}\ 2\text{s}$.

5.1.4 Tracking the bed and the water in unison

After tracking the moving bed with the same algorithm had been accomplished, the next step was to investigate if the water and the bed could be tracked simultaneously. The main difficulty stemmed from the fact that the particles were underwater; tracking the water was at risk of being intercepted by the presence of the particles. Consequently, only an appropriate choice for the water colour channel would resolve this issue and keep the water and the particle areas seperate.

Indeed instead of using the red channel to track the water, as done in the cases without the bed, a different strategy was preferred for the colour channels. The red-channel image (C_1) was subtracted from the original image thus creating image $CC = C - C_1$. Lastly, the green channel of image CC was selected, $CC(:,:,2)$, as the one to be further processed. An example of tracking the water and the bed simultaneously can be visualised in Figure 5.5.

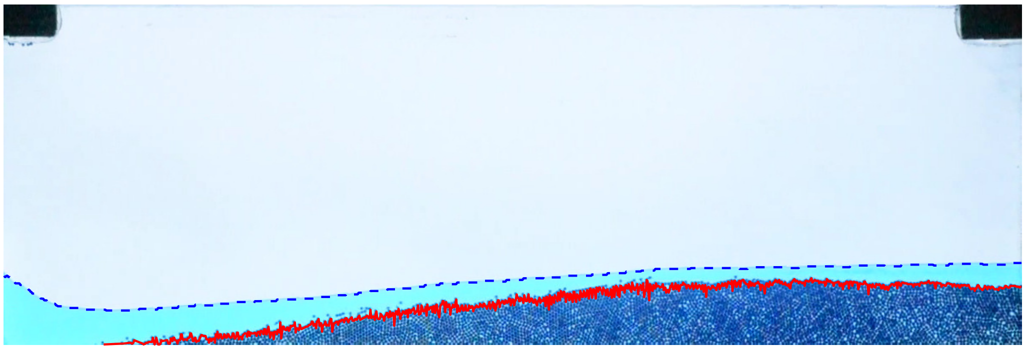


Figure 5.5: Tracking the water (blue) and the bed (red) simultaneously. In this image, the bed is fully submerged under water.

The image processing algorithm was then used to track the water and the bed for all experiments. Applying the tracking algorithm to various topographies was the best way to test the effectiveness and accuracy of the colour-detection method. In particular, tracking the water around a submerged and emerged topography was of interest. A summary of the main steps of the adapted algorithm has been provided in the following figure.

The tracked experimental images for all cases have been visualised in the figures below.

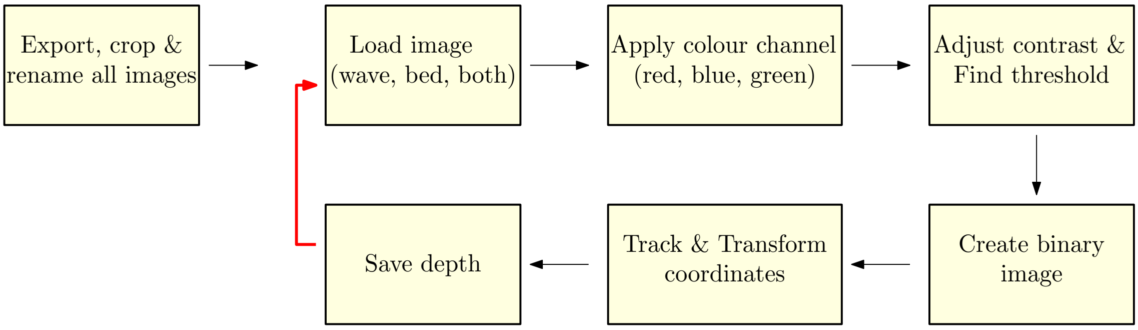
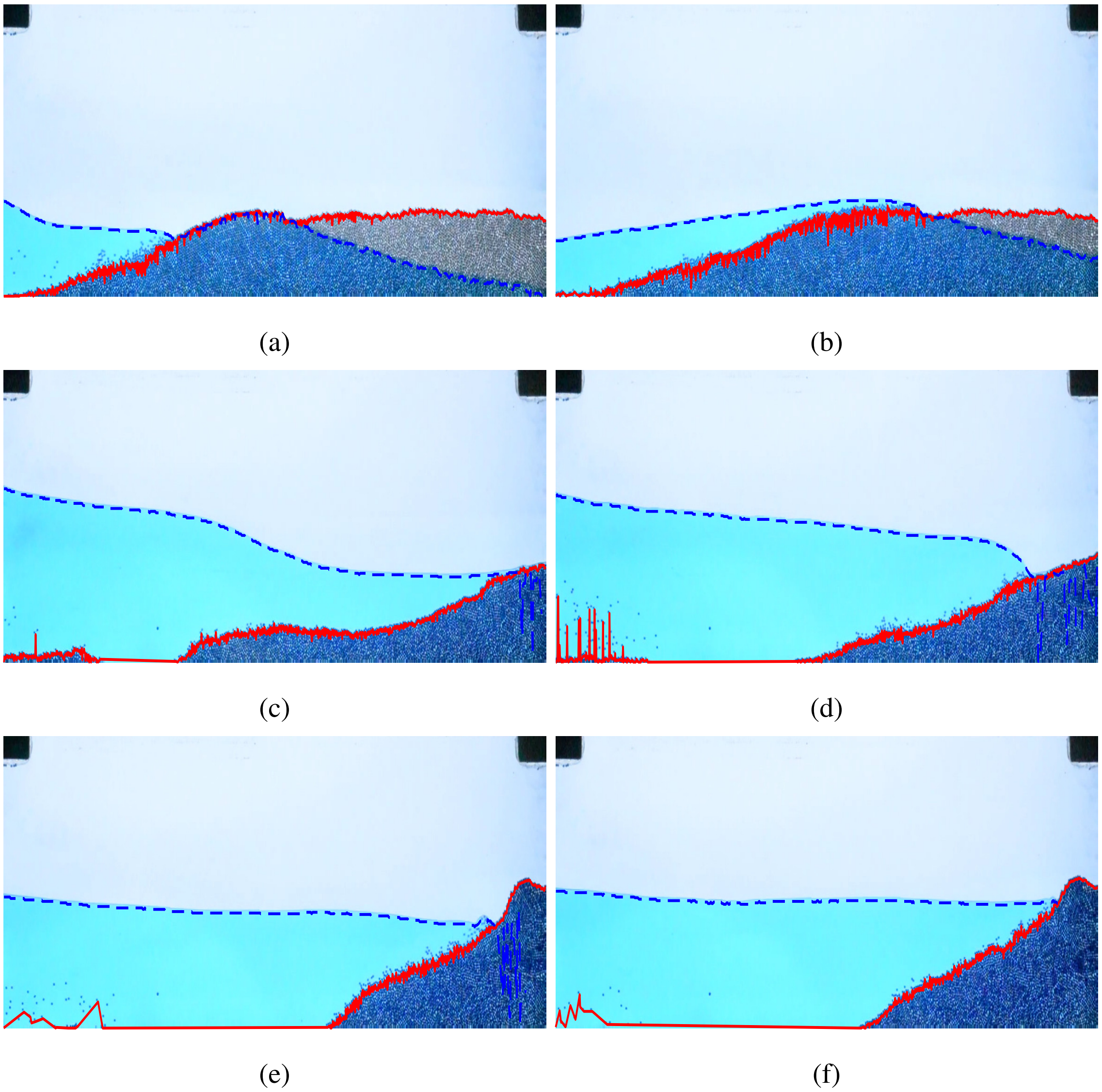


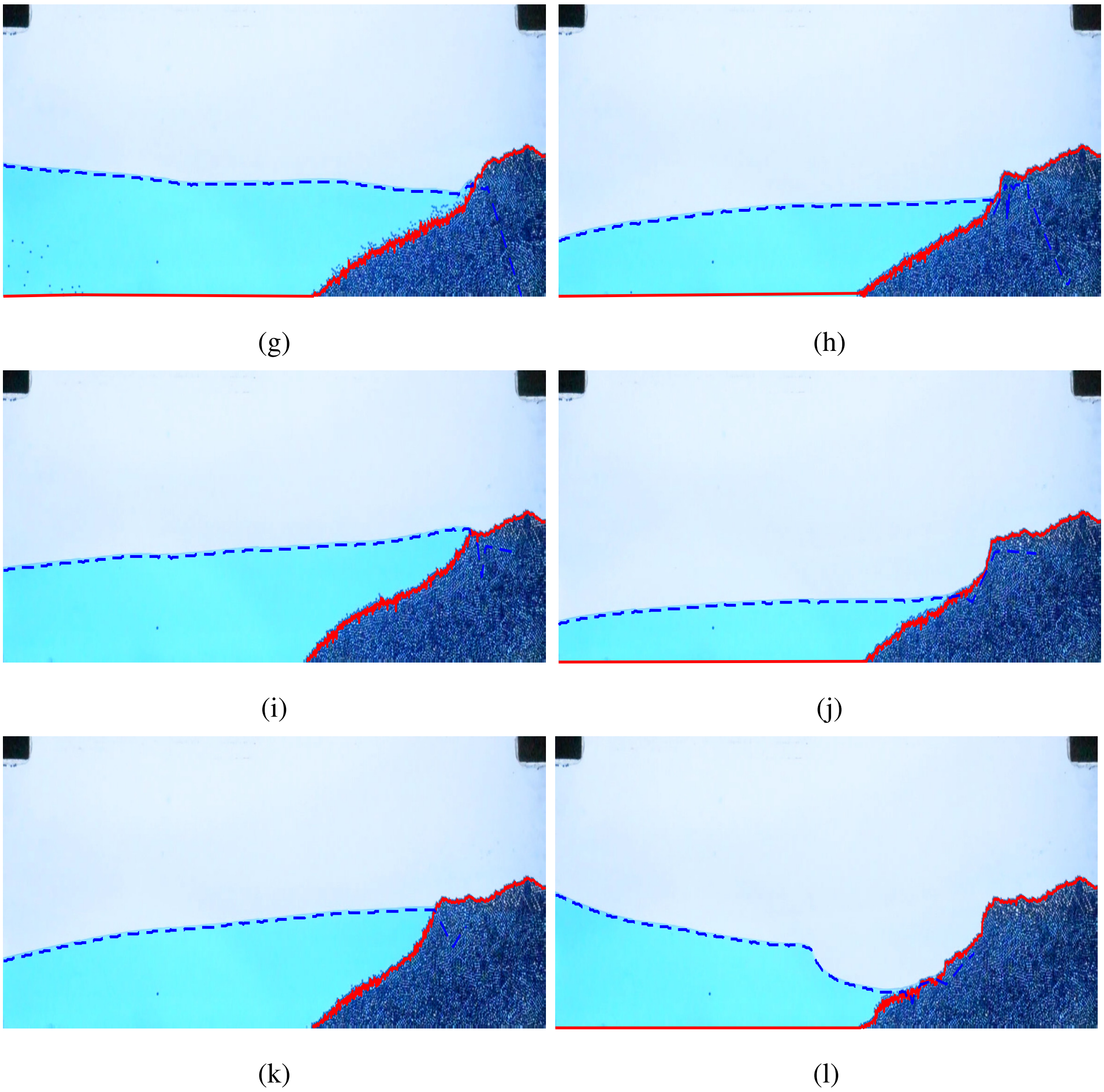
Figure 5.6: Summary of the colour-channel-tracking algorithm. One image is processed for the water and one for the bed simultaneously. Then the image-analysis stage translates the height of the water and bed from images into data, same as before.


(a) (b)

(c) (d)

(e) (f)

Figure 5.7: Tracking the water and bed for videos 219-223, at $t = 0\text{min}\,30\text{s}, 43\text{s}, 5\text{min}, 7\text{min}\,41\text{s}, 11\text{min}\,20\text{s}, 18\text{min}\,20\text{s}$.

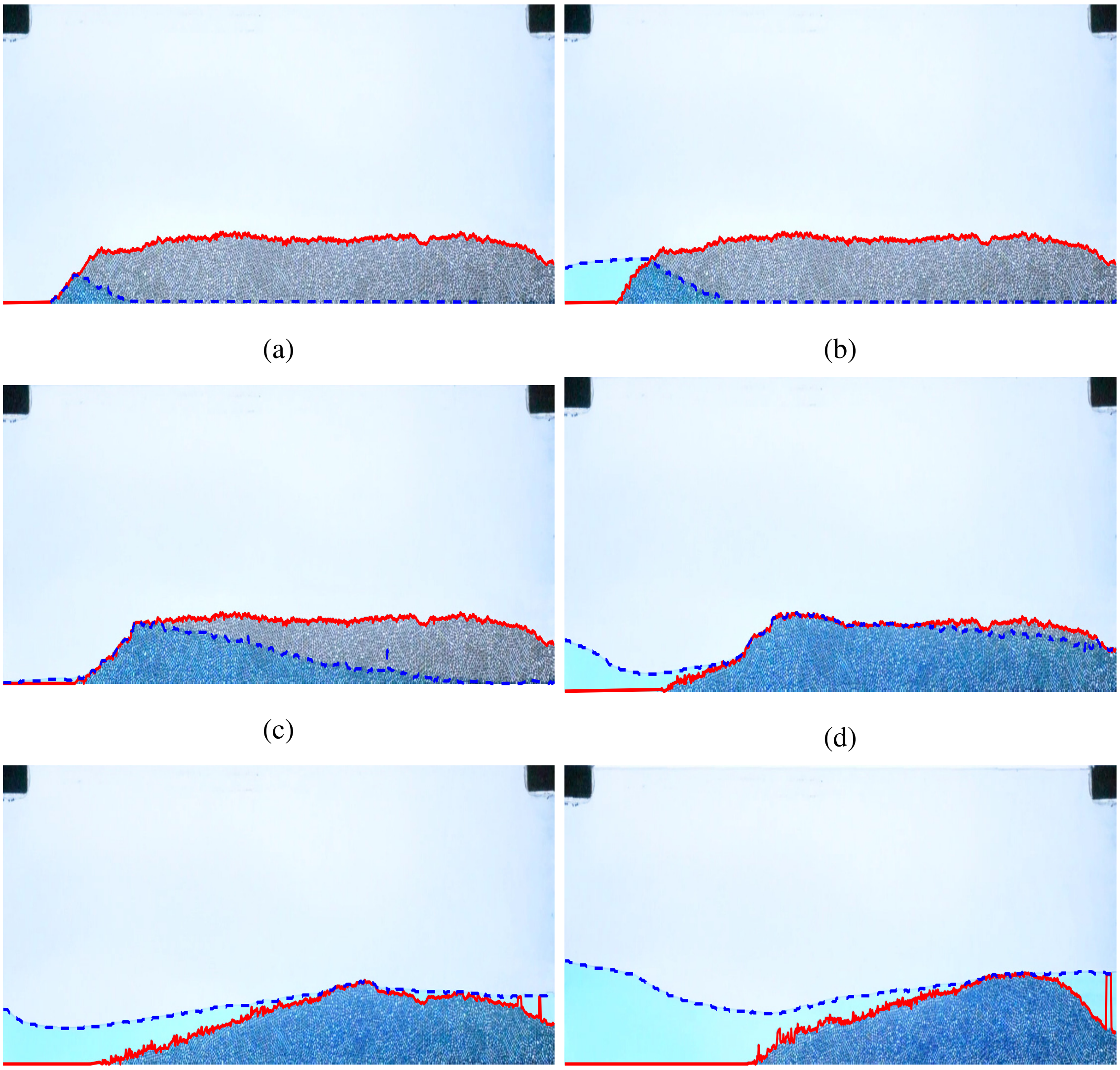


(g) (h)

(i) (j)

(k) (l)

Figure 5.7: Tracking the water and bed for videos 219-223, at $t =$ 22min 12s, 30min, 35min 20s, 50min 30s, 68min 44s, 69min 48s. A video of the above tracking can be found here: `https://youtu.be/UYhgxUMKBQ0`.

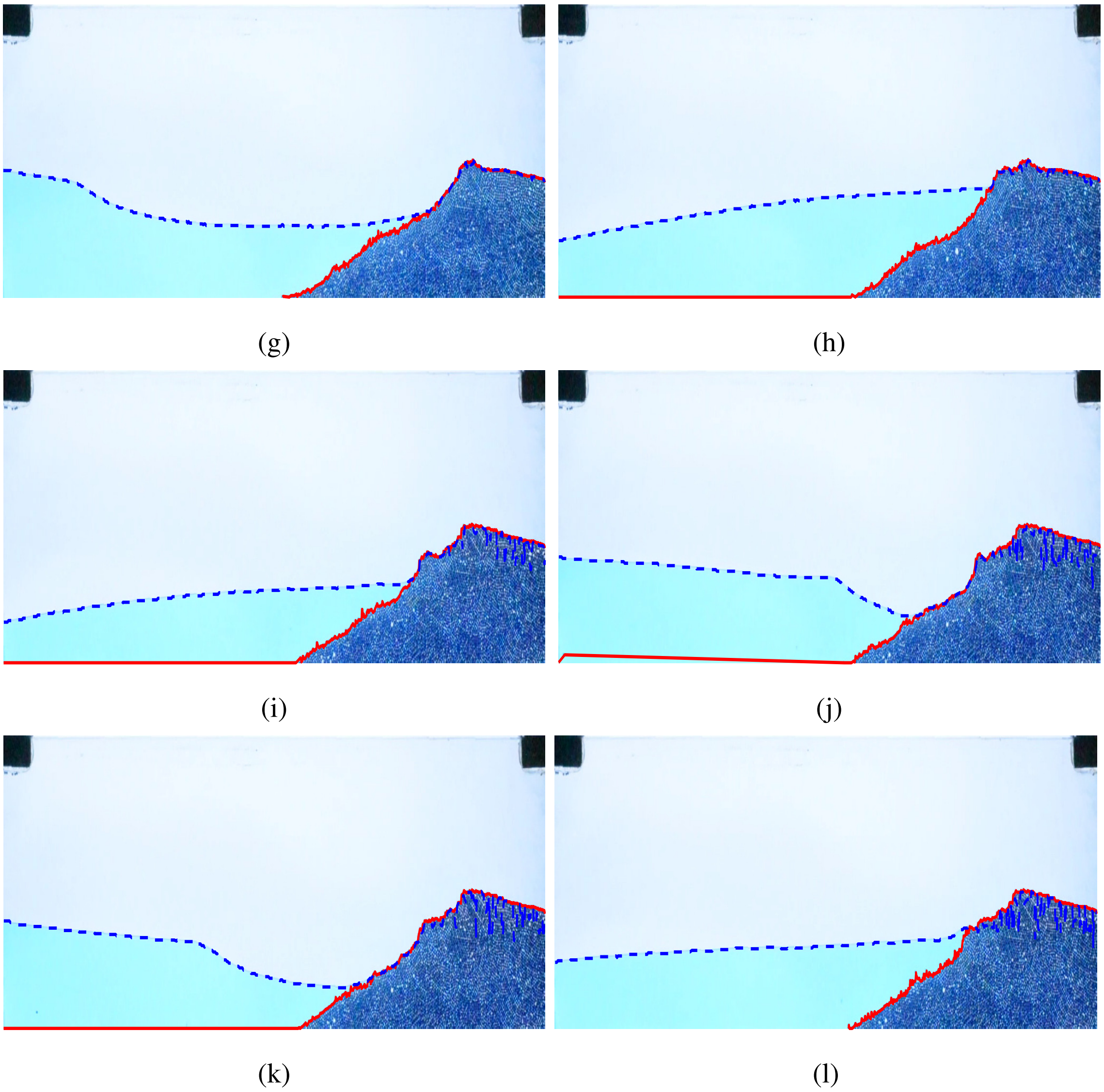


(a) (b)

(c) (d)

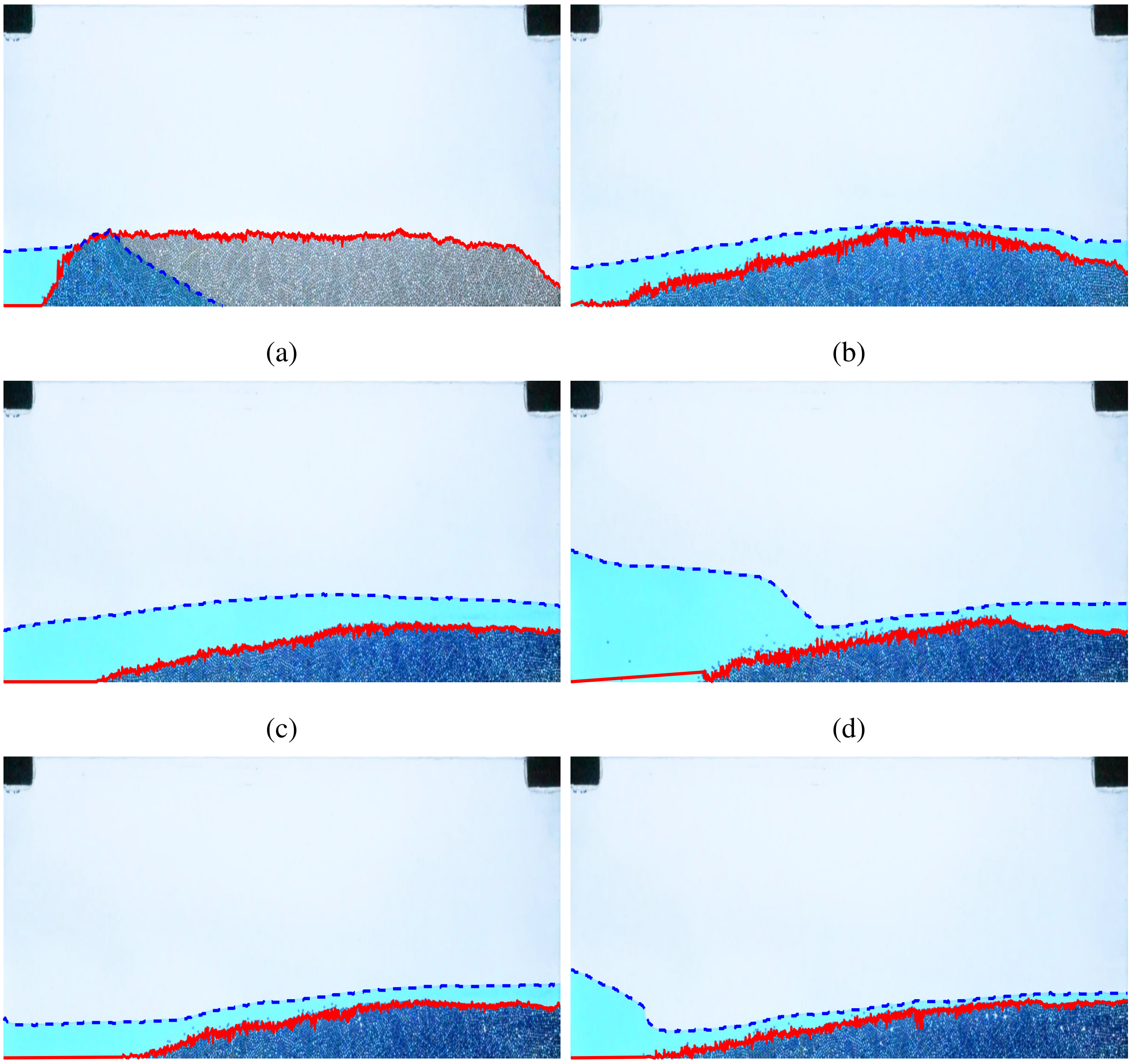
(e) (f)

Figure 5.8: Tracking the water and bed for videos 232-237, at $t = 17\text{s}, 19\text{s}, 45\text{s}, 1\text{min}\ 43\text{s}, 2\text{min}\ 45\text{s}, 7\text{min}\ 25\text{s}$.



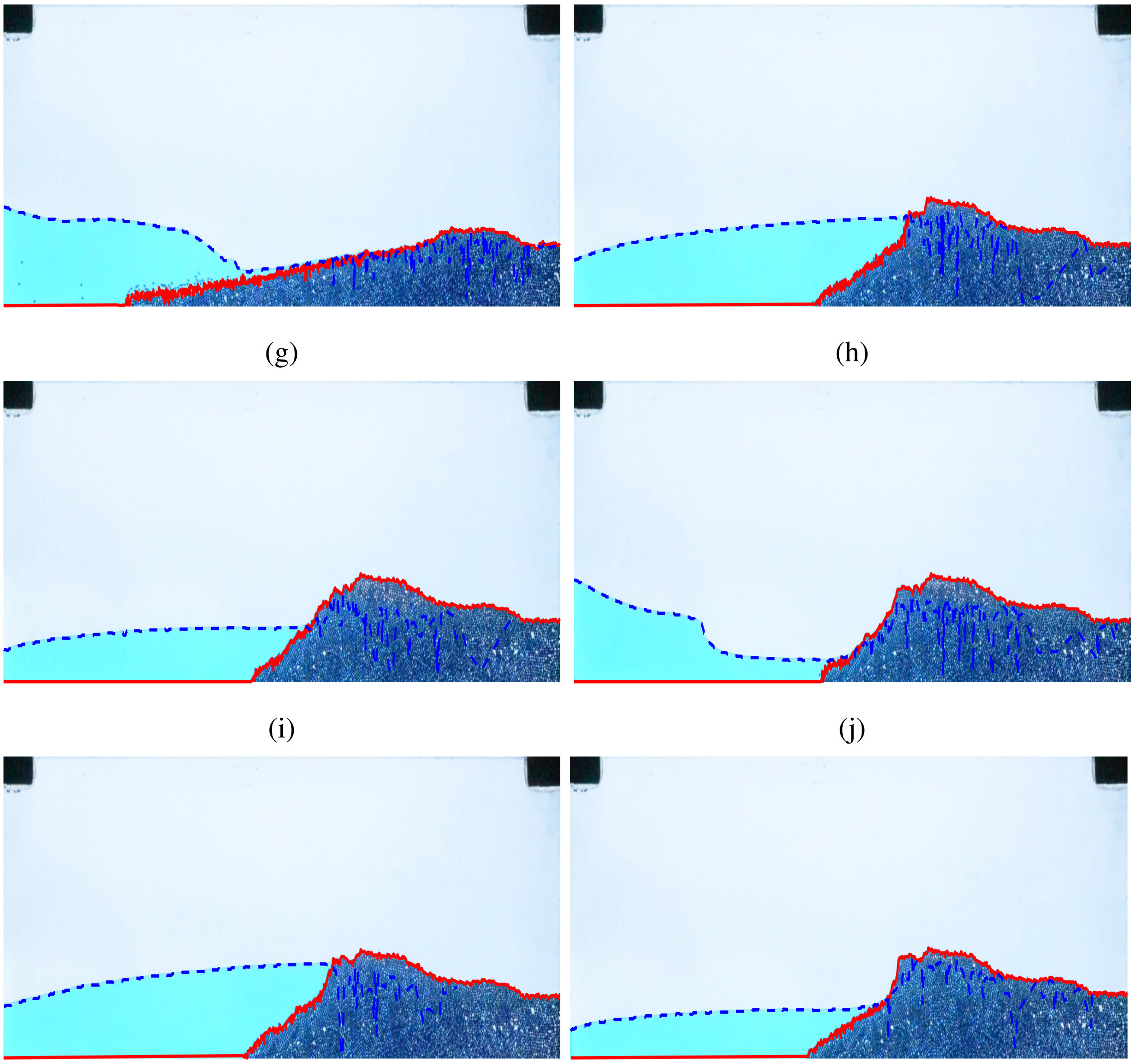
(g) (h) (i) (j) (k) (l)

Figure 5.8: Tracking the water and bed for videos 232-237, at $t =$ 23min 24s, 50min 6s, 70min 40s, 83min 24s, 101min 40s, 103min 5s. A video of the above tracking can be found here: `https://www.youtube.com/watch?v=1ov8JdjNGzA&t=51s`.



(a) (b) (c) (d) (e) (f)

Figure 5.9: Tracking the water and bed for videos 248-256, at $t = 16\text{s}, 50\text{s}, 10\text{min}\ 20\text{s}, 20\text{min}, 23\text{min}\ 27\text{s}, 30\text{min}\ 4\text{s}$.



(g) (h)

(i) (j)

(k) (l)

Figure 5.9: Tracking the water and bed for videos 248-256, at $t =$ 40min, 50min 40s, 60min, 62min 28s, 81min 20s, 82min 34s. A video of the above tracking can be found here: `https://youtu.be/uza-NCZZV_Q`.

The tracking of the water and the bed was carried out successfully, as evidenced in the previous figures. One of the issues that became apparent through the various experiments was the fact that the porous nature of the bed interfered with the tracking of the water. In the cases where the water depth was lower than the bed height of the wet beach, the part of the code tracking the water often picked up the bed height instead of the water depth. This meant that the code could not distinguish the saturated areas from those actually in contact with the water, which resulted in the saturated particles contaminating the water measurements. Some examples of this effect can be seen in Figures 5.8j-5.8l and 5.9g-5.9l.

Using a ruler and knowing the scaling of the tank from snapshots to reality, it was of interest to quantify potential tracking errors in the water and the bed depth by manually inspecting some snapshots, as they were the only way to visually compare the real with the tracked depth over the course of each experiment. The error in the tracking of the water line was more visible when tracking the water and the bed in unison, and was around 1mm. However, the tracking error for the bed height was found to be between 1.5-2mm, the same size as a single particle, only in some regions of the snapshots. This is believed to be due to the colour channels, which ‘misinterpreted’ the particles at the top of the bed for water thus ignoring them. In addition, the reduced resolution of the cropped snapshots resulted in less pixels per unit distance, ultimately providing a less accurate representation of the bed line. In order to visualise the effect of the colour channels on the tracking, one can compare Figures 5.7-5.9 with their binary version that is ultimately tracked during the image-analysis stage in Figures E.22-E.24, in Appendix E.1. Unfortunately the computation of the area flux error was not possible; neither the computation would have been carried out effectively, given the large number of snapshots, nor would have yielded accurate results as all depth and errors would have to be computed by hand while accounting for scaling errors between the snapshots and reality.

The novelty of tracking the water and the moving bed with the colour channel method lies in its simplicity while being relatively sufficient. Not only is the method easy to understand and replicate, but both its computational time and effort are greatly reduced since the same software and the same piece of algorithm are used to simultaneously perform two similar, yet different, tasks.

5.2 Results

In this section, the obtained data were used to further understand gravel beach formation by relating it to variables such as the wave period and the water depth. The bed angles were also computed for each case with the angles of repose compared with angles of repose for real gravel beaches. Lastly, wave-breaking was investigated with the types of breakers found in the experiments recorded and identified.

5.2.1 Final-time bed profiles and angles

The swash zone is the portion of the sloping beach that gets alternately wet and dry during the wave action. The wave action comprises of the incident waves that reach the shore, run up the beach slope (uprush), infiltrate into it and run down again (backwash) being less powerful than before. The amount of fluid infiltrating the beach, determines not only the strength of the backwash but also the final profile of the beach [43].

Porosity and permeability are physical properties responsible for the passage of fluids through a mass of solid. Porosity n measures how much of the total volume of a material V_t is taken up by void space V_v and is defined in [39] as:

$$n = \frac{V_v}{V_t}. \tag{5.1}$$

Porosity is affected by sediment properties such as the shape and the degree of sorting. For this type of particles, porosity is about 0.55 [90].

Hydraulic conductivity/permeability measures how easily the flow infiltrates a porous material under gravity [39, 25]. Permeability, depends on the properties of the porous material and the fluid that percolates through it and has units of velocity. This dependence is also revealed through the following formula:

$$K = k_i \frac{\gamma}{\mu}, \tag{5.2}$$

with $\gamma = \rho g$ the specific weight of the fluid, μ the viscosity of the fluid and $k_i = Cd^2$ the specific permeability; the value of C is determined solely by the porous medium properties such as the pore diameter d and the shape of the pore openings of the material [39, 88]. Permeability thus depends on the diameter of the sediments; the higher the grain diameter is the higher the infiltration will be. Consequently, sediments with higher diameter, such as gravel, have higher permeability than sand.

The angle of repose θ_r, also known as the angle of internal friction [88], is the steepest slope angle at which the sediments can stay on the bed without sliding down [88, 103]. If the angle of repose is exceeded, then avalanching occurs and sediments flow down the slope. The angle of repose is determined when the bed has reached a final quasi-equilibrium state where its overall shape does not change further under wave action. In general, the steeper the slope the higher the angle of repose. Percolation is a phenomenon highly correlated to the angle of repose; the higher the infiltration of a fluid into the bed, the steeper the bed slope can be. Consequently, the factors that explicitly affect percolation, implicitly affect the angle of repose. These factors are the degree of sediment sorting and the grain size as well as the angularity of the sediments.

High and uniform sediment sorting and larger grain sizes, favour the amount of fluid infiltrating the bed [43]. This is why gravel beaches, with no large deviations in sediment sorting, have higher permeability than sand. Inversely, higher sediment roundness translates to less gaps between the particles, *i.e.* decreased porosity [88], consequently leading to decreased percolation and to a decreased angle of repose [20, 85]. There are different methods of measuring the angle of repose of soils such as the tilting box method, the fixed funnel method and the revolving cylinder/drum method to name a few. Unfortunately, not all methods give the same angle of repose [24]. The angle of repose of natural gravel is considered to fall in the range of 25° - 30° [4]. An investigation into the slopes of the beach profiles, created at the end of each recording of the experiment, was carried out. The final bed profiles of all experiments were used to determine the bed angles, and have been plotted against their wave frequency in Figure 5.10.

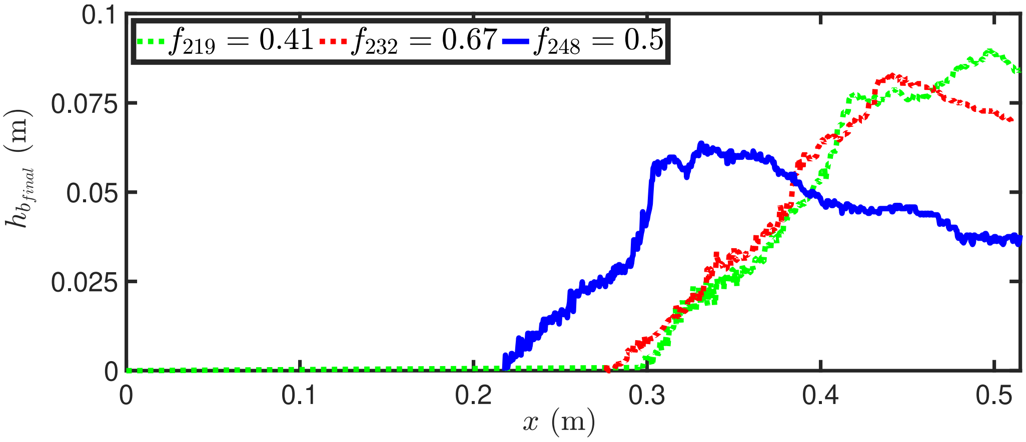


Figure 5.10: Final bed profiles formed as a result of the effect of monochromatic incident waves entering the tank through the left boundary. Each case had a measured frequency of f_{219}=0.41Hz (green), f_{232}=0.67Hz (red) and f_{248}=0.5Hz (blue).

Each bed profile was assumed to consist of two regions, the wet and swash region. The wet region was always submerged under water while waves entered and exited the cell. The region was located between the toe of the bed and the minimum water depth. The swash area, was alternately wetted and dried as the water entered and exited the cell. Consequently, during the experiments the swash zone was located at the part of the beach surrounded by the minimum and maximum water depth in the tank.

Identifying the points that belonged to each region was possible by inspecting the corresponding snapshots of the final quasi-equilibrium states and using the bed coordinates of the tracked images to determine the slope. Essentially, three position vectors were specified for each region. Namely, at the wet region the toe of the bed was denoted with the vector (x_1, y_1), the bed at the minimum water depth with (x_3, y_3), while (x_2, y_2) was a measurement between the two. The same logic applied for the swash region with vectors (x_3, y_3) at the minimum water depth, (x_5, y_5) at the maximum depth, with (x_4, y_4) the measurement between the two. Note that the upper limit of the wet region located at (x_3, y_3) was the lower limit of the swash region. Consequently, for three known position vectors, located at each region, the slope of the bed was calculated as the average of two angles using:

$$\text{slope} = \tan\theta = \frac{\mathrm{y}_2 - \mathrm{y}_1}{\mathrm{x}_2 - \mathrm{x}_1} \text{ and } \frac{\mathrm{y}_3 - \mathrm{y}_2}{\mathrm{x}_3 - \mathrm{x}_2}, \text{ or } \frac{\mathrm{y}_4 - \mathrm{y}_3}{\mathrm{x}_4 - \mathrm{x}_3} \text{ and } \frac{\mathrm{y}_5 - \mathrm{y}_4}{\mathrm{x}_5 - \mathrm{x}_4}. \tag{5.3}$$

Taking the inverse tangent of the slope provided the required angle θ. The final bed angles corresponding to the wet and swash region for each experimental case are visualised in Figure 5.11 and have been summarised in Table 5.2.

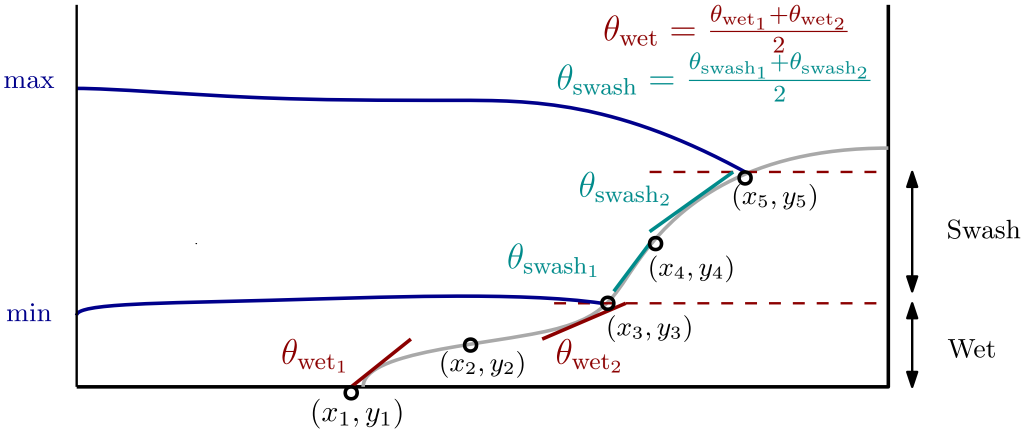


Figure 5.11: Bed profile, consisting of the wet and the swash region. Wet region consists of position vectors $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ and the swash region consists of vectors $(x_3, y_3), (x_4, y_4), (x_5, y_5)$. The vectors then help determine two angles in each region, $\theta_{wet_{1,2}}$ (red) and $\theta_{swash_{1,2}}$ (green). The average of the two determined the angles θ_{wet}, θ_{swash} for each region and each case, summarised in Table 5.2.

It can be observed in Table 5.2 that the swash zone angles are larger than the wet zone angles, cf. [90]. This difference stems from the presence of a reduced-in-magnitude cohesive force acting on the particles, affected by the existence of trapped water between them. This reduced cohesive force between the immersed particles affected their stability thus resulting in smaller bed angles, with the volume of water between the particles known as liquid bridge [5, 24]. In [5], this phenomenon was attributed to the fact that the contact regions between spherical polysterene beads (diameter 0.8±0.2mm) were small hence contributing only a little volume of liquid bridge to the cohesive forces, with the higher fraction of the liquid surrounding the bead area being passive. In contrast to the wet zone, the swash zone was not constantly wet so the inter-particle forces, such as friction,

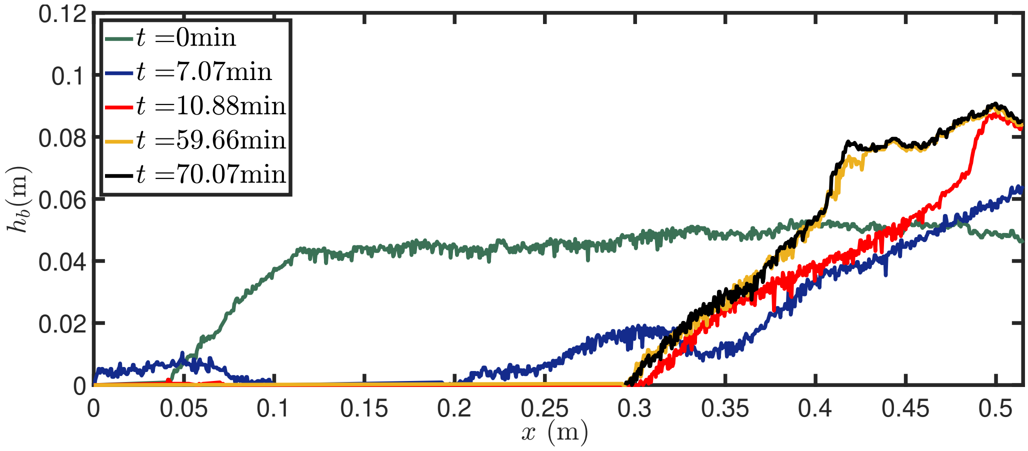
Table 5.2: Final-time bed angles for each experiment. Two sub-regions, wet and swash with corresponding angle values θ_{wet}, θ_{swash} have been determined. For each region, *i.e.* wet and swash, two angles were computed using equation (5.3). Taking the average of the two angles at each region, yielded an improved angle estimate, which was more accurate in the cases of very steep bed slopes. The data figures, indicated below, used for the angles calculation can be found in Appendix E.2.

Videos	$\boldsymbol{\theta_{wet}}$	$\boldsymbol{\theta_{\mathrm{swash}}}$	**Figures**
219-223	29 °	40 °	E.25
232-237	26 °	50 °	E.26
248-256	26 °	40 °	E.27

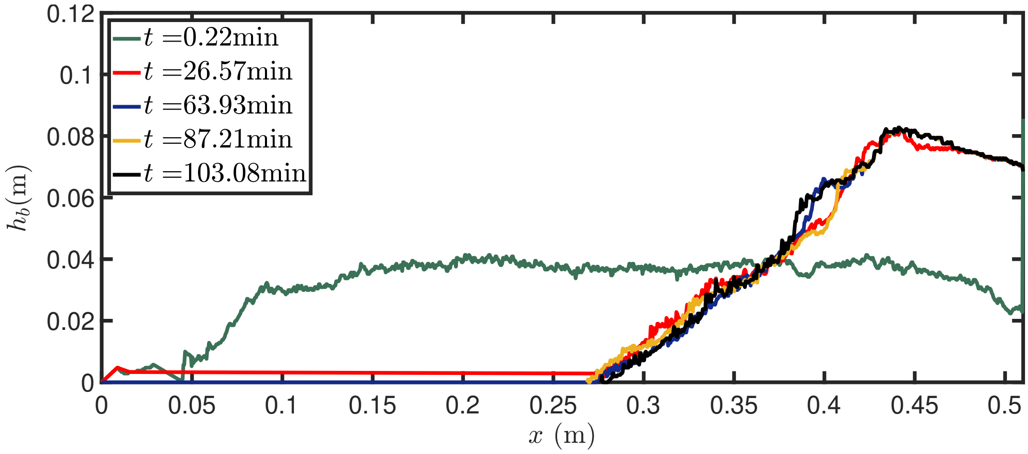
weight, repulsion and cohesion, were stronger thus allowing for a better slope stability and larger angles to be reached. Furthermore, the wet angles were found to be in the natural range of angles discussed earlier (25° - 30°), for all three experiments, with the remaining one being close to the lower limit. Following the discussed effect of the liquid bridges, it is believed that the presence of the water resulted in smaller angles of repose compared to those found for dry gravel beaches. Lastly, no correlation between the above angles and the initial bed conditions has been discovered.

5.2.1.1 Duration of the experiments

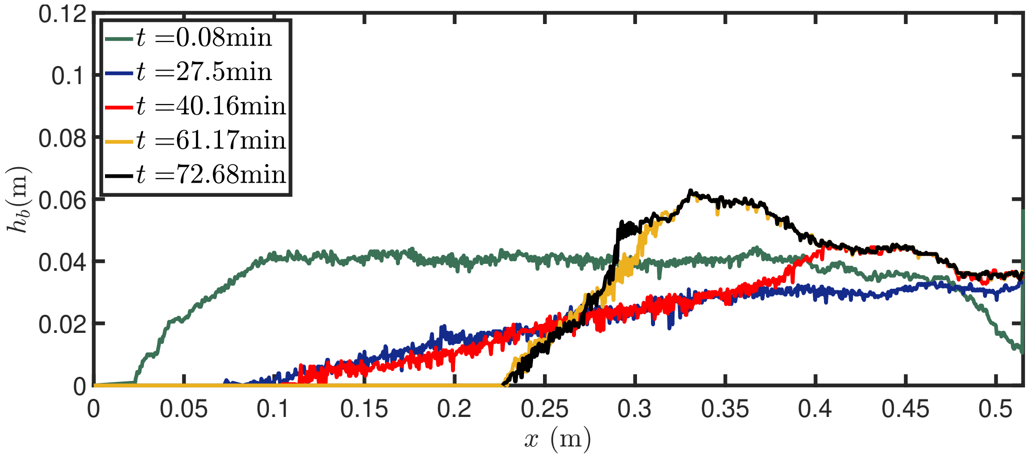
By the end of each experiment, the bed had reached a final quasi-equilibrium profile, with small variations on its slope and shape, as a result of the interaction it had with the incident waves. This final profile was determined by observation and in this thesis has been referred to as duration of the experiment/recording. In order to ensure that the duration of each experiment did not overestimate the time required for the beach to reach a final quasi-equilibrium, plots comparing the final bed profiles were created. Essentially, the time required for the beaches to have a more or less constant shape, under the continuous influence of wave action, was referred to as *proposed duration* (PD). The chosen final bed profile, which did not change further and was selected by observation, was used to determine the proposed duration.



(a) First quasi-equilibrium (red) for case 219 reached at t = 10min 53s. PD: 59min 40s (orange).



(b) First quasi-equilibrium (red) for case 232 reached at t = 26min 34s. In the videos of the experiment the step, located at the beach front, collapsed between t =72min-85min, and was built back up by the end of the recording hence PD: 103min 5s (black).

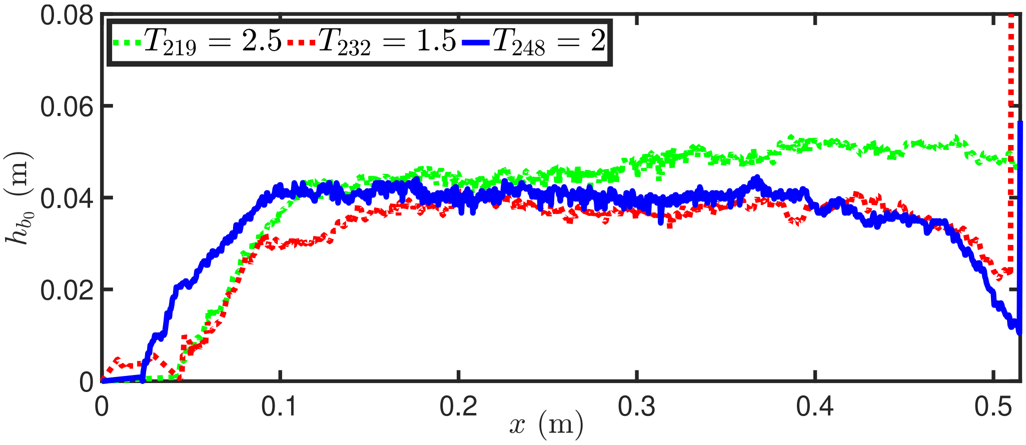


(c) First quasi-equilibrium (red) for case 248 reached at t = 40min 9s. PD: 61min 10s (orange).

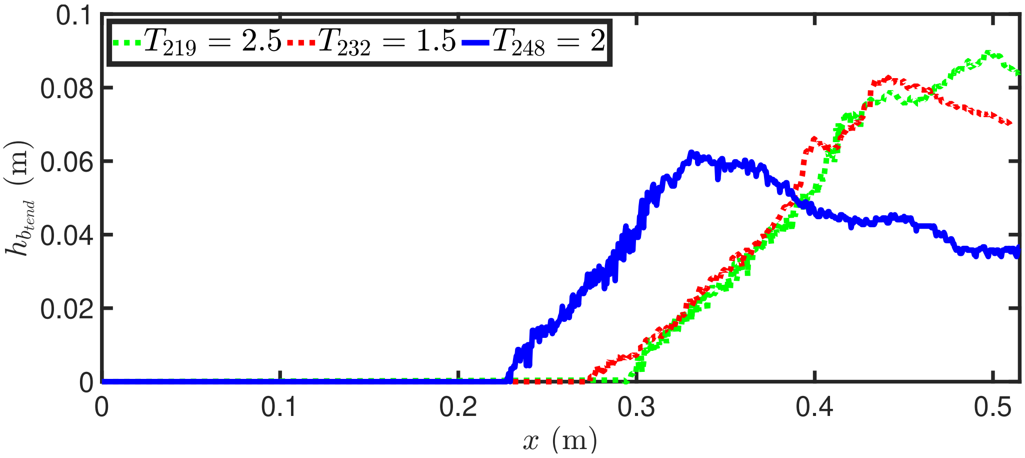
Figure 5.12: Bed evolution over time for cases 219 (top panel), 232 (middle panel) and 248 (bottom panel). The time required for each beach to acquire a quasi-steady shape, *i.e.* final quasi-equilibrium, is calculated in the above panels and has been referred as proposed duration (PD). See also the 3D beach evolution in Figure 5.17.

5.2.1.2 Effect of wave period on the final quasi-equilibrium beach profile

The effects of the wave period and (updated) duration were identified as important by Powell [75] and have been further investigated during these experiments.



(a) Initial bed profiles for experiments. All cases were initialised with a uniform horizontal bed profile of similar depth h_{b_0}. The initial mean bed depths for each case were: 4.36cm (219-223), 3.29cm (232-237) and 3.6cm (248-256).



(b) Final bed profiles of various depths $h_{b_{tend}}$, as a result of the effect of incident waves of various periods. Final bed times are 59min 40s (219), 63min 56s (232) and 61min 10s (248).

Figure 5.13: Comparison of the initial uniform bed profiles (top panel) against the final bed profiles (bottom panel) for each experiment. Wave period for each case: $T_{219} = 2.5$s, $T_{232} = 1.5$s, $T_{248} = 2$s.

A negative correlation between the proposed duration and the wave period can be observed. The larger the wave period, the shorter the duration of the experiments.

5.2.1.3 Effect of water depth on the final bed profiles

Over the course of the experiment a beach was gradually formed and ultimately acquired a quasi-permanent shape at a certain time (proposed duration, see Figure 5.12). The final beach shape in this section is equal to the quasi-permanent beach shape captured at the time that the proposed duration indicated. As the action of waves is continuous in principle can affect the shape of the beach at all times, hence the beach shape can only be almost permanent. However, by the time that the quasi-permanent shape had been reached, the effects of the wave action were minor thus not causing significant changes to the beach profile. As seen in Figure 5.12, the final bed profile was determined by observation.

A correlation between the mean water depth and the proximity of the top of the bed to the right wall was observed. Essentially the deeper the water, in comparison to the bed depth, the narrower in the horizontal and taller in the vertical the beach would be; as observed in Figure 5.13. This argument holds experimentally too if one considers a wave that overturns when it is near the beach. The deeper the wave, the further on the right the sediments move and only when the beach is high enough and near the right wall, then the waves break hence resulting in the top of the beach being very close to the right wall. The validity of the above statement was investigated through the available data. The mean water depth present in the tank during each experiment has been plotted in Figure 5.14.

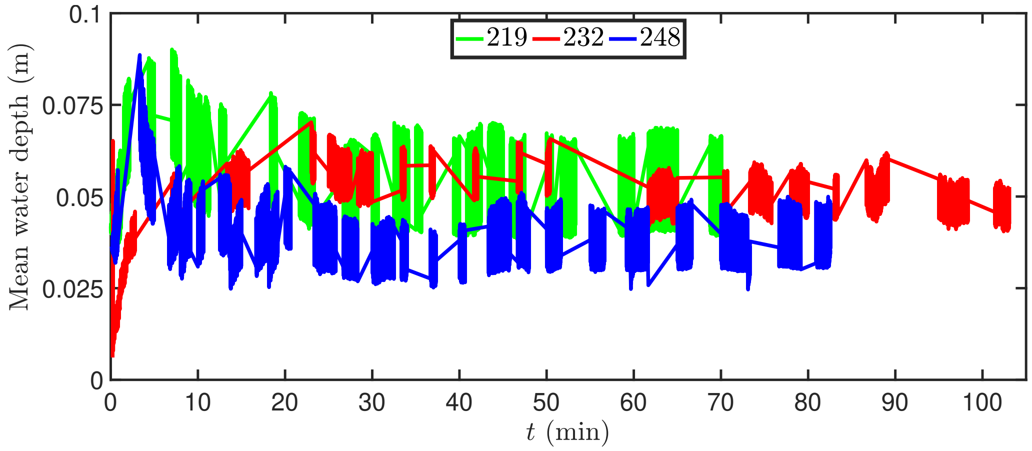


Figure 5.14: Mean water depth in the tank, computed at selected time batches. Case 219 (green) had the highest depth, 232 (red) had a medium water depth, while 248 (blue) had the shallowest depth. As each system starts to stabilise ($t_{219} \simeq$ 10min, $t_{232} \simeq$ 26min, $t_{248} \simeq$ 40min) the minimum and maximum depth values are stabilised too, thus facilitating the recovery of the mean wave heights. Mean wave heights are roughly: 2.7cm (219), 1.4cm (232) and 1.7cm (248). Cases have different wave frequencies, wave depths and thus steepness, which have been summarised in Table 5.1.

Even though the above measurements include the effect of the beach, due to the contamination of the water measurements from the latter, they still allowed for a classification of the experiments based on the water depth to be made. This is the reason why 219 has been classified as deep water experiment, 232 medium water and 248 as shallow-water in Table 5.1. Using this classification system as is, the bed depth just off the right wall in Figure 5.15 and the length of the beach in Figure 5.16 have been compared against each other.

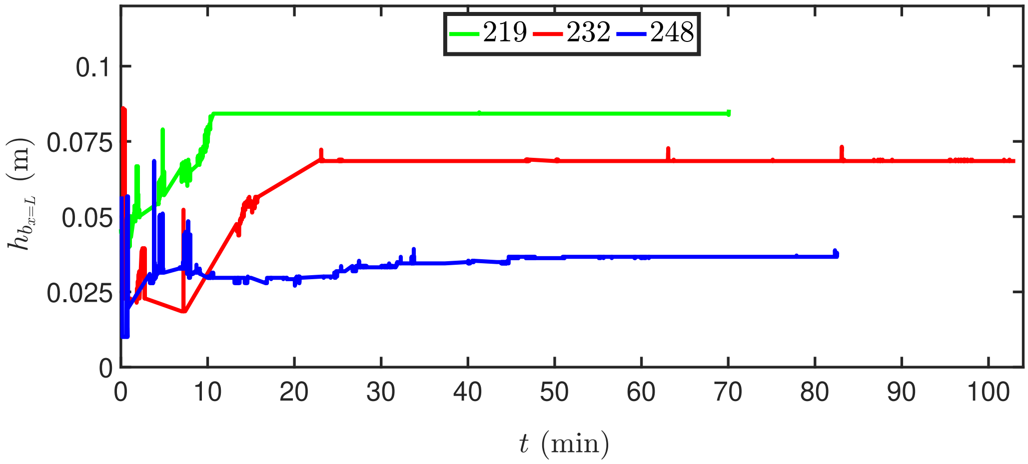


Figure 5.15: Value of the height of the beach at the end of the tank ($x = L$), left of the right wall. The height of the beach for each case follows the same order as for the mean water depth, provided the system has stabilised ($t_{219} \simeq$ 10min, $t_{232} \simeq$ 26min, $t_{248} \simeq$ 40min). The higher the mean depth, the closer the top of the beach is to the right wall. Cases have different wave frequencies, wave depths and thus steepness, which have been summarised in Table 5.1.

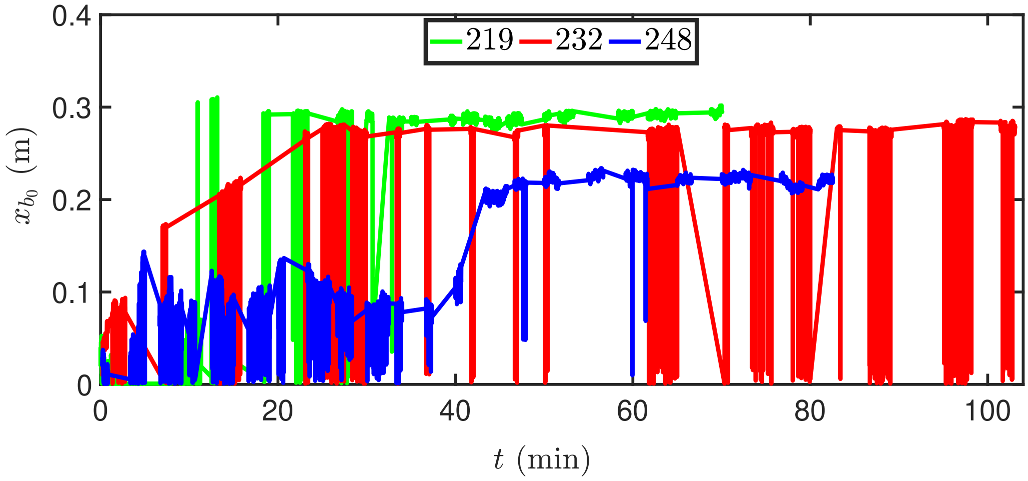


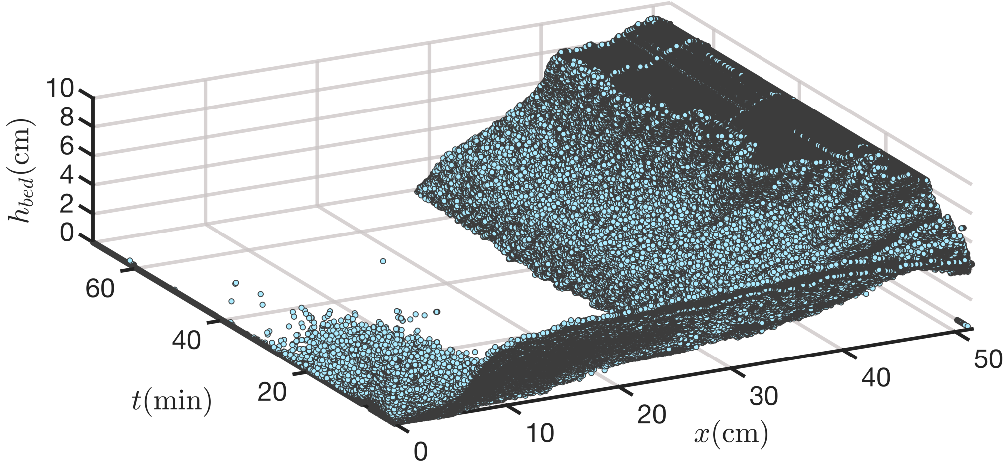
Figure 5.16: Evolution of the toe of the bed over time. As soon as the system has stabilised ($t_{219} \simeq$ 10min, $t_{232} \simeq$ 26min, $t_{248} \simeq$ 40min), each beach begins at $x_{219} = 0.29$m, $x_{232} = 0.275$m and at $x_{248} = 0.22$m. The beach ends at $x_{end} = 0.515$m. Cases have different wave frequencies, wave depths and thus steepness, which have been summarised in Table 5.1.

Comparing Figures 5.14, 5.15 and 5.16, it can be indeed confirmed that experiments in deeper water, relatively to the bed depth, led to taller and narrower beaches at the cross-shore direction with their top being very close to the right wall. It should be noted that these graphs are not an adequate diagnostic for determining the (final) quasi-equilibrium. The reason for this is because Figures 5.15, 5.16 demonstrate how the two ends of the beach, the toe on the left and the depth on the right, behave. As a result, after the initial beach profile (red) is formed in Figure 5.12, the beach front builds up without changing the beach height and length much. Hence Figures 5.15, 5.16 can be used to only roughly determine when the bed reaches the first quasi-equilibrium state. It is therefore concluded that at the end of the beach, *i.e.* the left toe (x_{b_0}) and depth at right wall ($h_{b_{x=L}}$), the bed equilibrated the quickest while the swash region equilibrated last with the waves honing the beach front.

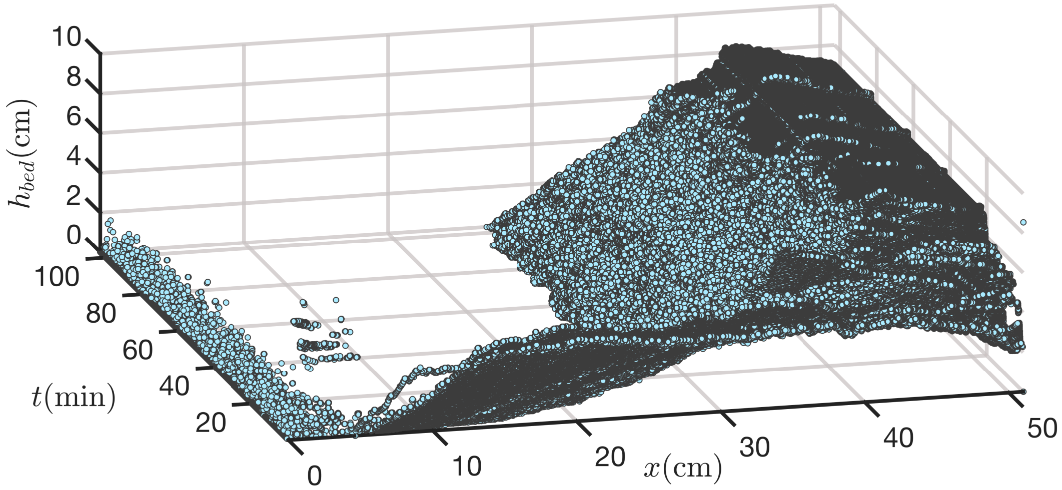
The initial bed profile was kept uniform as much as possible throughout all experiments. The initial bed depth on the other hand, the values of which are displayed in Figure 5.13a, presented some deviations throughout the experiments. Although the mean water depth has been found to affect the shape of the formed beach in the experiments outlined in this thesis, it is expected that the difference between the water and bed depth ($h_w - h_b$) is what ultimately determines the final bed profiles. This effect was not apparent in the experiments undertaken as the initial bed profiles were set to have similar thickness values thus masking the influence of the bed depth on $h_w - h_b$. The current data is therefore inadequate to allow a conclusion to be drawn on the effect of the initial bed conditions on the final bed profile. An investigation on the effect of the beach thickness on $h_w - h_b$ is thus suggested as future work, with the effective beach thickness considered influential by Powell [75].

The effect of the initial beach shape, though not investigated in this thesis, is expected to have an effect on how fast waves start breaking. At the moment the initial bed profile is horizontal and a variety of breaking waves are captured as the bed gradually steepens, due to wave action, ultimately forming a beach. If an experiment with a sloped initial bed profile is performed, it is expected that it will either lead to a reduced spectrum of created breaking waves or to some breakers being formed more often than others. More information on breaking waves can be found in the next section.

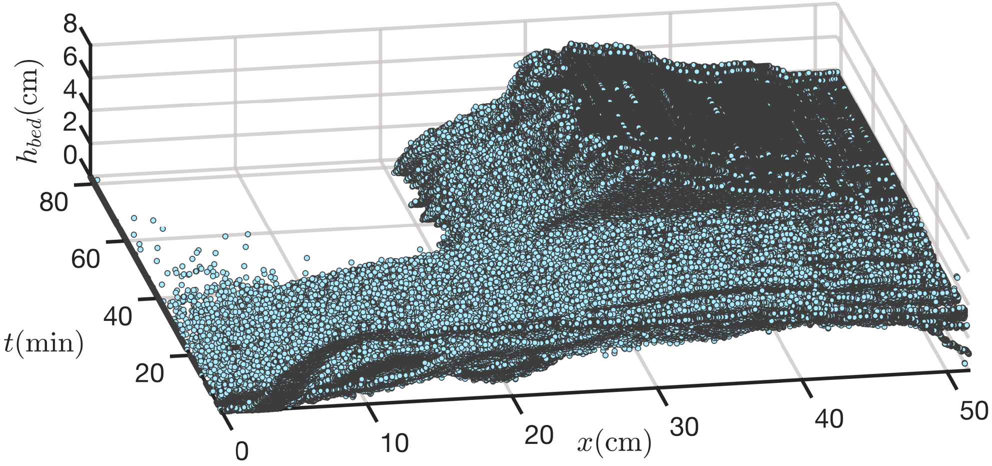
In the following 3D figure, the bed evolution has been tracked for all cases for the duration of the recording, with some interim times omitted so as to assist with the figures visibility. As mentioned earlier, the deeper the water the narrower in length and taller in height the beach is.



(a) Case 219. As time goes by, the top of the beach builds up. The beach is wide with its top high up.



(b) Case 232. As time goes by, the top of the beach builds up. In this case, a step is observed between the top of the beach and the area in front of it, a result of the water depth being smaller than in case 219. In videos of the experiment it was observed that between t =72min-85min the step collapsed. By the end of the recording, the previously collapsed step had been built back up.



(c) Case 248. The formed beach is the widest out of three with its top the furthest offshore.

Figure 5.17: Tracked bed evolution ultimately leading to beach formation for each experiment. Some interim times have been omitted from the above graphs to assist with the data visualisation. See 1D beach evolution in Figure 5.12.

5.2.1.4 Sediment flux

Tracking the bed height over the course of the experiments paved the way towards understanding how the flux of sediment over time changed the bed. It was thus of interest to quantify the bed flux, q_b, and understand how it varied in the tank over a wave-cycle, *i.e.* a wave period. More specifically, the interest was on monitoring whether a transition to a quasi-steady state occurred as the bed flux approached zero ($q_b \to 0$). To this end, the cell was divided into bins of equal length (see Figure 5.18), in which the bed height was tracked by the image-analysis algorithm at various wave-cycles over the course of the recorded wave-bed experiments, previously summarised in Table 5.1.

As the bins always had the same coordinates, it was easier to track the bed 'flowing' in and out of each of them. This was not the case during the wave-bed tracking, because the bed coordinates would solely indicate the regions in the tank where the bed was present, while being updated accordingly to follow its movement over time. A summary of the calculation of the bed flux can be found in Figure 5.19. Superscripts denote time and subscripts space/bins, the bed height at time t^n is equal to:

$$b^n = (b_1^n, b_2^n, ...b_N^n), \tag{5.4}$$

where b^n the vector consisting of the bed height contributions from each bin $j = 1, 2, ..., N$.

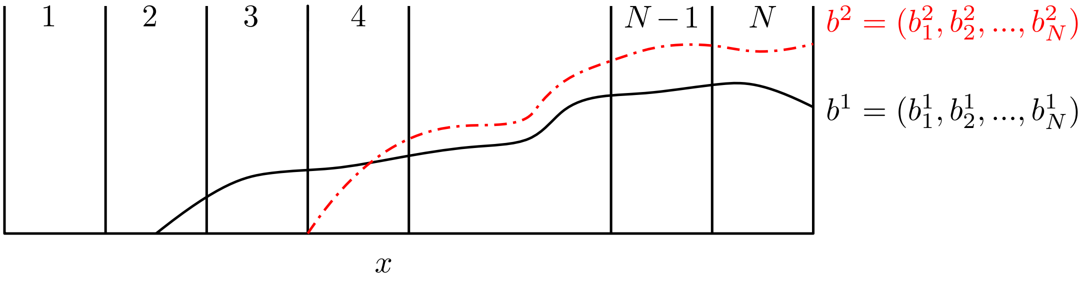


Figure 5.18: The tank's length has been divided into $N = 40$ bins with the bed height coordinates mapped into each bin at various times. Sample bed heights at two different times (b^1, b^2) have been sketched above.

The bed flux was computed in the same way as the area flux for the water (see Section 4.1). The formula for the bed, computed at time t^n, was thus the following:

$$q_b(t^n) = \frac{dA_b^n}{dt}, \tag{5.5}$$

with $A_b^n = \sum_{j=1}^{N=40} b_j^n * \Delta x \approx \int b dx$ the total bed area enclosed in all bins, defined as the sum over the number of bins of the product of the bed height b_j^n at time t^n by the bin length Δx.

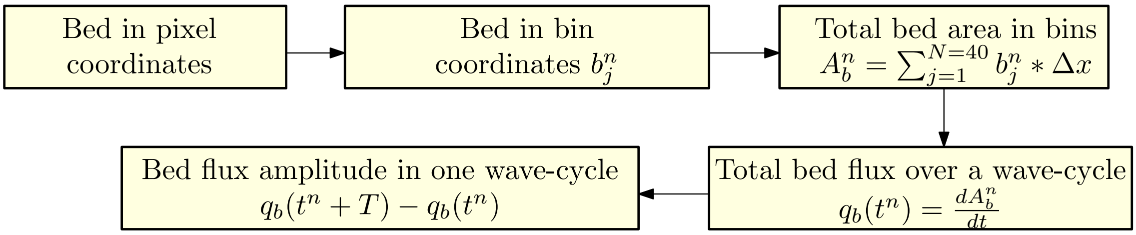
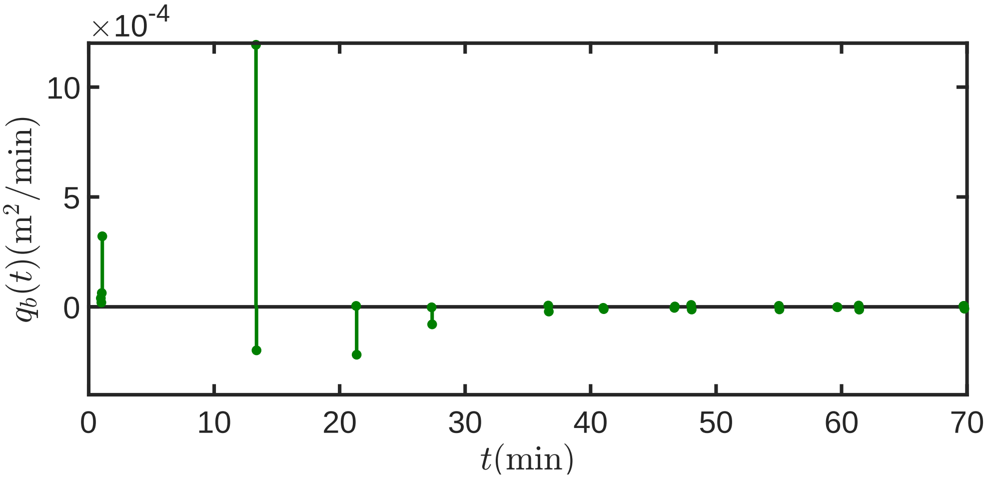
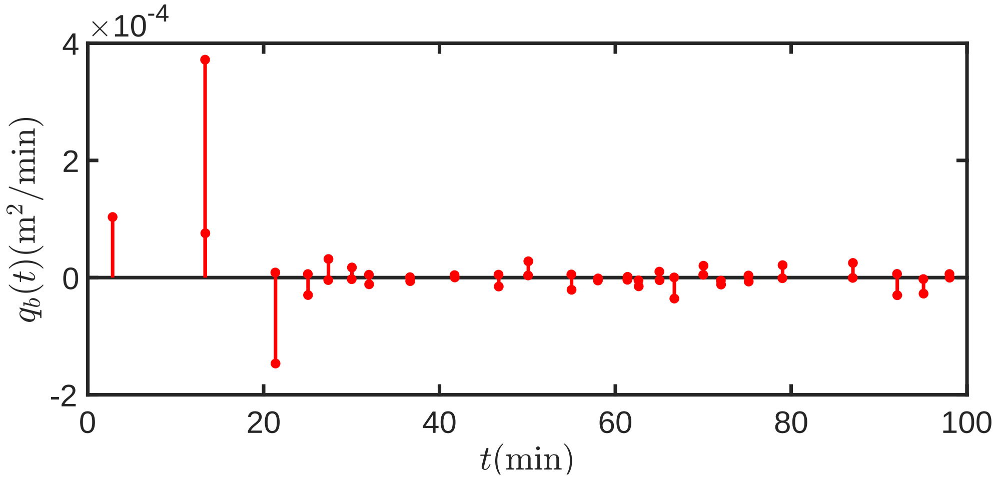


Figure 5.19: Steps to bed flux computation: i) transform the bed height from pixel to bin coordinates, ii) compute the total bed area present in all bins A_b^n and iii) determine the bed flux by taking the difference in the total area over one wave-cycle/period, with iv) the bed flux amplitude equal to $q_b(t^n + T) - q_b(t^n)$.

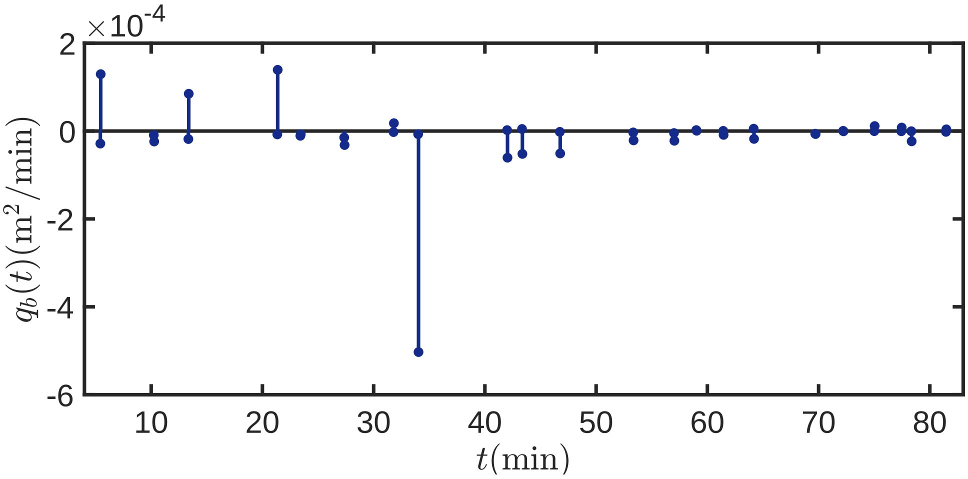
The bed flux for each case has been visualised in the following figures:



(a) Bed flux for case 219-223. The value of q_b starts to decrease at $t = 36\mathrm{min}\,40\mathrm{s}$ with the transition to a quasi-steady state happening gradually as $q_b \to 0$.



(b) Bed flux for case 232-237. The value of q_b decreases at $t = 41\mathrm{min}\,43\mathrm{s}$; after that the profile of the bed flux fluctuates in a small neighbourhood close to zero. In this case the bed flux fluctuates the most, an event also observed in Figures 5.12b and 5.17b, and attributed to the medium water depth that caused the bed to alternately collapse (t =72min-85min) and rebuild ($t = 85$min-end) therefore delaying the system from reaching a final quasi-equilibrium.



(c) Bed flux for case 248-256. The value of q_b starts to decrease at $t = 42\mathrm{min}\,1\mathrm{s}$ and remains in a small neighbourhood about zero after $t = 53\mathrm{min}\,22\mathrm{s}$.

Figure 5.20: Bed flux computed over a wave-cycle at various times for case 219-223 (top panel), case 232-237 (middle panel) and case 248-256 (bottom panel). The system reaches a quasi-steady state when the bed flux approaches zero. The amplitude of the flux in each wave-cycle can be found in Figure 5.21.

It can be seen in Figure 5.20 that each experiment had a different bed flux profile. In case 219, the bed flux q_b gradually reached zero after the first twenty minutes without any deviations. Case 232 behaved differently than case 219 in the sense that the values of q_b decreased after the first twenty minutes, but still fluctuated close to zero until the end of the recording. Lastly, the bed flux for case 248 took forty minutes, the longest out of all cases, to decrease with some fluctuations in its value close to zero visible. Recalling Figures 5.12 and 5.16, the times for the first quasi-equilibrium partly agree with the temporal changes in the profile of the bed flux. It should be noted that the bed flux profile would not have been impacted if the tracked bed heights had been smoothed first (see Appendix E.3, Figure E.28). The effect of smoothing on the bed flux values in particular, has been found to be in the order of O(-8) to O(-6) (see Appendix E.3, Figure E.29).

A relationship between the water depth and the different bed flux profiles is revealed in Figure 5.20. Case 219-223 had the highest water depth, and the smoothest transition to zero of the bed-

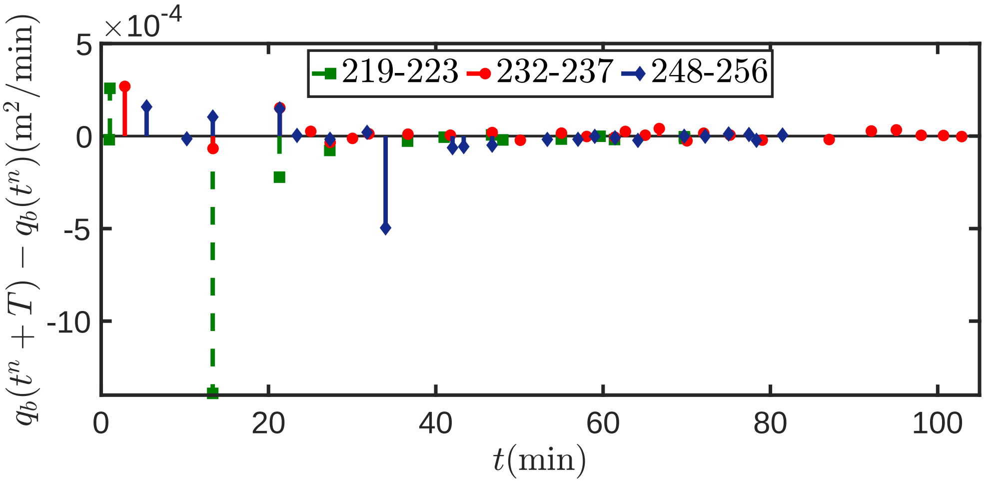


Figure 5.21: Amplitude of the variation of q_b over each wave-cycle at various times for each experiment. Case 219 (green squares), case 232 (red circles) and case 248 (blue diamonds) have been depicted in the above panel.

flux profile. On the other hand, cases 232-237 and 248-256 resulted in bed fluxes that approached zero but still varied, hence revealing that the bed profile was potentially more sensitive to changes the closer it was to the water. This was particularly the case for case 232-237, where the bed collapsed and was rebuilt during the last forty minutes of the recording.

It can be thus confirmed that the wave-bed system starts to equilibrate, *i.e.* to approximate a quasi-steady state, once the bed flux starts to decrease. However as long as water is pumped into the tank, the interaction between the waves and the beach does not cease; an event manifested by the presence of small bed flux fluctuations. Due to the limitations of the current work, there are not enough data to fully investigate the effect of various water depths on the bed flux profile and development in this area is proposed as future work.

5.2.1.5 Sediment loss

It can be seen in Figure 5.22, that sediment loss was an issue that all experiments faced. Videos 232-237 endured the least amount of sediment loss out of three experiments. The reason for this goes back to the experimental set-up. The wave tank used, though very easy-to-operate with pumps that could reach a broad range of frequencies, had originally been created solely for demonstration purposes with limited access to the inside of it.

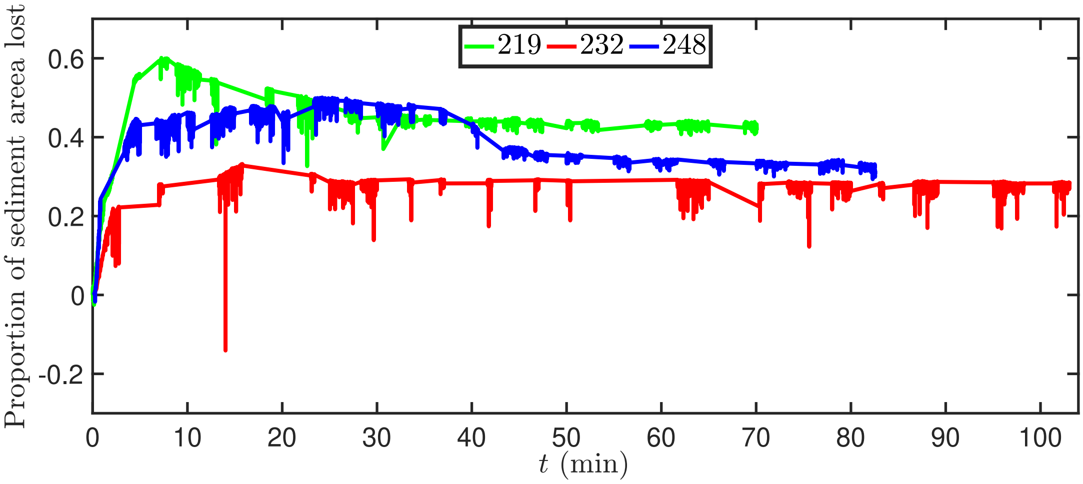


Figure 5.22: Proportion of sediment area lost over the duration of each experiment. Case 219 (green) lost 43.5% of sediment, 232 (red) lost 29% and 248 (blue) lost 36%.

The left side of the tank, where the water was entering the cell, was equipped with a mesh thus not allowing for particles to leave the tank and reach the pumps. In contrast to the left wall, the right wall of the cell had no such protection. The presence of an inner gap between the right wall and the glass plates edges created a hiding spot for the particles. As a result, the particles often ended up at this "hidden" gap as a result of their transport by waves. Ergo, the image tracking could not access the particles that were lost; hence sediment volume was not conserved. As this gap was difficult to access, the only way to minimise the effect of this space on the measurements was by filling the gap with plastic tubes. These vertical tubes occupied some of this void space thus preventing some of the particles from going in.

A positive correlation between the wave period and percentage of sediment loss has been observed. In the framework of the examined experiments, higher wave periods coincided with higher percentages of sediment loss. This is possibly related to the fact that smaller wave periods and wave heights lead to beach formation at a slower rate, thus keeping sediment loss constantly at lower levels.

5.2.2 Breaking waves

Ocean waves reach the coasts by propagating from deep to shallow-water. The region where this transition occurs is the shoaling zone; in essence the wave depth goes from being larger than half of the wavelength to less than half of the wavelength. This transition affects the wave speed, wave height and wavelength. The water waves then enter the shallow-water zone in which the wave length is much larger than the depth. As a rule of thumb, in shallow-water the wavelength is often selected to be twenty times or more than the depth. In the shallow-water zone the depth decreases hence the wave speed, which is depth-dependent and determined from the linear theory as $u_g = \sqrt{gh}$, decreases too. Consequently, as the kinetic energy of the wave group decreases, the potential energy increases thus causing the wave height to rise. The wave height keeps increasing until the wave front becomes too steep and the wave front overturns, a phenomenon known as wave breaking. Right after breaking, breaker jets of water of varying strength are created that reach the beach and dissipate the waves energy. From a coastal protection point of view, gravel beaches have the advantage that their permeability causes extra dissipation of wave energy, in the form of turbulence and heat [28], in addition to that lost during wave breaking in the surf zone.

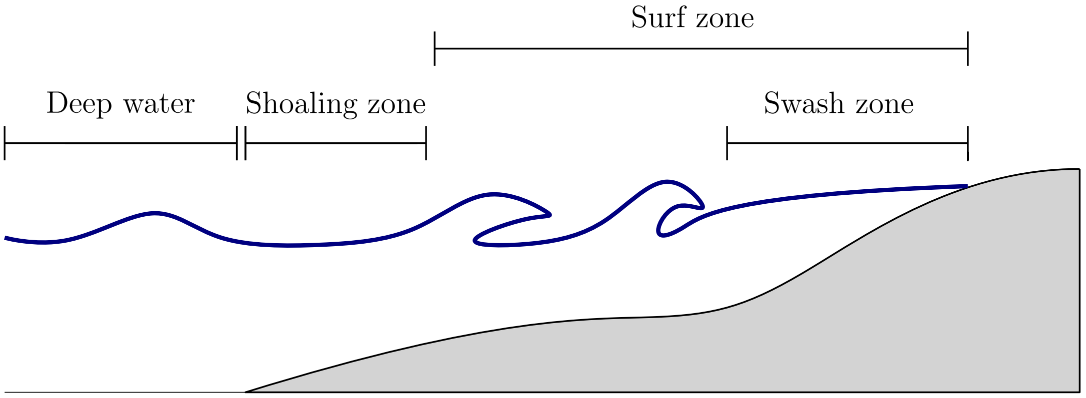


Figure 5.23: Waves travel from deep water towards the shore, thus entering the denoted coastal zones. They first enter the shoaling zone and later on break in the surf zone. At the swash zone the beach is alternately wetted and dried with sediment transport being prevalent.

5.2.2.1 Types of breakers

There are four types of breakers: spilling, plunging, collapsing and surging, deduced in [13, 31, 68]. Each breaker has a different profile, summarised in Table 5.3, with the wave steepness and beach slope influencing the type of breakers created.

Table 5.3: Types of breakers and their description, as found in [56, 68].

Breaker	**Description**
Spilling	Presence of foam at the wave crest, water spills down the front face
Plunging	Wave crest overturns and a jet plunges into the trough ahead
Collapsing	The lower front face overturns and behaves like a truncated plunging breaker
Surging	Associated with flat incident waves that overturn at the moving shoreline

The effects of the wave and beach slope on wave breaking were cast into a formula by Iribarren (1948):

$$I_b = \frac{\tan\alpha}{\sqrt{\frac{H_b}{\lambda_b}}}, \tag{5.6}$$

with $\tan\alpha$ the slope of the bed of angle α, H_b the height of the breaking wave measured at the toe of the bed and $\lambda_b = \frac{gT^2}{2\pi}$ the wavelength as determined from the linear deep water theory under a wave period T. The wavelength can be recovered if one takes the dispersion relation

$$\omega^2 = gk\tanh(kh), \tag{5.7}$$

with k the wavenumber, h the water depth and g the momentum acceleration due to gravity. At the deep water limit the depth is much larger than the wavelength ($kh >> 1$) translates the dispersion relation to

$$\omega^2 = gk, \tag{5.8}$$

as $lim_{kh\to\infty}\tanh(kh) = 1$. Substituting the wavenumber $k = \frac{2\pi}{\lambda_b}$ and $\omega = \frac{2\pi}{T}$, for a known wave period T, to (5.8) the deep water wavelength λ_b is acquired. In spite of the fact that wave breaking occurs in shallow-water, a deep water wavelength formula is in use. According to Battjes [13], taking the deepwater wavelength is equivalent to taking the shallow-water wavelength because the height-to-depth ratio is of order one right before breaking.

The classification of the breaker types is thus possible based on the values of the Iribarren number, defined in equation (5.6). Consequently, spilling breakers were identified for Iribarren values in the range of $I_b < 0.4$, plunging for $0.4 < I_b < 2$, collapsing for $2 < I_b < 3.3$ and surging for $I_b > 3.3$.

5.2.2.2 Identifying breaker types in experiments

During each set of experiments water waves of varying frequency, depth or steepness were created. The shape of the beach changed over time as a result of the incident waves; hence a gradual transition from uniform to steeper bed profiles was observed. For each experiment the breaker types were determined initially by observation and their type was then confirmed by computing the Iribarren number.

The first step towards determining the Iribarren number was to calculate the slope of the bed, wave period and wave height at the toe of the bed. The wave period had already been measured during the recording of the experiments and it was thus known. The wave height at the toe of the bed was calculated by substituting the maximum and minimum water depth measurements, located at the start/toe of the beach x_0 and acquired from runs of the image-analysis code at the times of interest (x_{b_0}), into the wave height formula:

$$H_b|_{x_{b_0}} = h_{max} - h_{min}\Big|_{x_{b_0}}. \tag{5.9}$$

The slope of the bed was computed by substituting the coordinates of the minimum and maximum bed measurements located at the wet region, acquired from the image-analysis code, in equation (5.3); thus working in a similar way as for the angle of repose. The only difference was that this time the slope of the bed was required at interim times from beginning to end, at the times of the observed breakers, while previously it was only computed at the end of the experiments. Substituting the values of the slope of the bed, wave period and wave length in equation (5.6) determined a value for the Iribarren number. Finally, this value was compared against the values classifying the breakers mentioned earlier. Comparing the observed breakers and their respective Iribarren number, it was found that the latter predicted the breaker types successfully. Further information on the specific times, wave and bed heights that the breakers were computed for can be found in Appendix E.4.

After the breakers had been identified, they were organised in a timely manner from beginning to end for each experiment. In other words, a timeline was created depicting the transition from one breaker type to another throughout each experiment, elucidating the interaction between beach formation and breaking waves. In the timeline figures that follow, each identified breaker type has been denoted with a different colour. Spilling breakers are marked with orange, plunging with cyan, collapsing with yellow and surging with green.

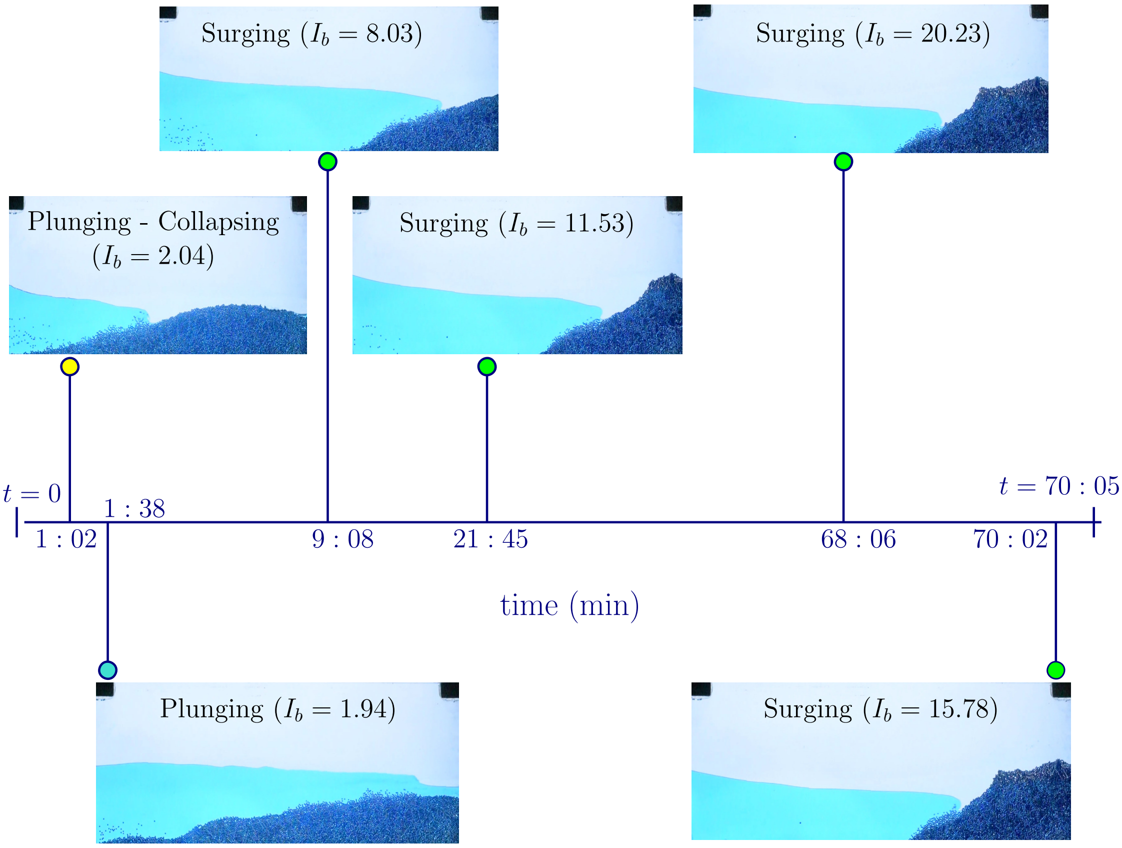
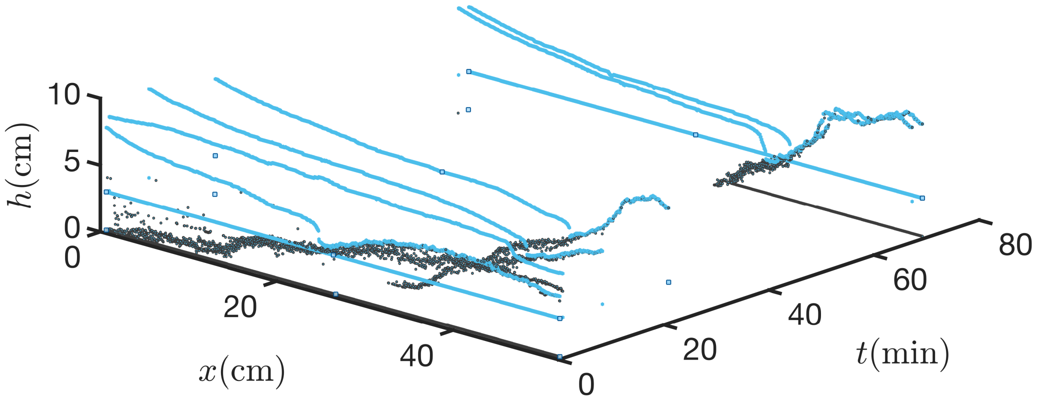
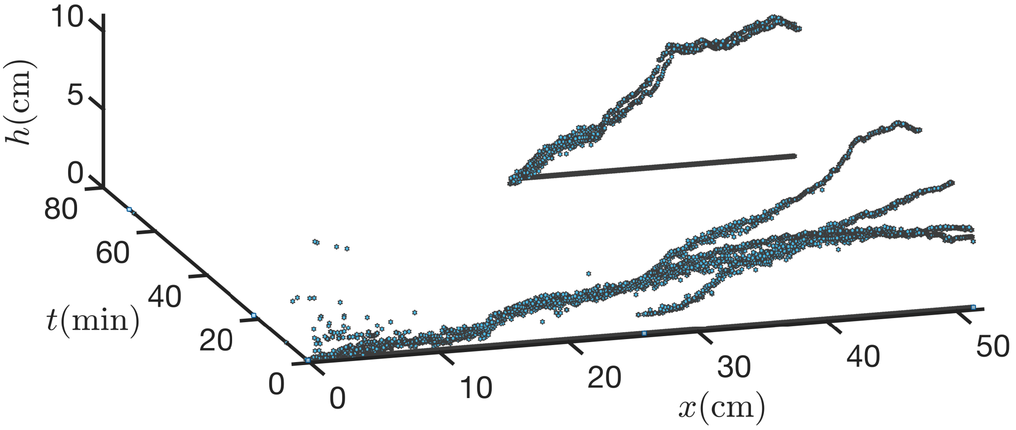


Figure 5.24: Timeline of breakers observed in videos 219-223 at times $t =$ 1min 2s, 1min 38s, 9min 8s, 21min 45s, 68min 6s, 70min 6s. The first breaker is transitioning from plunging to collapsing and a surging breaker appears on the first ten minutes of the experiment.



(a) Tracked breaking waves. The porous bed has contaminated the water measurements during the last sixty minutes.



(b) Bed morphology evolution viewed from a different angle. The beach front becomes taller, steeper and is closer to the right wall.

Figure 5.25: The interaction between the incident waves and the bed has led to wave breaking (top panel) and beach formation (bottom panel), tracked and reproduced by the image-analysis algorithm for videos 219-223, at times $t = 1\text{min}\,2\text{s}, 1\text{min}\,38\text{s}, 9\text{min}\,8\text{s}, 21\text{min}\,45\text{s}, 68\text{min}\,6\text{s}, 70\text{min}\,6\text{s}$.

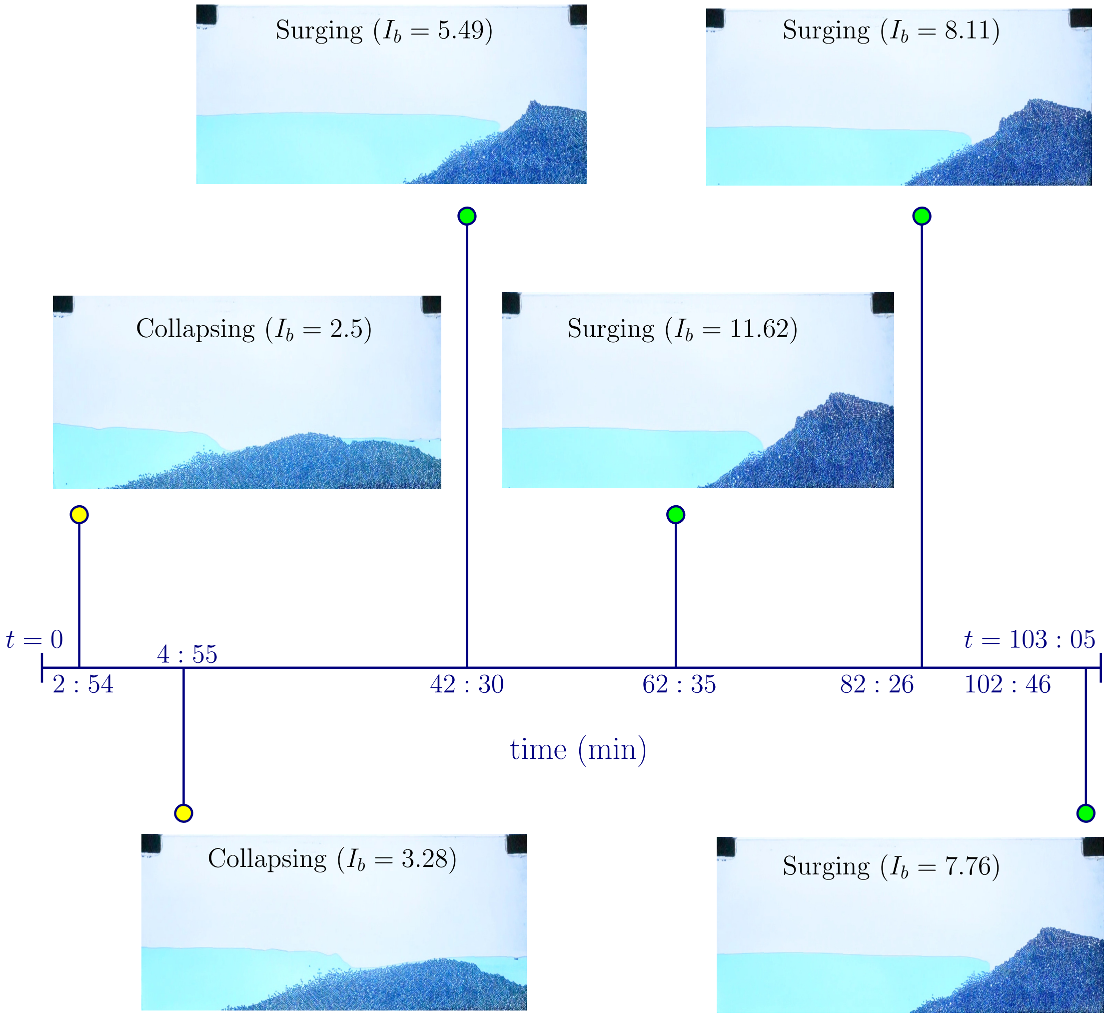
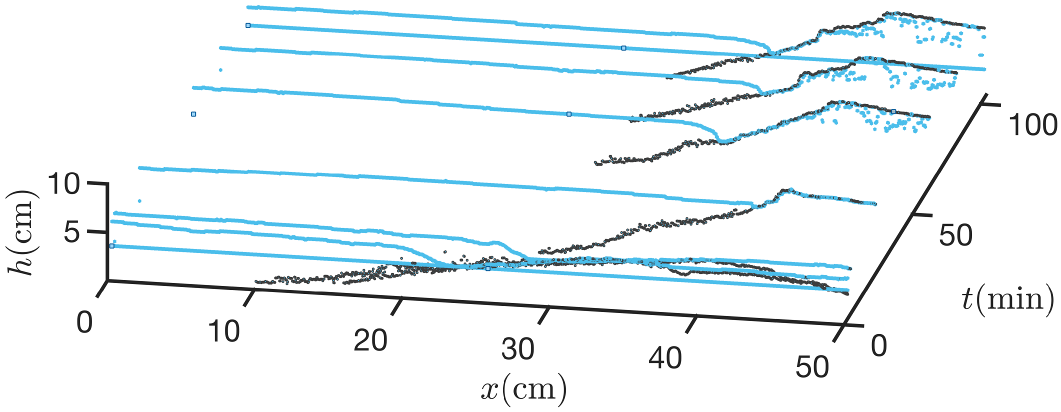
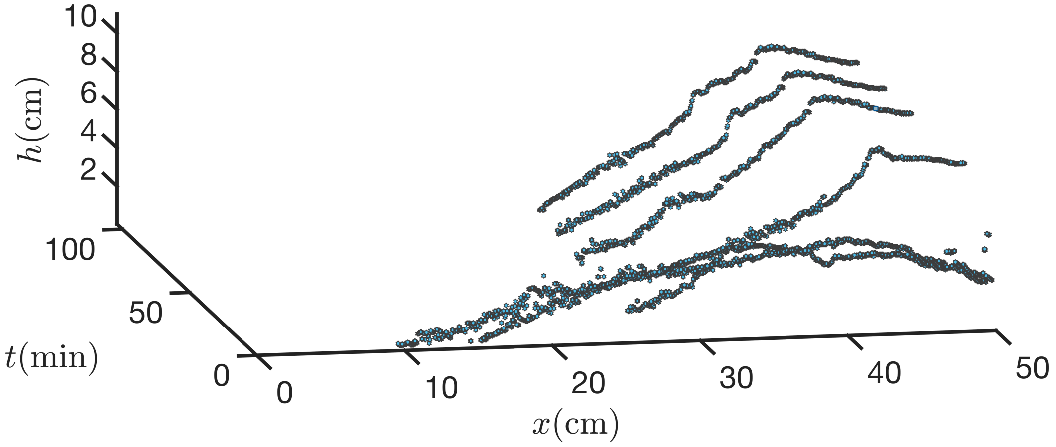


Figure 5.26: Timeline of breakers observed in videos 232-237 at times $t = 2\text{min}\,54\text{s}, \text{t} = 4\text{min}\,55\text{s}, \text{t} = 42\text{min}\,30\text{s}, \text{t} = 62\text{min}\,35\text{s}, \text{t} = 82\text{min}\,26\text{s}, \text{t} = 102\text{min}\,46\text{s}$. The first breaker is of a collapsing type.



(a) Tracked breaking waves. The porous bed has contaminated the water measurements during the last forty minutes.



(b) Bed morphology evolution viewed from a different angle. The beach front becomes steeper, taller and closer to the right wall.

Figure 5.27: The interaction between the incident waves and the bed has led to wave breaking (top panel) and beach formation (bottom panel), tracked and reproduced by the image-analysis algorithm for videos 232-237, at times $t = 2\text{min}\,54\text{s}, 4\text{min}\,55\text{s}, 42\text{min}\,30\text{s}, 62\text{min}\,35\text{s}, 82\text{min}\,26\text{s}, 102\text{min}\,46\text{s}$.

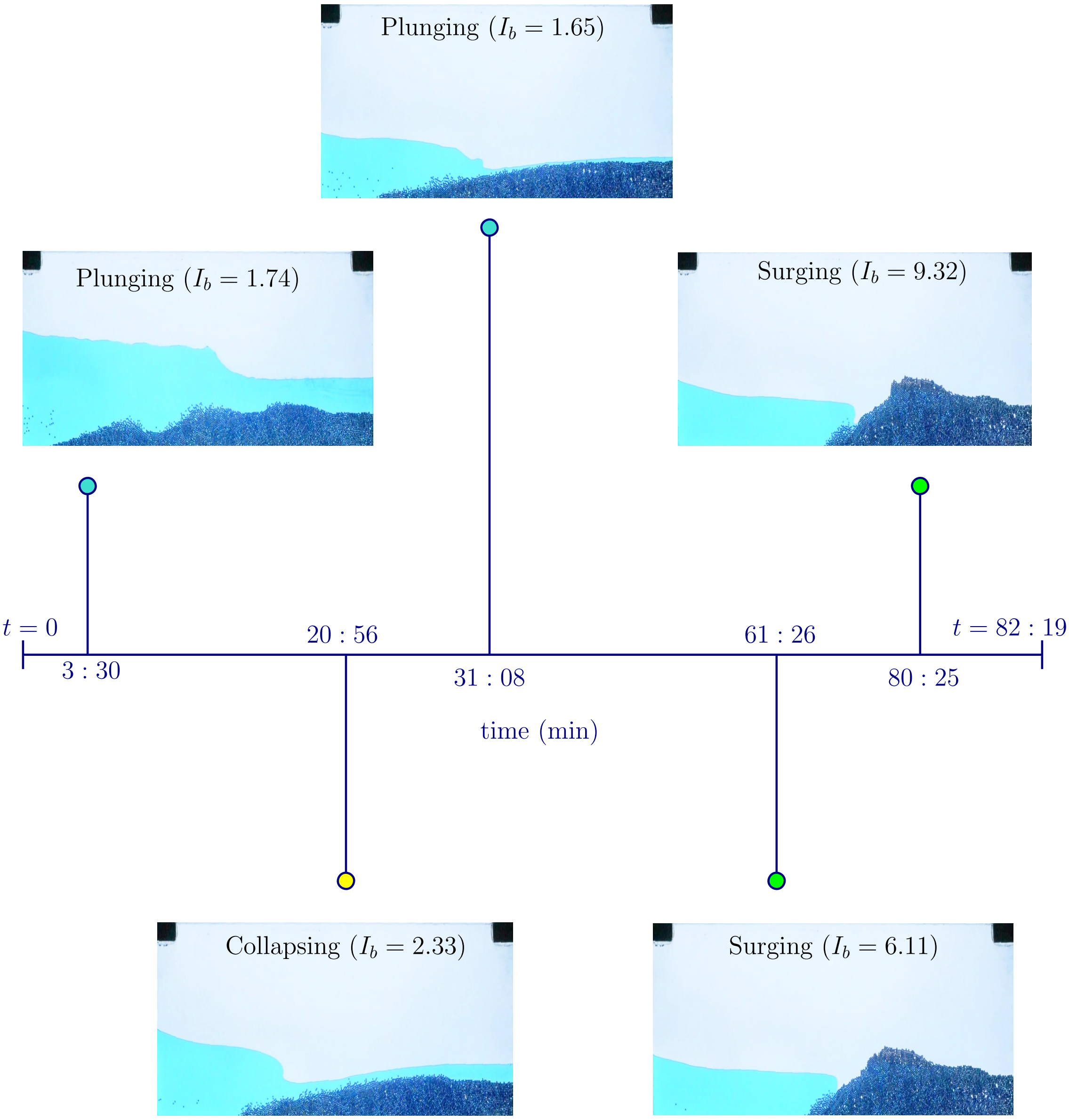
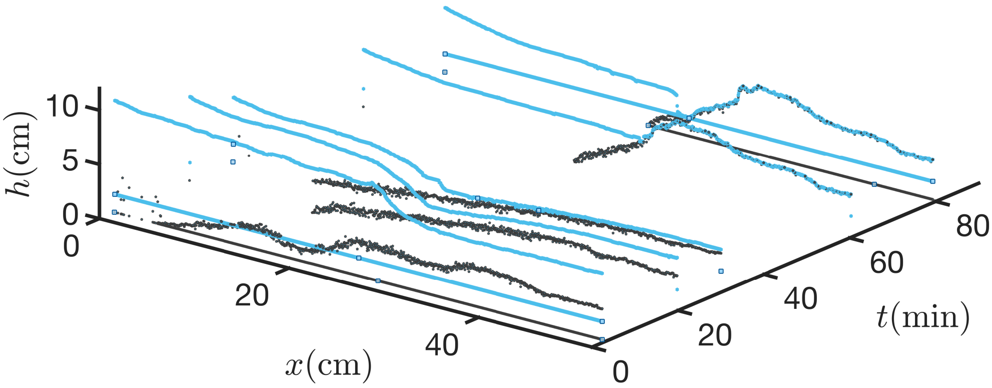
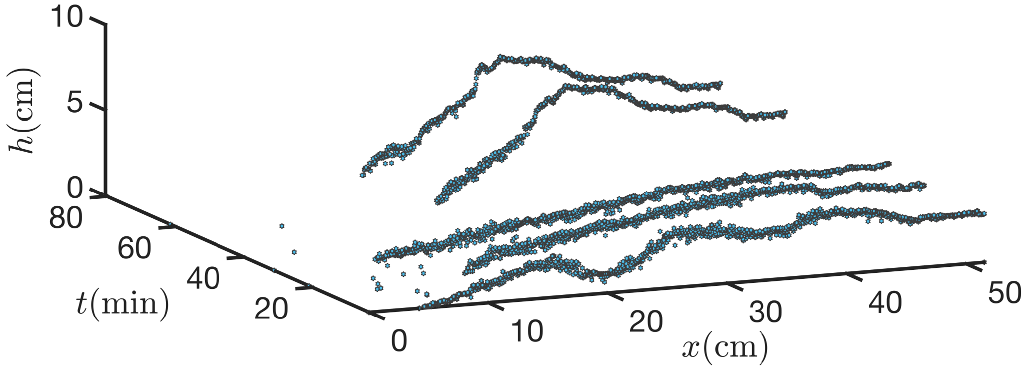


Figure 5.28: Timeline of breakers observed in videos 248-256 at times t = 3min 30s, 20min 56s, 31min 8s, 61min 26s, 80min 25s. The first breaker is of a plunging type.



(a) Tracked breaking waves. The porous bed has contaminated the water measurements during the last twenty minutes.



(b) Bed morphology evolution viewed from a different angle. The beach front becomes steeper as it gradually builts up and widens towards the left.

Figure 5.29: The interaction between the incident waves and the bed has led to wave breaking (top panel) and beach formation (bottom panel), tracked and reproduced by the image-analysis algorithm for videos 248-256, at times $t = 3\mathrm{min}\,30\mathrm{s}, 20\mathrm{min}\,56\mathrm{s}, 31\mathrm{min}\,8\mathrm{s}, 61\mathrm{min}\,26\mathrm{s}, 80\mathrm{min}\,25\mathrm{s}$.

5.3 Conclusions

In this chapter an extended image-analysis algorithm, created primarily for tracking the water and moving bed, has been discussed. Its performance has been successfully tested against three experiments involving water and sediments. The algorithm tracked the moving bed and the water successfully, with water tracking errors around 1mm and bed tracking errors in the range of 1.5-2mm, thus manipulating and analysing hundreds of thousands of images accurately for each of the examined cases.

The tracked images provided useful data that were further analysed. In some cases, the wet bed interfered with the tracking of the water thus contaminating the acquired data. Analysing the obtained data, revealed that lower frequencies affect beach formation the fastest and that the distance between the water and the bed affects how wide and tall the final beach profile will be. It was also found that at the end of the beach, *i.e.* the left toe and the beach height at the right wall, the bed equilibrates the fastest while the continuous wave action hones the beach shape at the swash region. In fact, the wave-bed system reached a (final) quasi-equilibrium for each case when the bed flux in the domain approached zero over various wave-cycles. Unfortunately, due to the practical limitations of the set-up these effects have not been investigated extensively. The presence of a gap in the tank has also resulted in loss of sediment, at different rates for different wave heights and wave periods, which in turn affected the experiments and ultimately the beach measurements.

Furthermore, the angles of each of the final bed profiles have been computed manually, using the measurements provided by the image-analysis algorithm, for two regions of the bed, the wet and the swash region. The angles of the wet region were always smaller than the swash angles, a phenomenon also observed in [90] and attributed to the effect of liquid bridges. The angles of the wet region of the final-bed profiles have also been compared against reported angles of repose for gravel beaches; in all cases θ_{wet} was within the range provided by the literature. The presence of the water in the wet region may potentially be the reason for obtaining smaller angles.

Lastly, the image-analysis algorithm has been used as a proxi for the identification and classification of the observed breaker types, essentially by providing measurements for each experiment. Measurements such as the angle of the bed, the wave height and wave period have been used for the computation of the Iribarren number with plunging, collapsing and surging breakers identified in the experiments. Indeed, the Iribarren number predicted the breaker types that were observed during the experiments successfully.

Chapter 6

Conclusions and future work

This thesis concerned the development of an image-analysis algorithm tailored to the vertical Hele-Shaw cell's shape and size. It allows the conversion of experimental images with water and a moving bottom topography into useful data in a simple yet precise way. The acquired data were not only used as a means of validating the relevant mathematical models, but also led to better understanding of beach formation and the factors that affect it. A summary of the research outcomes can be found in Section 6.1, with the research aims revisited in Section 6.2. Lastly, future work recommendations are provided in Section 6.3.

6.1 Summary

Chapter 2 of the thesis presented the Hele-Shaw set-up in detail for two wave tanks of different length and width. The water-only experiments carried out within these tanks were also discussed in this chapter. During the experiments, the operation of the wave pumps remained the same throughout the tanks and allowed for a range of frequencies to be reached: between 0.3 and 0.7 Hz. The depth of the water also varied between experiments. The image-analysis algorithm, the linking point between images and data, was also explored in chapter 2. The algorithm originally provided an option to choose between two different tracking methods; an edge-detection method and a colour-threshold-detection method, with their corresponding detection results discussed and compared against each other in Section 2.3.3.2 and Appendix A. The colour-tracking method, also aided from the optimised experimental set-up, provided optimal results for a variety of wave cases.

Chapter 3 examined the suitability of two one-dimensional mathematical models to describe the hydrodynamics in the Hele-Shaw cell. A shallow-water model and a potential-flow shallow-water model were extended so as to include a time-dependent forcing term and a momentum dissipation term. The forcing term accounts for the wave sloshing that occurs as a result of the operation of the pumps and is in effect at the left wall, through which the water enters the cell. For this reason an inflow boundary condition was imposed at the left wall in both models. Momentum dissipation, attributed to the tank's fixed-by-construction narrow width, depends on the kinematic viscosity and the half-width of the tank. The shallow-water model was solved with first-order numerical schemes: a finite-volume (Godunov) method in space and a Forward Euler method in time. In contrast, the potential-flow for shallow-water model was solved using second-order numerical schemes: a finite-element spatial discretisation utilising linear Lagrange polynomials and a Störmer-Verlet time-discretisation.

The models were validated in chapter 4, by gradually imposing the obtained (smoothed) data to the left-wall boundary condition; this is the novelty of the validation process. Initially, a fitted function to the smoothed data was imposed at the inflow boundary condition of each model during simulation one. Then, during simulation two, the smoothed data values for the inflow and the depth at the left wall entered the models through the existing inflow boundary and a Dirichlet boundary, respectively. Comparisons between either the simulations or the numerical methods were carried out.

It was found that simulation two yielded more accurate results than simulation one for the majority of the cases examined. The FE method became unstable for higher influx amplitudes combined with shallower prescribed depths. In contrast to the FE method, the FV method handled steeper waves better then the FE method. However, the numerical wave fronts were often different than the ones observed during the experiments, with the difference increasing for waves of increasing steepness. Even though overestimating reflection was one of the main sources for this error that both numerical models faced, it was nonetheless an effect applied uniformly to all cases. Hence the errors may originate from the model and in particular from a term missing from the shallow-water equations, ignored after the width- and depth-averaging.

In an effort to understand beach formation further, water-bed experiments were performed and have been presented in chapter 5. This constant interaction between the water and the bed was further analysed with the help of the extended image-analysis algorithm, discussed in the same chapter. In a nutshell, the algorithm from chapter 2 was modified so as to track the water and the moving bed simultaneously. Its novelty lies in the fact that it uses the same principles as before, whilst tracking two objects and working in a precise, with water tracking errors around 1mm and bed tracking errors between 1.5-2mm, and iterative way. The successful tracking allowed for an investigation into the types of breaking waves created as well as the angles of the final bed profiles and their comparison to real-life gravel beaches.

It was found that smaller frequencies led to beach formation faster than higher frequencies. In addition higher water depths, in relation to the bed depth, resulted in beach profiles that were narrower and taller as opposed to wider and shorter. To this end, both types of final beach profiles were obtained during the experiments. It was also found that at the end of the beach, *i.e.* the left toe and the bed height next to the right wall, the bed equilibrated the quickest with the continuous wave action honing the beach shape. Then, the wave-bed system also reached a final quasi-equilibrium, an event confirmed by observation as well as by the magnitude of the bed flux which was close to zero. The final beach angles were computed for the wet and the swash region with the latter being larger than the angles at the wet region. This result was found to agree with a phenomenon also known as liquid bridges, which essentially weakens the inter-particle forces in the presence of water. The angles of repose of the final beach profiles were also computed at the wet region area and were found to be in the range of angles attributed to real-life gravel beaches. Finally plunging, collapsing and surging breakers were identified during the experiments and were further confirmed by calculating the Iribarren number for each one of them.

The negative influence of the incomplete design of the tank has been accounted for. It affected both the numerical and the experimental results. Either by providing underestimated wave depth measurements near the right wall or by interfering with sediment transport and beach formation and not conserving total sediment volume.

6.2 Aims revisited

The aims set in Chapter 1 have been revisited with the work done to meet these aims summarised below.

1. **Determine which mathematical models can capture the dynamics of waves in a vertical Hele-Shaw cell.**
   Two one-dimensional models, presented in Chapter 3, were used throughout this work; a shallow-water model and a potential-flow shallow-water model. The potential-flow shallow-water model was obtained from Luke's (extended) variational principle and assumes that the flow is irrotational. Extended in that the effect of forcing at the left boundary and momentum dissipation had to be included in both models, to account for utilising a wave pump and having a small gap width between the glass plates. The linear models were verified successfully against a standing-wave and forced-wave exact solution of the linear problem with momentum damping being absent.

2. Measure how close the chosen mathematical models are in relation to reality, and to adapt the models if necessary in the light of experimental results.

   The models' ability to describe reality was then tested by imposing experimental data for the water depth and the area flux of water to the numerical models; a process also referred as validation process. The validation of the models entailed imposing the smoothed experimental data to the boundary conditions gradually through simulation one and simulation two. Simulation one imposed the fitted function for the area flux. Then, simulation two imposed the smoothed area flux data and the depth at the left wall.

   Simulation two yielded better results than simulation one, with the waves being less steep and closer to those observed in the experiments. This improvement was a direct consequence of the inclusion of the Dirichlet boundary. Furthermore, for cases with higher forcing, steep waves appeared that were in the limit of violating the irrotational flow assumption of the potential-flow model, creating instabilities. The FV method performed better than the FE method, especially for the cases with higher forcing. The suggestion could therefore be made that the finite element method is used only when the forcing is weaker, i.e for smaller inflow amplitudes. Even though both numerical models overestimated reflection in comparison to reality, it was an expected outcome attributed to the almost-solid right wall of the tank.

It was also found that numerical wave fronts computed by the FV method were faster than those observed during the experiments. The reason for this difference may be either the altered surface tension due to the dyed water or a term missing from the shallow-water equations.

3. Perform wave-only experiments for various wave frequencies and water depths and extract useful measurements of the water depth, through a bespoke tracking algorithm. Then, insert the measurements to the mathematical models and challenge their performance.

   Optimising the experimental set-up was also accomplished and it was of great importance, as it allowed for improved visibility of the tracked phenomena. An LED-bank provided uniform background light and combined with the dyed-blue water, it resulted in appropriate shades of blue that greatly assisted the colour-channels with the tracking.

   In Chapter 2, water experiments as well as the development of a robust and accurate image-analysis algorithm were presented. The algorithm's operation, importing images and exporting time series for the depth and the area of water under it, rely on the use of a colour-detection method. It was tested and verified against propagating water waves of various frequencies, depths and steepness in an automated way, with a selection of three depicted in this thesis.

   The algorithm was extended in Chapter 5 so as to recursively track two moving objects (water and bed) in thousands of snapshots, while using the same principles as before with water tracking errors around 1mm and bed tracking errors between 1.5-2mm. Ultimately, the extended algorithm performed very well, often capturing effects such as water infiltration, wave breaking and bed load sediment transport. Water measurements contamination was an issue that all cases in shallow-water faced. Essentially the wet bed was translated by the code as belonging to the water, thus overestimating the water depth computation.

4. Perform experiments with both water and particles. Vary the wave frequency, water depth and wave height so as to investigate beach formation and beach-related phenomena such as sediment transport and wave breaking. Then, collect measurements of the bed height and the water depth from the experiment by extending the tracking algorithm.
   Pouring aluminium oxide particles of a diameter of 1.75 ± 0.05mm in the tank and allowing water waves to transport them, resulted in the formation of a beach. In the framework of the vertical Hele-Shaw cell, gravel beach formation occurs as a result of the interaction of the incoming waves and the particles leading to bed load sediment transport. Different settings for the incoming waves can lead to different beach profiles and evolution rates.

   The parameters that varied in the experiments in the framework of this thesis were the water depth (4.9-7cm) and wave frequency (0.3-0.7Hz), indirectly influencing the wave height to be between 1-2cm, while the initial bed depth was kept relatively constant throughout all cases (3.29-4.26cm). Analysing the acquired data it was discovered that the larger the wave period, *i.e.*, the lower the frequency, the faster the bed reaches an equilibrium; an event confirmed not only by observing the tracked bed profiles but also by observing when the magnitude of the bed flux approached zero. In addition, the shape of the final bed profiles revealed a correlation to the depth between the water and the bed. It was found that in cases where the water was above the bed, the final bed profiles were taller and narrower with their peak located closer to the right wall of the tank. Equivalently, for the cases where the water was at shallower depths the beach was wider (closer to the left wall) and shorter with its peak further way from the right wall. Unfortunately, experiments were negatively influenced by the tank's incomplete design.

   The angles of the final bed profiles were also measured with the help of the data obtained from the tracking. Wet and swash regions were identified as a result of the different dynamics in effect in the Hele-Shaw cell. The wet region stayed submerged while the swash region alternately became wet and dry. The angles of the bed in both regions were computed, with the wet region angles always being smaller than the swash region angles. This was also confirmed in [90] and is attributed to a phenomenon known as "liquid bridges"; the presence of the water interfering with the cohesion properties between the particles thus preventing them from stacking at steeper slopes, and eventually collapsing. As the swash zone was not constantly wet, it allowed the cohesion properties to magnify and form steeper slopes. The angles of the wet region were then compared against angles of repose for gravel beaches (25° - 30°). The measured angles were found to be in the angle range for gravel for all cases. This finding further confirmed that the Hele-Shaw cell is a suitable set-up for observing gravel beach formation, with the aluminium oxide particles of a 1.75mm diameter imitating gravel sediments.

Breaking waves appeared in the tank and played a crucial role in transporting the sediment eastwards towards the beach, as the waves enter the tank through the left wall, thus leading to beach formation. In light of this, plunging, collapsing and surging breakers were observed, with transitions from one type to another occurring at different rates for each case.

6.3 Future work

This thesis has discussed the development of an image-analysis algorithm which can translate videos of experiments performed in a Hele-Shaw cell into useful data. The chosen mathematical models were validated against reality by using the obtained data. Despite the fact that the selected image processing method and models performed well, below can be found several directions for the work to move forward in the future.

1. **Improve the tracking precision of the image-analysis algorithm.**
   The image-analysis algorithm vertically tracks the area occupied by the bed or the wave. Each image is divided into vertical bins and each one of them is scanned with the aim to locate the area occupied by the tracking object (water or bed). The shape of water is overall easier to track than the shape of the bed. In contrast to the water, the particles forming the bed are constantly moving and can be stacked in various ways, resulting in different bed profiles. In cases where the bed morphology is more undulating

(see figure below), or the morphology more complex, the code is expected to find tracking challenging. In view of this, the tracking should be extended so as to occur both vertically and horizontally. In this way, any information lost during the vertical scanning can be recovered by scanning the same location from a horizontal perspective, therefore taking into account undulating and complex morphology. This improvement could potentially tackle the contamination of the water measurements from the porous bed.

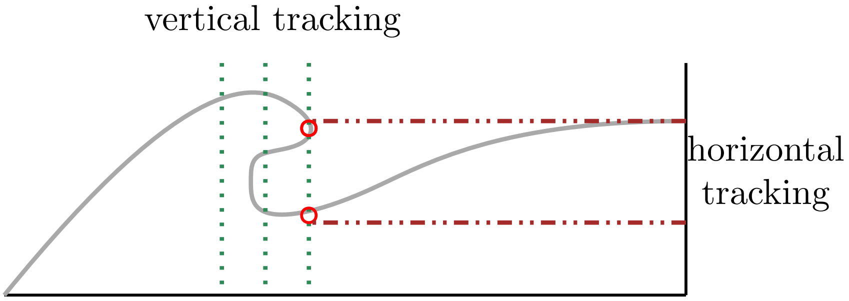


Figure 6.1: Example of undulating bottom geometry. Horizontal tracking will locate the curve under the tip of the bed that vertical tracking may miss.

2. **Introduce a higher level of automation during the image-analysis step, to further assist with the computation of the bed angles and the Iribarren number.**
   The bed angles and the Iribarren number have been determined manually in this work. This was due to several issues such as:
   - The code was not equipped yet to distinguish the wet from the swash region. As the wet bed interfered with the water measurements, it was not possible to distinguish between the wet and the swash regions of the bed in an automated way.
   - The wave steepness was not known a priori from the measurements. The Iribarren number was determined after the the bed slope and the wave height (max depth-min depth) at the toe of the bed, had been computed from the measurements. As the bed conditions and the bed shape changed over time, it would mean that one would have to track all of the above simultaneously over time, to determine the Iribarren number as a function of the wave steepness and the bed slope. The bed slope was computed in a similar, but not automated, way as the angle of repose. The wave height was determined by taking the difference in wave depth at the toe of the beach, at wave

breaking and when the water depth is at its minimum. Automating the process would thus mean that a criterion would have to be set such that the wave depth is saved when the wave is steep enough to break. Currently, there is no way of determining the wave steepness from the measurements, in a consistent and automated way.

- The coordinate systems for the water and the bed were different, thus making them more difficult to cross-refence. The bed and water coordinates had independent coordinate systems. Identifying the wet and swash regions, as well as computing the wave height at the toe of the bed, require a certain level of cross-checking between the coordinates of the two systems. Thus translating the water and bed coordinates from locally-dependent measurements, originating from the tracking, to bin-dependent coordinates that are constant over each snapshot, may help cross-reference the water and bed coordinates thus preventing the wet bed from interfering with the water tracking. This step may thus provide the basis so gradually deal with the above issues.

3. **Add moving bottom topography to the shallow-water equations.**

   The effect of a bottom topography $b(x,t)$ on the water dynamics can be modelled by adding a term in the shallow-water equations ($-gh\partial_x b$), found within the source term. Audusse [3] developed a numerical scheme for solving the shallow-water equations with a slowly varying bottom topography. The scheme is well-balanced as it can preserve steady states such as the "lake at rest" ($h+b$ =constant, $u=0$). In the case that the topography is moving [65, 80], extra equations are required; a bed evolution and a sediment transport formula. A bed evolution formula relates the bed evolution over time to the way the sediments, which form the bed, are transported:

   $$\frac{\partial b}{\partial t} + \frac{f_{\text{mor}}}{1-p}\frac{\partial q_B}{\partial x} = 0, \tag{6.1}$$

   with $b(x,t)$ the bed and p the porosity of the bed directly related to the type sediments forming the bed. q_B is a bedload sediment transport formula describing how the sediments move depending on their physical characteristics, for example their grain diameter. There are a variety of bed-load transport formulas due to currents, waves or combination of both.

4. **Improve the mathematical modelling and simulations.**

   The FV handled steep wave fronts well but there was still some disagreement between the numerical and the measured wave fronts. As mentioned earlier, this may be due to a term missing from the shallow-water equations. Including this term in the equations will introduce a prefactor $\frac{6}{5}$ to the inertia term such that $\frac{6}{5}(\mathbf{u}\nabla)\mathbf{u}$, and a non-conservative product ($(0, -0.2u)\partial_x U$) in the sense that it cannot be re-written in terms of the derivative $dF(U)/dx$ of a flux $F(U)$. Its inclusion will affect the eigenvalues and the HLL flux and is expected to slow down the bore; as a result the work done for non-conservative products will need to be employed. To this end, more information on how to handle these products and information about using the Discontinuous Galerkin finite element methods, equivalent to a one-dimensional finite volume Godunov method, can be found in [89], [81].

   Including surface tension in the FE model can introduce dispersion which can offset instabilities created from higher forcing and shallower depths. Hence a surface tension should be included in the potential-flow shallow-water model. An alternative solution could be to use the Gerris Flow Solver software [74]. A surface tension force term is added to the Eulerian Navier-Stokes equations ($\sigma k \delta_s n$). It provides a balance-force approach between the surface tension and the pressure gradient whilst allowing for a varying resolution along the interface.

   Traditional potential-flow models, cannot handle breaking waves due to the irrotational flow assumption. A two- or three-dimensional boundary element potential-flow model (wave generation and shoaling up to the point in time directly prior to when breaking) combined with a higher order time-stepping scheme and coupled to a volume of fluid (VOF) model (wave-breaking and post-breaking) could solve this problem, thus modelling wave generation, shoaling and wave breaking [38, 44]. The Laplace equation is turned into an integral equation and solved by a boundary element method, while the time-stepping provides the time level that the equation is to be solved [34] and the volume of fluid model can compute wave-breaking and post-breaking. The latter solves the Navier-Stokes equations on a fixed grid and checks the presence of fluid in each grid cell.

5. **Improve the set-up and perform further experiments**

   Throughout the experiments a significant portion of the sediments was lost at the stilling wells located at the right end of the glass plates and the wall of the cell. Placing plastic tubes into their gap was the only non-intrusive solution to stop particles from entering. Unfortunately, this did not prove to be successful. Pouring liquid plastic into the gap and allowing it to set could possibly eliminate the enclosed space, stop sediment loss and enhance the right solid-wall assumption. This type of task would require an appropriate experimental setting and knowledge, resources not available at the time of the experiments.

   The aquarium pumps that are currently used in the set-up though easy to use, have a limited frequency range of [0.3,0.7]Hz. Increasing the upper limit of this range to $1 - 1.5$Hz, by using appropriate wave pumps, would allow for the correlation between the duration and wave frequency and wave frequency and final bed profiles to be investigated.

   The performed experiments started with a similar initial bed profile and depth, while the water depth varied from shallow to deep. These experiments resulted into different final bed profiles, hence revealing a correlation between the latter and the difference in depth between the water and the bed ($h_w - h_b$). The final bed profile of case 232 in particular, changed continuously up until the end of the recording. This was a result of the medium water depth which did not allow for the bed to reach a final quasi-equilibrium sooner, and should be thus investigated further. Additionally, the tank's lost sediment did not allow for accurate measurements of the bed depth to be acquired. Improving the set-up and investigating this difference further, by varying the water and the bed depth, could provide insight not only on the different types of beaches that can exist in this set-up, such as a dry-beach, a wet-beach or a dune-beach, but also on the timescales that lead to these beach types.

   Lastly, the effect of the initial bed slope on the breaker types and the final bed profiles should be further investigated. During the experiments presented in this thesis, the initial bed profile was horizontal and uniform for all experiments. Its effect on the final bed profiles was not evidenced by any of the current data, while the observed breakers varied from plunging to surging. Changing the initial bed slope, and keeping the rest of the settings the same as before, may result in some breaker types being formed more often than others, while investigating closely whether the initial slope has any effect on the final bed profiles.

Appendices

A Water tracking

The monochromatic edge detection, originally used as an alternative method to translate images from tank one to data in [42] is presented in Section A.1. Then, in an effort to confirm the superiority of the colour-channel-detection results, a comparison between the edge-detection and the colour-channel-detection method is performed for tank two (case 154) in Section A.2.

A.1 Tracking the water line with monochrome edge detection

A.1.1 Image conversion to greyscale and background removal

Monochrome edge detection was preferred to coloured edge detection due to its easy-to-do implementation and manipulation of the greyscale image matrices. In addition, as the background light in that set of experiments was not uniform, coloured edge detection was considered of being at risk of picking up more edges near the change of colour hence creating more problems than it could solve. Monochrome edge detection meant that the images had to turn to greyscale before any further processing could occur. This process was also done in Matlab iteratively with each coloured image being imported, turned to grayscale through manipulating its image matrix, and then being exported as greyscale image.

The presence of a “static” background in the images interfered with the processing by adding more edges to the snapshots in question, hence making it harder to distinguish the free surface line. As the lighting in the background was not uniform, the background interfered with the tracking and a consistent way of ignoring its effects was nonexistent. In order for the static background to be removed, the matrix of one snapshot A_0 was subtracted from the matrices of each of the loaded snapshots A_i. In this way, the final image matrix $B_i = A_i - A_0$ would have a zero intensity at the location of the pixels corresponding to the background, with only the free surface lines at those two snapshots visible (see Figure A.3). By analysing the snapshots it was discovered that having a constant A_0 was not rendering optimal results, therefore a different snapshot selection strategy was chosen. It was found by trial and error that if A_0 was not fixed but in a close neighbourhood of the main snapshot A_i, then better results were obtained, as the tracked free surface lines were very similar to each other and not interfering with each other. Hence choosing A_0 to be four snapshots after A_i, with $A_0 \equiv A_{0_i} = A_{i+4}$, resulted in a more accurate detection of the water line.

A.1.2 Monochrome edge detection operators and the Sobel filter

The gradient method detects the discontinuities in brightness by looking for the maxima of the first derivative or zero-crossings in the second derivative of an image [72]. These discontinuities then reveal the edges of the image that aim to separate the object of interest from the background. The image's discrete derivative is approximated by applying horizontal and vertical linear filters/masks on the source image as shown in Figure A.2, and originally depicted in [18]. The size and operators of these filters vary depending on the selected edge detection method. For example some first order edge-detection operators applied at the first derivative of a monochrome or colour image are the Sobel, Prewitt and Roberts operators [18], [33]. The Laplacian and Mexican Hat operators are the second-order edge-detection operators [72].

The Sobel filter was ultimately selected for tracking the water line. The advantage that came with this filter was that it reduced the noise sensitivity, hence providing a smoothing effect in the image that none of the other operators could offer. This smoothing effect originated from the discrete derivative, a weighted summation of the filter and the nine-pixel values. Hence the derivative was computed in a neighbourhood of nine pixels whose intensity values did not vary much [18].

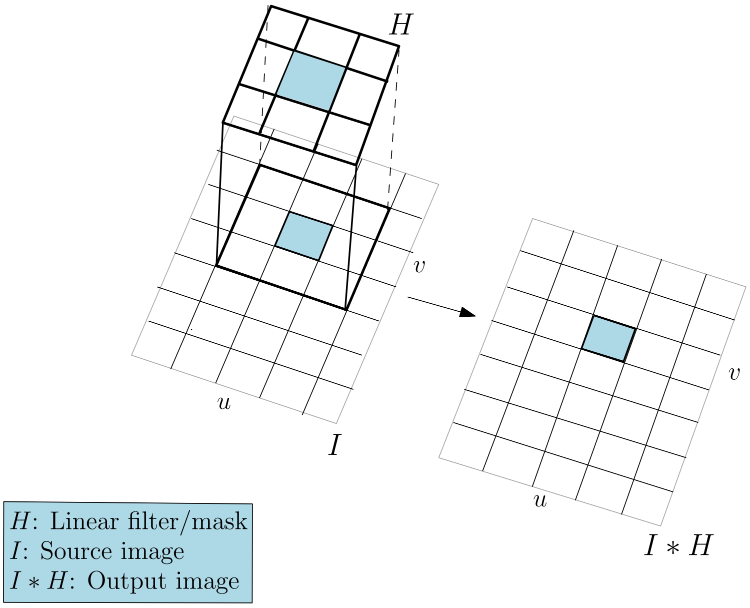


Figure A.2: The convolution of the source image I with the Sobel filter H results to the image on the right. The filter occupies nine pixels in the source image, three adjacent horizontally and vertically, to compute the derivative for a single pixel of the output image.

The convolution of image I with the Sobel mask, expressed in equation (A.2) and visualised in Figure A.2, produced an output image. The minimum and maximum gradient magnitude was then calculated with the help of equation (A.3). The first-order image derivative as defined in [18] is the following, with D_x, D_y denoting the discrete-in-space derivatives:

$$\begin{aligned}\nabla I(u,v) &\approx \frac{1}{8}(D_x(u,v), D_y(u,v))\\ &= \frac{1}{8}\Big((I * H_x^S)(u,v), (I * H_y^S)(u,v)\Big)\\ &= \frac{1}{8}\Big(\sum_{i=-1}^{1}\sum_{j=-1}^{1} I(u+i,v+j) * H_x^S(i,j), \sum_{i=-1}^{1}\sum_{j=-1}^{1} I(u+i,v+j) * H_y^S(i,j)\Big).\end{aligned} \tag{A.2}$$

The Sobel mask for the horizontal and vertical direction is:

$$H_x^S = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, H_y^S = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

The gradient magnitude $M(u,v)$ is:

$$M(u,v) = \sqrt{D_x^2 + D_y^2} \approx |(D_x)| + |D_y|. \tag{A.3}$$

It is less costly to use the approximation for the magnitude without losing the intensity information. After the filter was applied, the matrix of each snapshot was updated with the magnitude values that the filter indicated.

A.1.3 Thresholding

A threshold value was then determined, based on the known minimum and maximum gradient magnitudes. This threshold value was then used as a criterion to transform the greyscale image to a binary image; in a similar way to the colour-channel-detection method. The binary image, visualised in Figure A.3, would then reveal the water line with white and everything else depicted with black. Having the water line traced, paved the way for tracking its coordinates in the same way as was described in Section 2.3.4. Figures A.4 and A.5 illustrate the tracked free surface and the obtained data that were used in [42].

Figure A.3: Binary image depicting the tracked water line (white) and background (black). Two free surface lines are depicted originating from snapshots A_i, A_{i+4}.

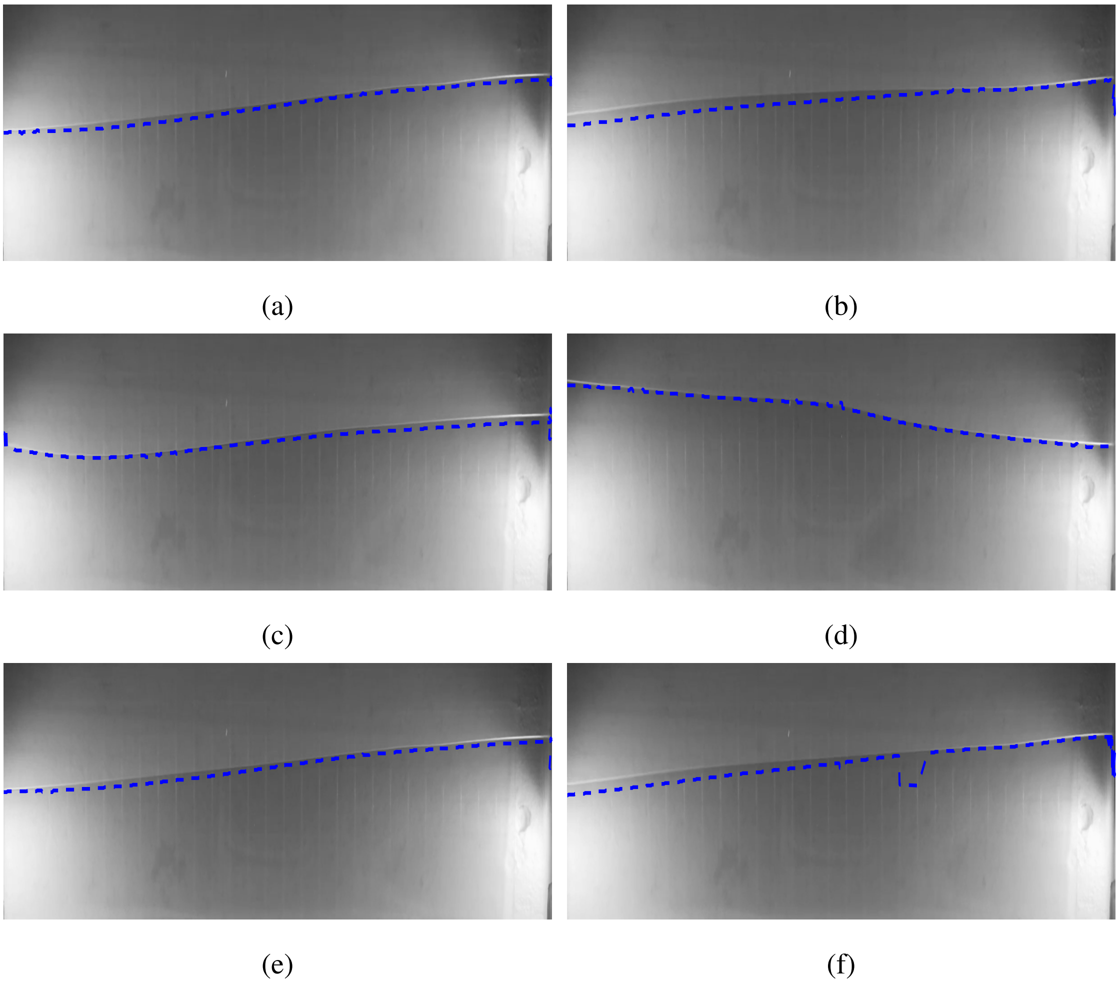
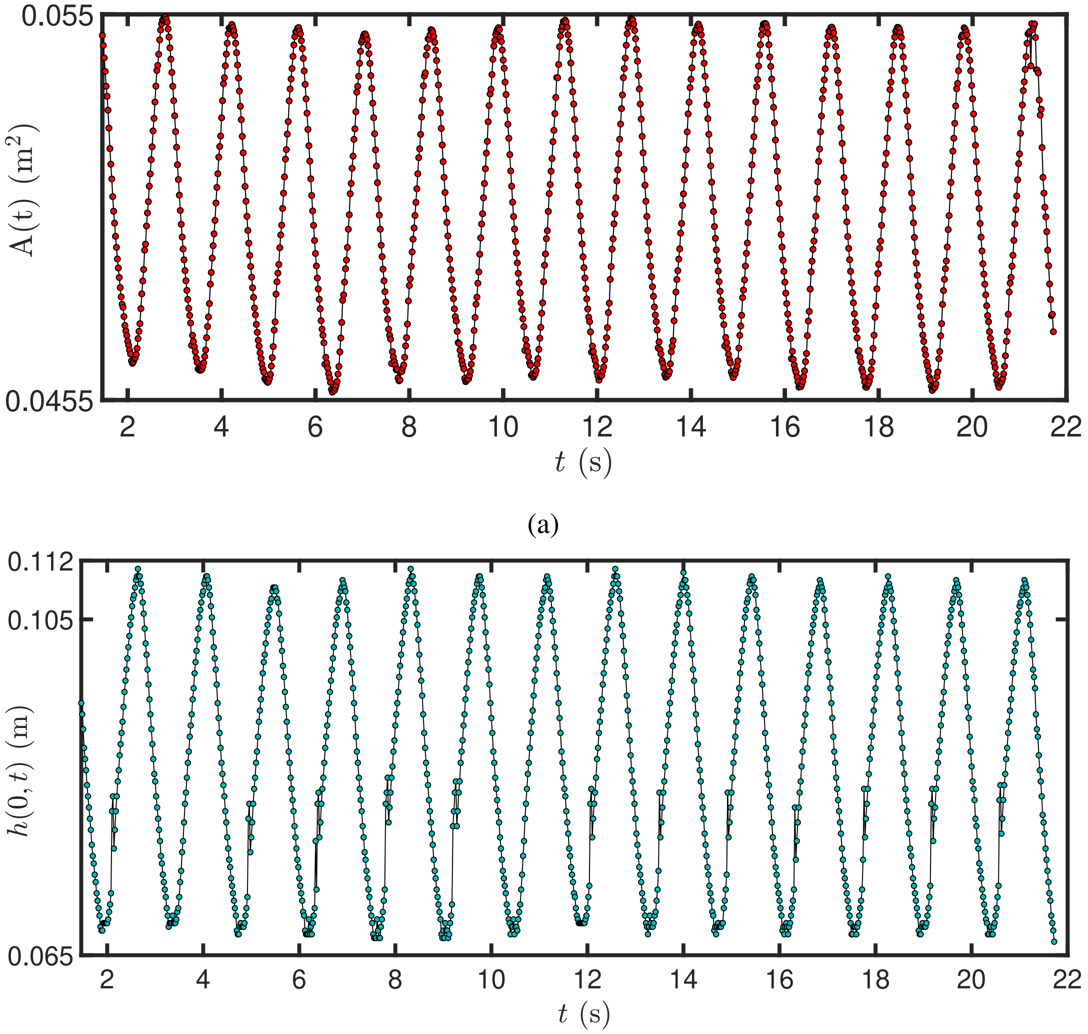

(a) (b) (c) (d) (e) (f)

Figure A.4: Tracked and reproduced water line for video 47 at times $t = 1.96\text{s}, 3.2\text{s}, 6.4\text{s}, 12.6\text{s}, 19\text{s}, 21.72\text{s}$. Wave period is $T = 1.42\text{s}$.

It can be deduced from the above figures that even though the data are of a sinusoidal shape, they underestimate the free surface line.



(a)

(b)

Figure A.5: Area (top panel) and depth at the left wall (bottom panel) data. This data set originated from case 47 (tank one) and was used in [42].

A.2 Comparison of the methods

The tracking methods had initially been applied to snapshots originating from different tanks with different light and camera settings. Hence, comparing the two methods within the same set-up further tested their performance. Video 154 was chosen for the method comparison, as it provided the least steep waves. A summary of each detection process has been illustrated in Figure A.6, with the tracking results overlaid against the snapshots in Figure A.7.

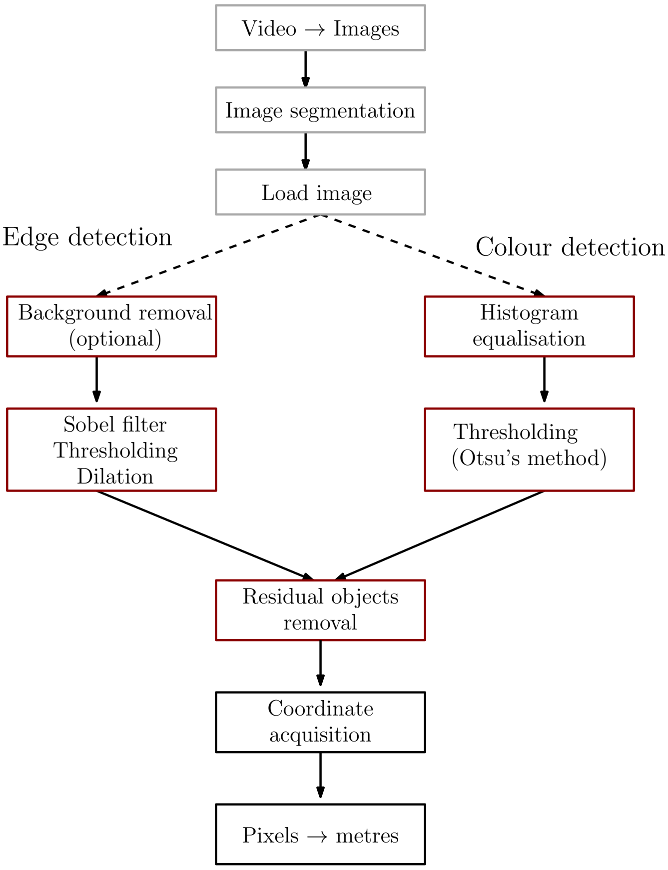
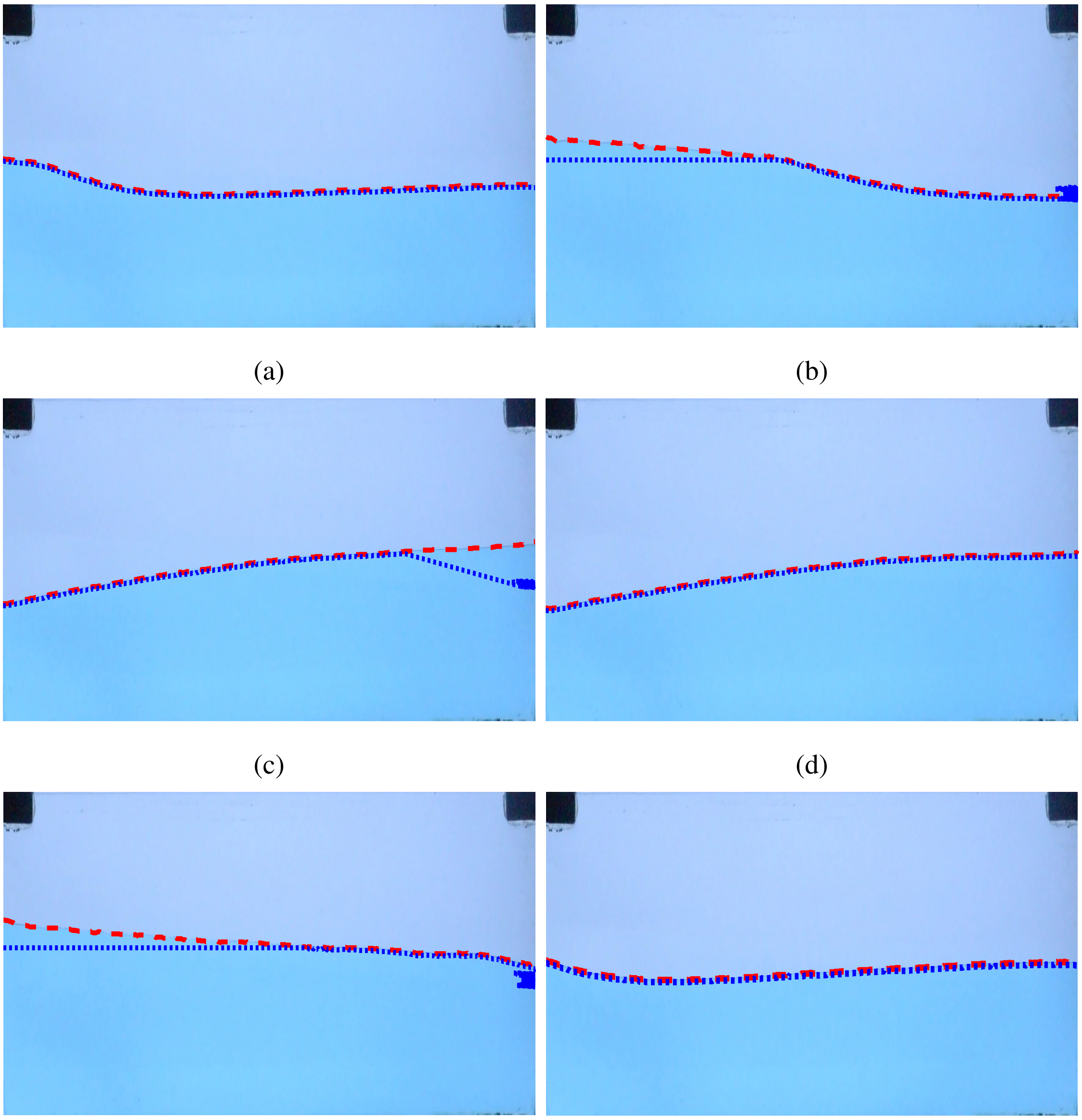


Figure A.6: Summary of the detection methods. Pre-processing steps (grey) lead to the main processing stage (red) for each tracking method. Lastly, the image analysis stage follows (black). The tracking methods read the same snapshots and only differ during the main processing stage.



(a) (b)

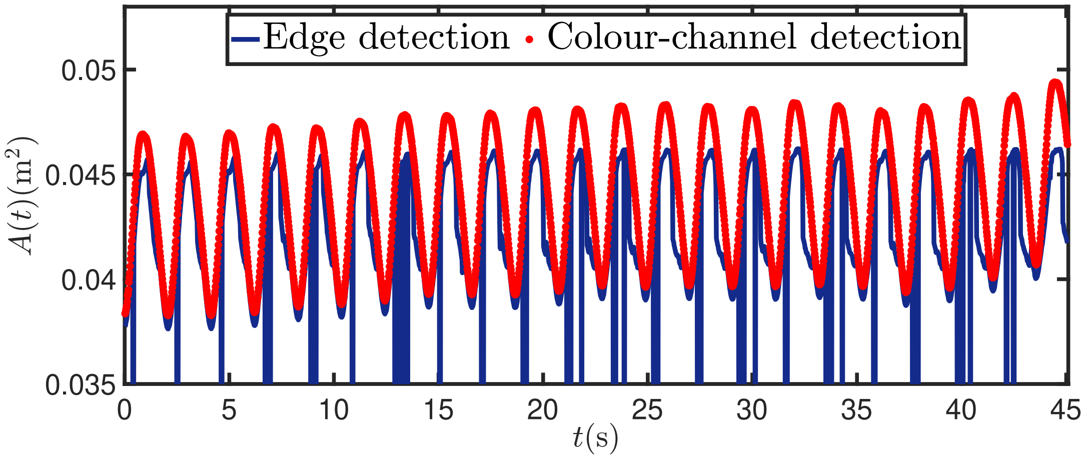
(c) (d)

(e) (f)

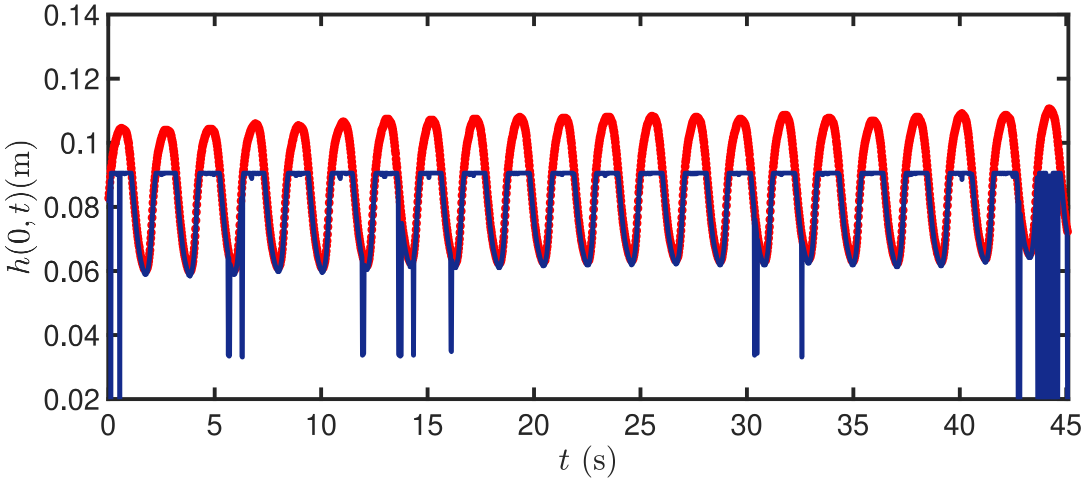
Figure A.7: Comparison between edge-detection (blue dots) and colour-channel-detection (red dash) results for video 154 at times $t = 4.28\text{s}, 4.6\text{s}, 7.82\text{s}, 7.98\text{s}, 11.14\text{s}, 26.76\text{s}$. Wave period is $T = 2\text{s}$. The fully tracked (colour-channel-detection) video can be found here: `https://www.youtube.com/watch?v=UxXw332EpEY`.

It can be seen in Figure A.7 that the colour-channel-detection method overall performed better than the edge-detection method. Both methods performed equally well when the water was either entering the tank (Figure A.7a) or reflecting on the right wall (Figure A.7d). The rest of the time when the waves were propagating in the tank, edge detection in principle underestimated the free surface line, while colour-channel detection remained accurate. It is believed that the use of the monochromatic snapshots during the edge-detection method, was partly the reason for the inaccurate detection of the free surface line. As illustrated in the RGB cube in Figure 2.4, greyscale lies in the diagonal between white and grey. The RGB intensity values thus are all halved. Hence, not much of a difference between the water and the background intensities was achieved. As a result, the threshold was difficult to determine with similar intensity values found throughout the image matrix.

An additional problem was the fact that edge detection tracked solely the free surface line based on the intensity discontinuities. The uniformity of the intensity values combined with tracking only a line, instead of a whole region like with colour-channel detection, made the tracking prone to more detection errors. The time series provided by each method have been compared against each other in the following figure.



(a)



(b)

Figure A.8: Comparison of area (top panel) and depth (bottom panel) data, acquired by applying the edge-detection (blue) and colour-channel-detection (red) method to images originating from video 154.

It can be observed in Figure A.8 that the colour-channel-detection method performs overall much better than edge-detection method. The overlaid snapshots in Figure A.7 revealed that edge detection was accurate only when the water was at its maximum depth and was either inserting the tank or reflecting on the right wall. During the rest of the stages when the water propagated in the tank, the edge detection in principle underestimated the water depth. Figure A.8b confirms this statement, as the values for the depth computed from edge detection are much smaller than those from colour-channel detection. This disagreement was then passed on to the area data, through the area computation described in equation (2.2). As a result, the edge-detection area data had a smaller amplitude and deviated from the sinusoidal profile of the colour-channel-detection area data.

B Numerical implementation of the finite-volume method

In this section, the Godunov method (Section B.1) and the HLL flux (Section B.2) have been further described.

B.1 The Godunov method

Integrating equation (3.11) over one space- and one time step reads:

$$\int_{t_n}^{t_{n+1}} \int_{x_{k-1/2}}^{x_{k+1/2}} \partial_t U \, dx \, dt + \int_{t_n}^{t_{n+1}} \int_{x_{k-1/2}}^{x_{k+1/2}} \partial_x F(U) \, dx \, dt = \int_{t_n}^{t_{n+1}} \int_{x_{k-1/2}}^{x_{k+1/2}} S \, dx \, dt. \tag{B.4}$$

If superscripts denote time and subscripts space, simplifying equation (B.4) yields:

$$\begin{aligned} &\int_{x_{k-1/2}}^{x_{k+1/2}} U(x, t^{n+1}) - U(x, t^n) \, dx + \int_{t_n}^{t_{n+1}} F(U(x_{k+1/2}, t)) - F(U(x_{k-1/2}, t)) \, dt \\ &= \int_{t_n}^{t_{n+1}} \int_{x_{j-1/2}}^{x_{j+1/2}} S(x, t) \, dx dt. \end{aligned} \tag{B.5}$$

The state vector U, flux F and source term S are averaged over one cell k and one time step $t_{n+1} - t_n$. A first-order approximation in time is achieved thus resulting to a Forward Euler scheme. The cell-averages are the following:

$$U_k^n = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} U_k(t) \, dt, \text{ with } U_k(t) = \frac{1}{|\Delta x_k|} \int_{x_{k-1/2}}^{x_{k+1/2}} U(x, t) \, dx, \tag{B.6}$$

$$S_k^n = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} S_k(t) \, dt \text{ with } S_k(t) = \frac{1}{|\Delta x_k|} \int_{x_{k-1/2}}^{x_{k+1/2}} S(x, t) \, dx. \tag{B.7}$$

The flux is computed at the respective edges of each cell with $k \to k - 1$ for the left cell and $k \to k + 1$ for the right cell:

$$F_{k\pm1/2} = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} F(U(x_{k\pm1/2}, t)) \, dt. \tag{B.8}$$

Substituting equations (B.6)-(B.8) in equation (B.5) and rearranging the terms yields the discretised equation:

$$U_k^{n+1} = U_k^n - \frac{\Delta t}{\Delta x_k}\left[F_{k+1/2} - F_{k-1/2}\right] + \Delta t\, S_k^n. \tag{B.9}$$

The flux F is then replaced by the HLL numerical flux described in the following section.

B.2 Harten, Lax, and van Leer (HLL) flux

The HLL approximate Riemann solver [47], [92] has been used to calculate the intercell flux contribution during each step of the computation of U_k^{n+1} for $k = 00, 1, ..., NN$ number of subintervals at time $n+1$. It essentially solves local Riemann problems not exactly, which can be computationally expensive, but approximately by using a wave configuration that allows for two waves to exist; thus determining three states and ignoring any intermediate waves [91].

Estimates of the wave speeds of these two waves are needed; these are determined by the eigenvalues of the problem $\lambda_\pm = u \pm \sqrt{gh}$. Essentially the values of the lower (S_L,) and upper bound (S_R) of the wave speeds are computed locally, in the sense that they use information from a left and a right state $U_L = (h_L, h_L u_L)$, $U_R = (h_R, h_R u_R)$. Information from each state enters the eigenvalues thus determining these two bounds as follows: $S_L = \min(\lambda_{-L}, \lambda_{-R})$, $S_R = \max(\lambda_{+L}, \lambda_{+R})$; note that $S_L < S_R$, $\lambda_{\pm L} = u_L \pm \sqrt{gh_L}$ and $\lambda_{\pm R} = u_R \pm \sqrt{gh_R}$. Three wave patterns/states can occur as a result of the values of S_L, S_R: a right supersonic wave ($S_L > 0$), a subsonic wave ($S_L \leq 0 \leq S_R$) and a left supersonic wave ($S_R < 0$). Based on the wave pattern that prevails, the numerical flux is determined exactly as follows:

$$F_{k+1/2} = \begin{cases} F_L & \text{if } S_L > 0 \\ \frac{S_R F_L - S_L F_R + S_L S_R (U_R - U_L)}{S_R - S_L} & \text{if } S_L \leq 0 \leq S_R, \\ F_R & \text{if } S_R < 0. \end{cases}$$

A summary of the computation of wave speeds and HLL flux for each moment in time has been sketched below.

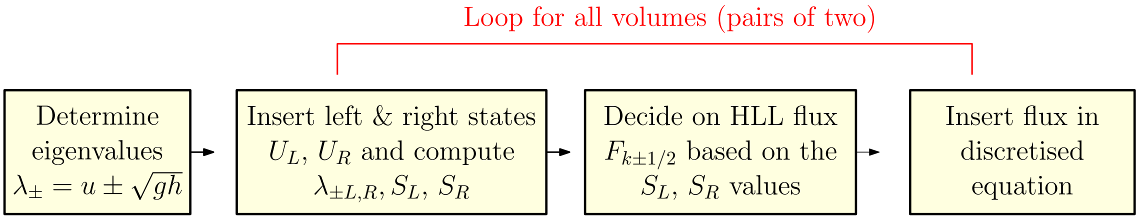


Figure B.9: Process of determining the HLL flux. Initially the eigenvalues are computed. For each moment in time, pairs of cells U_L, U_R feed their values of u, h to the eigenvalues. Then the two eigenvalues are used to calculate the lowest and highest wave speed. These speeds ultimately determine which one of the three wave states is in effect (right supersonic, subsonic, left supersonic). For the wave state in effect, the corresponding HLL flux $(F_{k+1/2}, F_{k-1/2})$ is computed exactly at the edges of each cell and is fed into the discretisation equation (B.9).

C Models verification against exact linear solutions

When one has finished the implementation of a mathematical model the next step is to check the work that has been carried out correctly and potentially locate any implementation errors. To this end exact wave solutions of the linearised potential-flow shallow-water equations, i.e. solutions that are known to be true under specific conditions, have been used as a means of verifying that each code is behaving in accordance to the physical properties for which the exact solution is true. In this section, the structure is as follows: the derivation of the exact solutions from the linearised equations is presented in Section C.1, where two cases have been considered. A standing wave and a forced-wave solution. The exact solutions are compared against the numerical solutions of the linear models. In Section C.2, the imposed boundary conditions for the FV method are provided and a comparison between the FV and the FE method follows for each exact solution.

C.1 Derivation of the exact solutions

The linearised potential-flow shallow-water equations, in the absence of damping, read:

$$\partial_t \phi + \eta = 0, \tag{C.10a}$$

$$\partial_t \eta + H_0 \partial_{xx} \phi = 0, \tag{C.10b}$$

with deviation from the depth at rest $\eta << 1$, and a perturbation to the velocity potential at rest $\phi << 1$.

Equations (C.10) can be merged into one differential equation. This can be accomplished by differentiating equation (C.10a) over time and replacing the term $\partial_t \eta$ with its equal from equation (C.10b) yielding:

$$\partial_{tt} \phi - H_0 \partial_{xx} \phi = 0. \tag{C.11}$$

One can assume solutions of (C.11) of the form:

$$\phi = \big(A \cos(kx) + B \sin(kx)\big) \sin(\omega t), \tag{C.12}$$

with k the wavenumber and ω the angular wave frequency. Two types of exact solutions have been derived. A standing wave solution with solid-wall boundary conditions at both walls and a forced wave solution with an inflow boundary condition on the left wall and a solid-wall boundary condition at the right wall. For each case, solutions for u and ϕ perturbation are determined.

C.1.1 Standing wave

For the standing wave, the solid-wall boundary conditions leads to $\partial_x \phi|_{x=0}, \partial_x \phi|_{x=L} = 0$. Applying the boundary conditions sets $B = 0$ and provides a wavenumber $k = 2m\frac{\pi}{L}$, hence the solution is $\phi = A \cos(kx) \sin(\omega t)$. Substituting this into equation (C.11) yields the dispersion relation $\omega^2 = H_0 k^2$. Substituting ϕ in equation (C.10b) and solving for η yields $\eta = -\frac{A H_0 k^2}{\omega} \cos(kx) \cos(\omega t)$. Hence for $C = -\frac{A H_0 k^2}{\omega}$:

$$\boxed{h = H_0 + C \cos(kx) \cos(\omega t)}. \tag{C.13}$$

Going back to equation (C.10a), substituting η and integrating in time for ϕ gives:

$$\boxed{\phi = -\frac{C}{\omega}\cos(kx)\sin(\omega t)}. \tag{C.14}$$

An exact solution for $u = \partial_x\phi$ can be recovered by differentiating equation (C.14):

$$\boxed{u = \partial_x\phi = \frac{C\omega}{H_0 k}\sin(kx)\sin(\omega t)}. \tag{C.15}$$

C.1.2 Forced wave

For the forced wave case an inflow boundary condition has been imposed at the left wall $hu|_{x=0} = Q(t)$, translated into $H_0 u|_{x=0} = H_0\partial_x\phi|_{x=0}$ and a solid-wall at the right wall $\partial_x\phi|_{x=L} = 0$ as before. The area flux is taken as a sinusoidal time-dependent function of amplitude q of the form $Q(t) = q\sin(\omega t)$.

Applying the boundary conditions provides the following expressions for the constants A, B: $B = \frac{q}{kH_0}$, $A = \frac{B\cos(kL)}{\sin(kL)}$. Substituting A in ϕ gives $\phi = B\frac{\cos(k(x-L))}{\sin(kL)}\sin(\omega t)$ and substituting ϕ in equation (C.11) provides the dispersion relation $\omega^2 = H_0 k^2$, same as before. Solving equation (C.10b) for η gives $\eta = -H_0\frac{Bk^2}{\omega}\frac{\cos(\omega t)}{\sin(kL)}\cos(k(x-L))$. Substituting B gives:

$$\boxed{h(x,t) = H_0 - \frac{qk}{\omega}\cos(\omega t)\frac{\cos(k(x-L))}{\sin(kL)}}. \tag{C.16}$$

Solving for ϕ knowing η gives:

$$\boxed{\phi = \frac{q}{H_0 k}\sin(\omega t)\frac{\cos(k(x-L))}{\sin(kL)}}. \tag{C.17}$$

The wavenumber is $k = \frac{(2m+1)}{2}\frac{\pi}{L}$. An exact solution for $u = \partial_x\phi$ can be recovered by differentiating equation (C.14):

$$\boxed{u = \partial_x\phi = -\frac{q}{H_0}\sin(\omega t)\frac{\sin(k(x-L))}{\sin(kL)}}. \tag{C.18}$$

C.2 Boundary conditions and results

The exact solutions for the linear problem have been compared against the numerical solutions of the linear shallow-water equations. Each exact solution has been plotted against the deviation from the depth at rest ($\eta = h - H_0$) and the velocity u at various times.

C.2.1 Standing waves

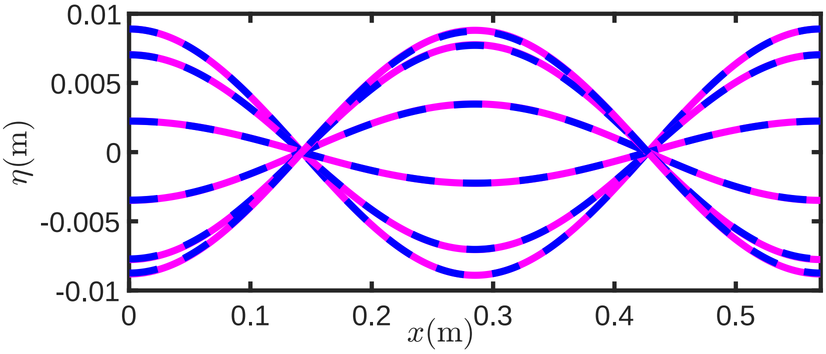
The boundary conditions used are:

a solid-wall on the left :

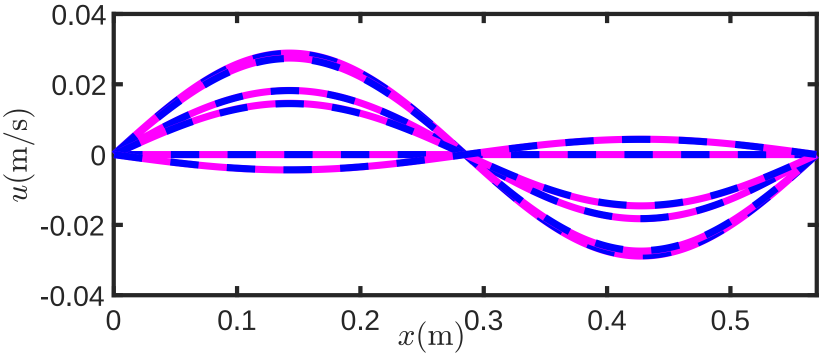
$$U_{00} = \begin{bmatrix} h_{00} \\ h_{00}u_{00} \end{bmatrix} = \begin{bmatrix} h_1 \\ -h_1 u_1 \end{bmatrix},$$

a solid-wall on the right:

$$U_{Nvol+1} = \begin{bmatrix} h_{NN} \\ h_{NN}u_{NN} \end{bmatrix} = \begin{bmatrix} h_{Nvol} \\ -h_{Nvol}u_{Nvol} \end{bmatrix}.$$



(a) Comparison for $\eta = h - H_0$.



(b) Comparison for u.

Figure C.10: Verification of the FV linear numerical solution (blue dash) for the water elevation $\eta = h - H_0$ (top panel) and velocity u (bottom panel) against the standing-wave solution (magenta) of the linear problem. 800 elements have been used.

It can be seen the numerical linear solution is in agreement with the exact standing wave solution. The numerical model is thus verified and the forcing wave case is further examined next.

C.2.2 Forced wave

The boundary conditions used are:

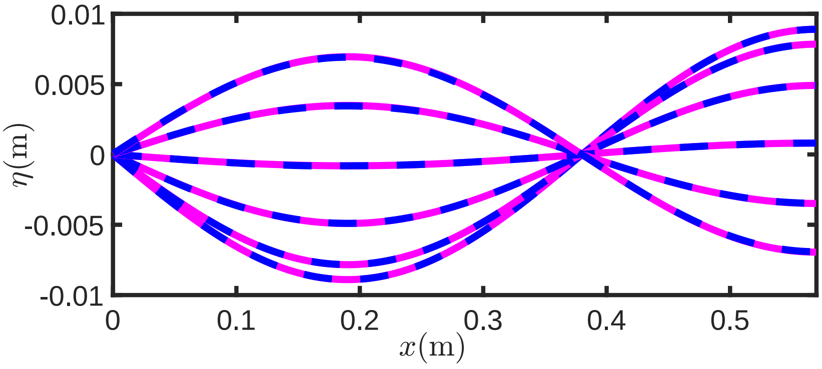
in- and outflow on the left wall:

$$U_{00} = \begin{bmatrix} h_{00} \\ h_{00}u_{00} \end{bmatrix} = \begin{bmatrix} h_1 \\ Q \end{bmatrix},$$

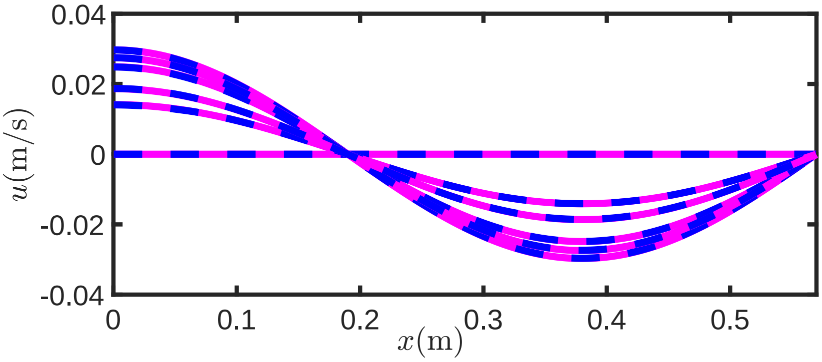
a solid-wall on the right:

$$U_{Nvol+1} = \begin{bmatrix} h_{NN} \\ h_{NN}u_{NN} \end{bmatrix} = \begin{bmatrix} h_{Nvol} \\ -h_{Nvol}u_{Nvol} \end{bmatrix},$$

with q the influx amplitude, as defined earlier. The comparison between the exact linear forced-wave and linear numerical solution ensues.



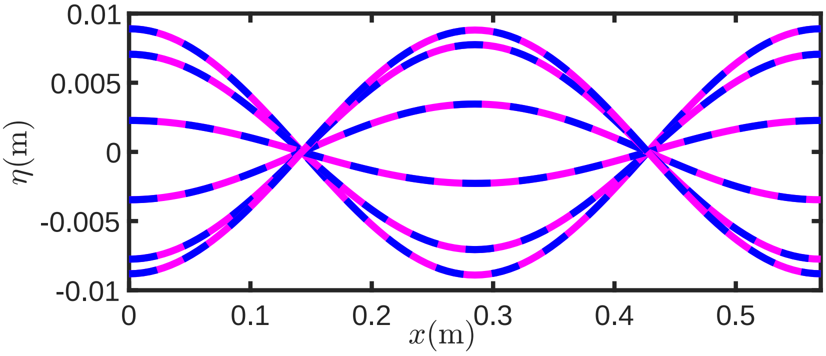
(a) Comparison for $\eta = h - H_0$.



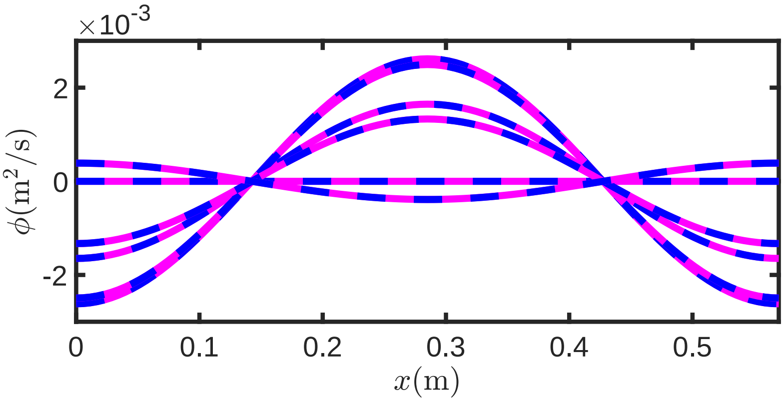
(b) Comparison for u.

Figure C.11: Verification of the FV linear numerical solution (blue dash) for the water elevation $\eta = h - H_0$ (top panel) and velocity u (bottom panel) against the forced-wave solution (magenta) of the linear problem. 800 elements have been used.

The comparison of the linear exact solutions with the linear finite-element solution, found in [42], has been reproduced below for closure for the same values as for the FV case. Similarly to the finite-volume method, the standing-wave case is initially presented. Then, the forced-wave case follows and momentum dissipation is zero as before.



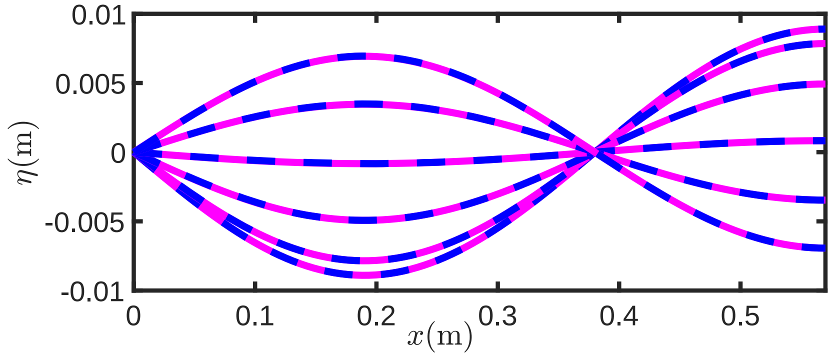
(a) Comparison for $\eta = h - H_0$.



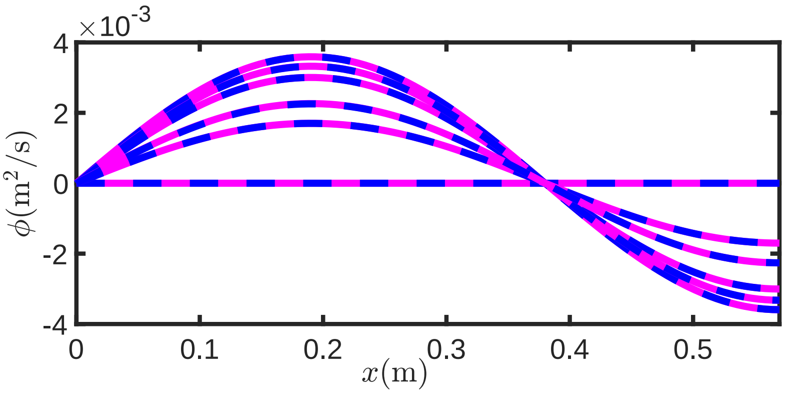
(b) Comparison for ϕ.

Figure C.12: Verification of the FE linear numerical solution (blue dashed) for the water elevation $\eta = h - H_0$ (top panel) and velocity potential ϕ (bottom panel) against the standing-wave solution (magenta) of the linear problem. 100 elements have been used.

It can be seen that there is a good agreement between the numerical and the standing wave solution. The solid wall boundary conditions that are in effect did not need to be implemented, as for the FV method. Furthermore, the same results are obtained from both numerical methods (Figures C.10a and C.12a). The comparison of the numerical and the forced wave solution of the linear problem follows.



(a) Comparison for $\eta = h - H_0$.



(b) Comparison for ϕ.

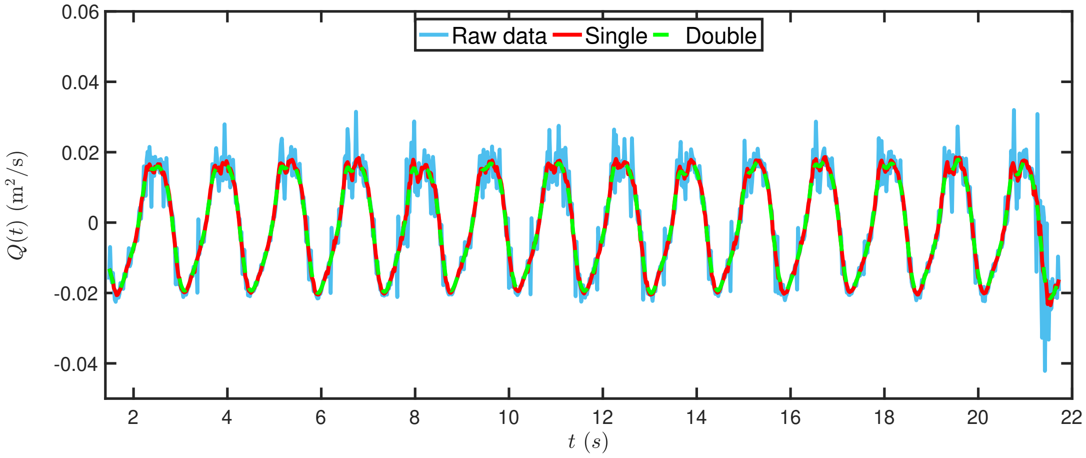
Figure C.13: Verification of the FE linear numerical solution (blue dashed) for the water elevation $\eta = h - H_0$ (top panel) and velocity potential ϕ (bottom panel) against the standing-wave solution (magenta) of the linear problem. 100 elements have been used.

The exact and forced linear solutions of the problem are in good agreement with each other. The inflow boundary condition was implemented for the inflow in a similar way as for simulation one and simulation two. In addition, Figures C.11a and C.13a reveal that the numerical methods yield the same results for the linear models.

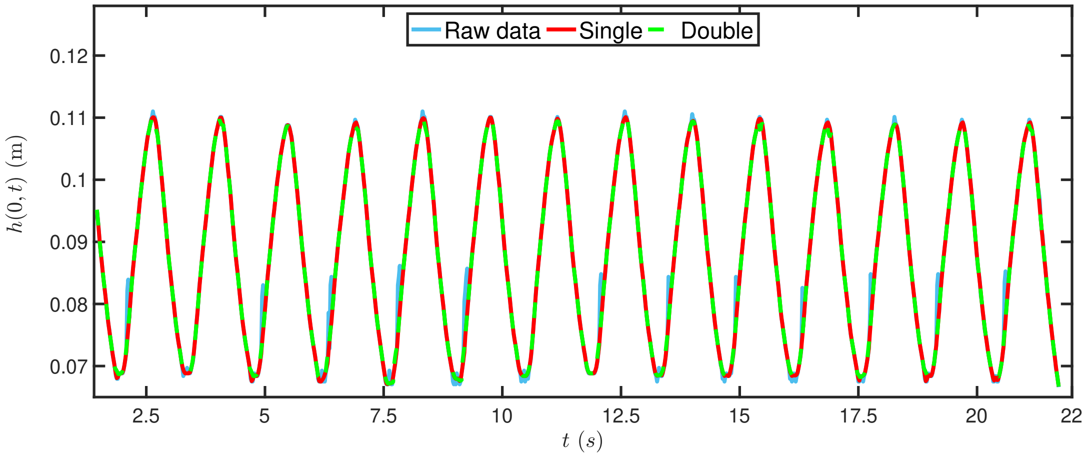
D Further data processing and numerical results

The (smoothed) experimental data from videos 47,154 are provided in Section D.1, with functions for the remaining Q data fitted in Section D.2. Numerical results for sim1 video 47 are presented in Section D.3. Lastly, the computation of the numerical wave speeds for cases 149,165 follows.

D.1 Smoothed data

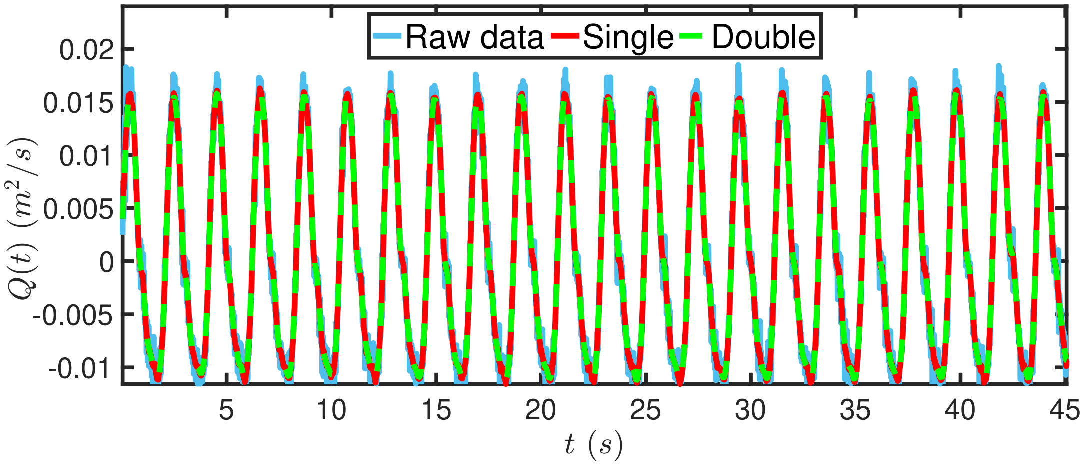


(a) Inflow $Q(t)$.

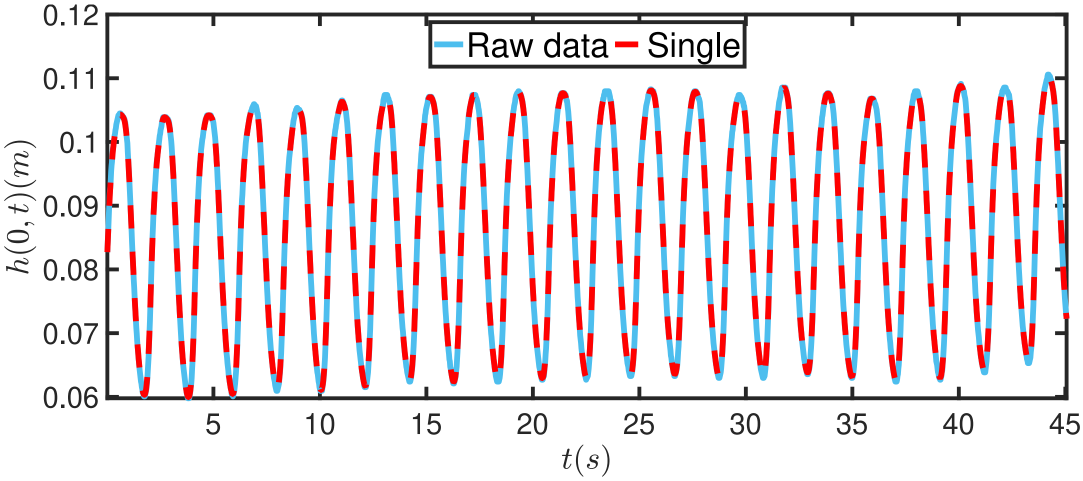


(b) Depth at the left wall $h(0, t)$.

Figure D.14: Comparison between double (green dash), single (red) smoothed and raw data (light blue) for $Q(t)$ (top panel), $h(0, t)$ (bottom panel), case 47 (tank one).

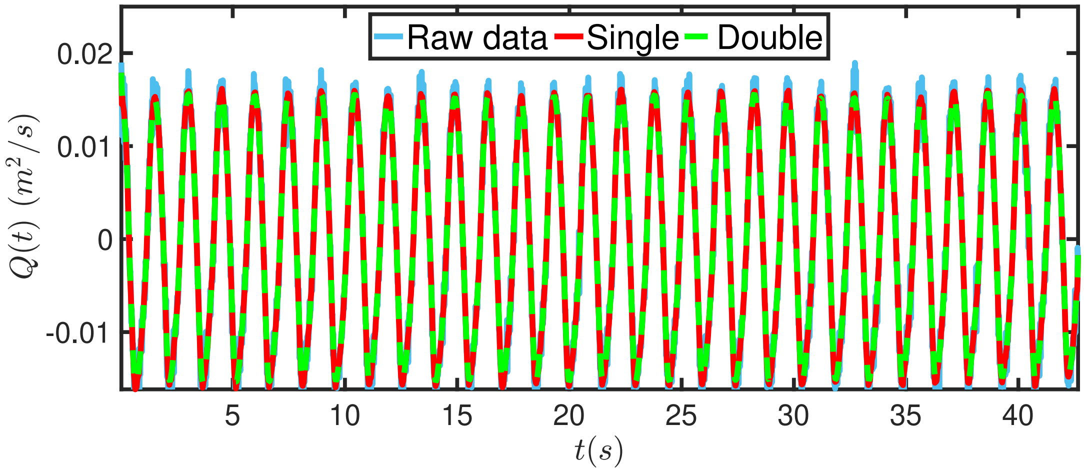


(a) Inflow $Q(t)$.

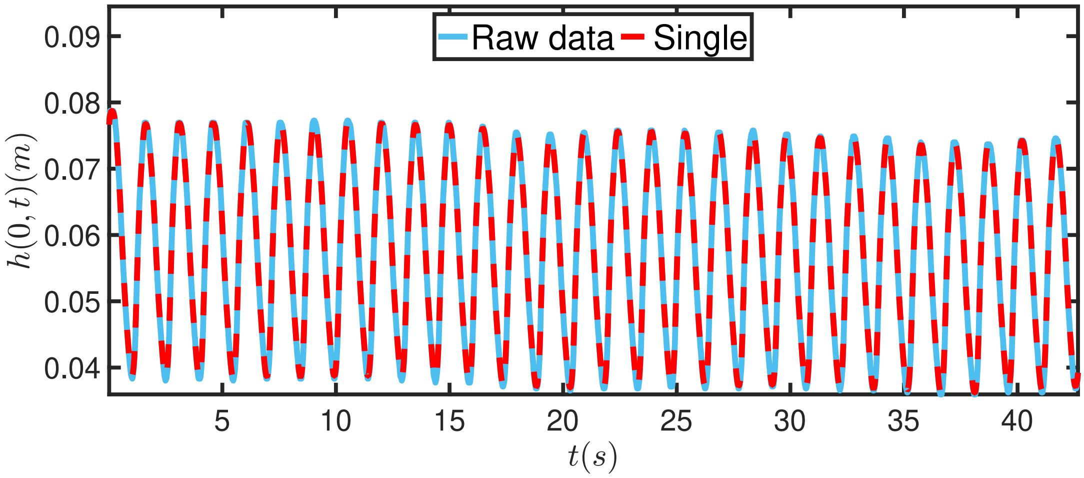


(b) Depth at the left wall $h(0,t)$.

Figure D.15: Comparison between double (green dash), single (red) smoothed and raw data (light blue) for $Q(t)$ (top panel), $h(0,t)$ (bottom panel), case 154 (tank two).



(a) Inflow $Q(t)$.



(b) Depth at the left wall $h(0,t)$.

Figure D.16: Comparison between double (green dash), single (red) smoothed and raw data (light blue) for $Q(t)$ (top panel), $h(0,t)$ (bottom panel), case 149 (tank two).

D.2 Fitted functions

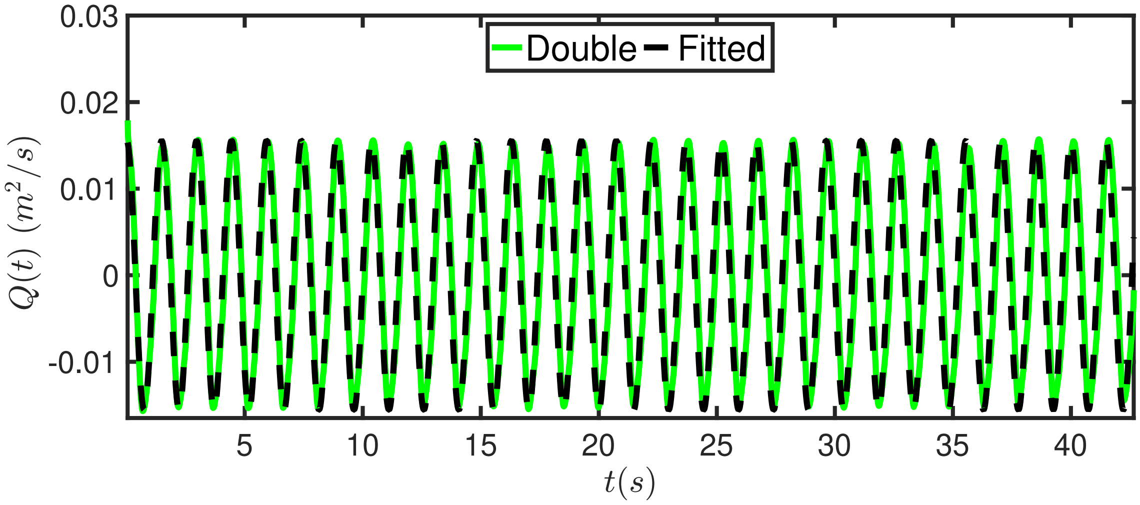


Figure D.17: Case 149: Double smoothed data for $Q(t)$ (green) compared with the fitted function $Q_{\text{fitted}}(t) = 0.01569\cos(2\pi * 0.675t)$ (black).

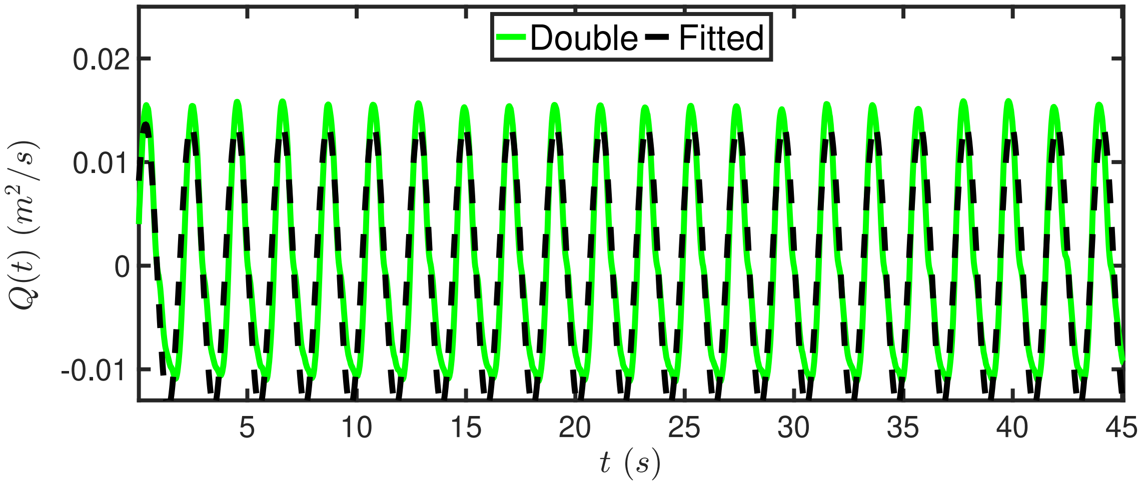
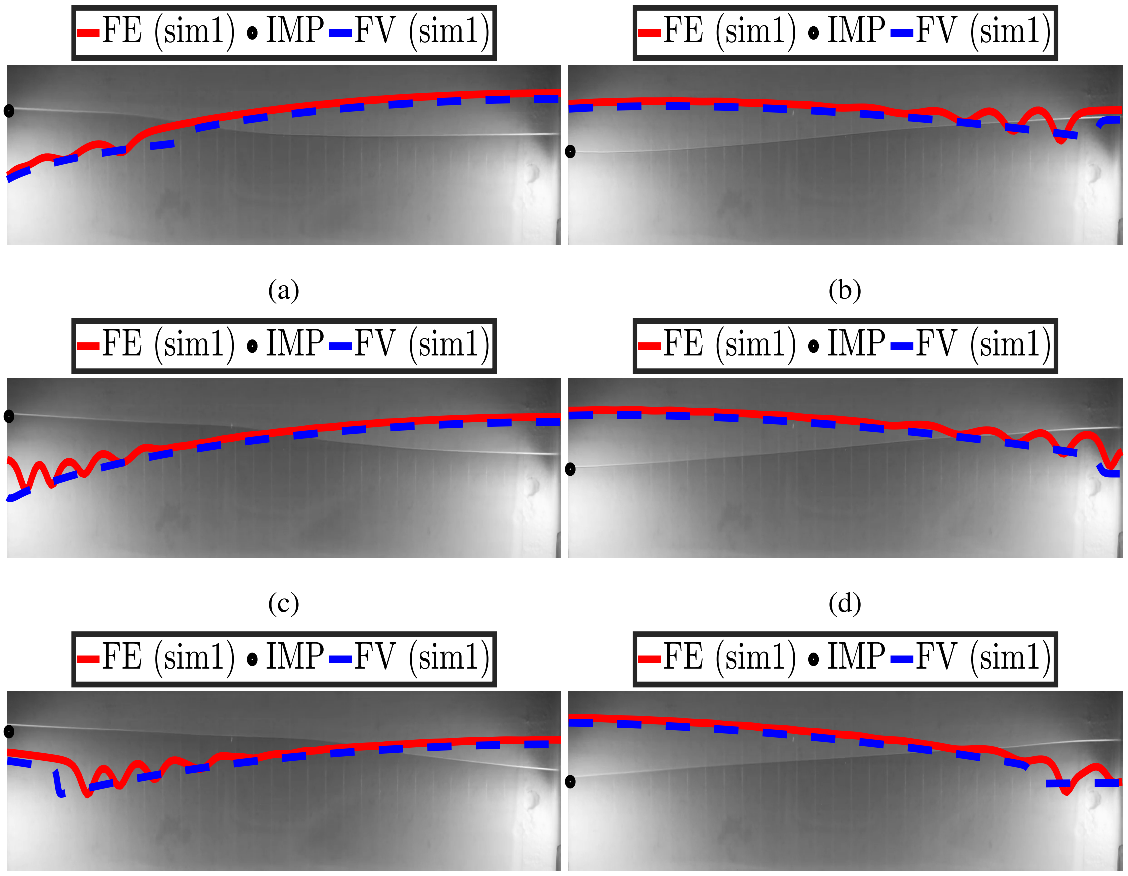


Figure D.18: Case 154: Double smoothed data for $Q(t)$ (green) compared with the fitted function $Q_{\text{fitted}}(t) = 0.0136\cos(2\pi * 0.48t - pi/3)$ (black).

D.3 Finite-volume versus finite-element results

It should be noted that a comparison between the numerical methods for simulation one has only been carried out for case 47. This comparison is important as it was the first data-validated result within this work, included in [42], that indicated the finite-element method's limitation in describing the water dynamics.



(a) (b)

(c) (d)

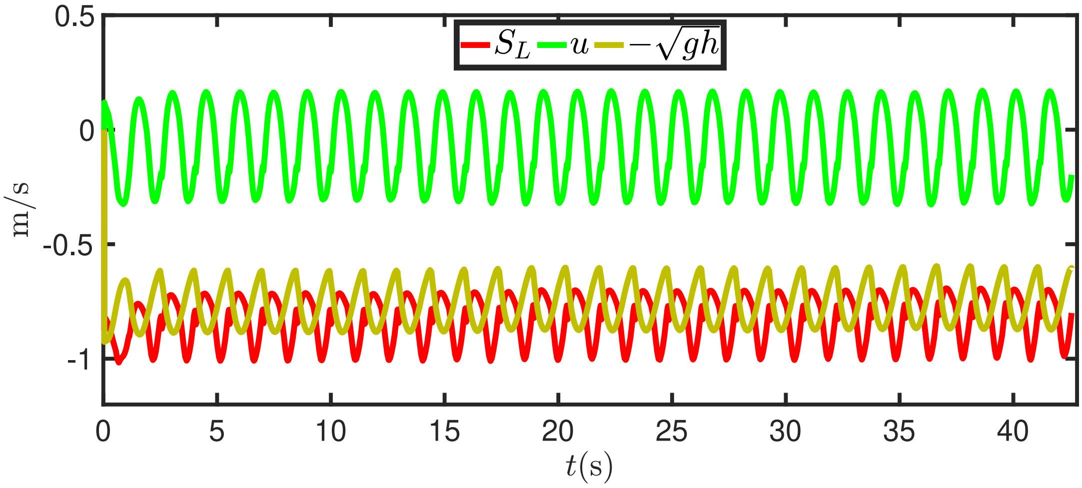
(e) (f)

Figure D.19: Comparison between the results of sim1 ($Q(t)$ imposed) for FV (blue dash) and the FE (red), case 47. Results have been plotted together with the raw images and the measured value $h0_{\text{final}}(t)$ (black dot) at times $t = 3.88\text{s}, 4.88\text{s}, 6.88\text{s}, 9.08\text{s}, 12.68\text{s}, 14.68\text{s}$. Momentum damping is $\gamma = 2.479\ \text{s}^{-1}$. 800 (FV) and 200 (FE) elements have been used.

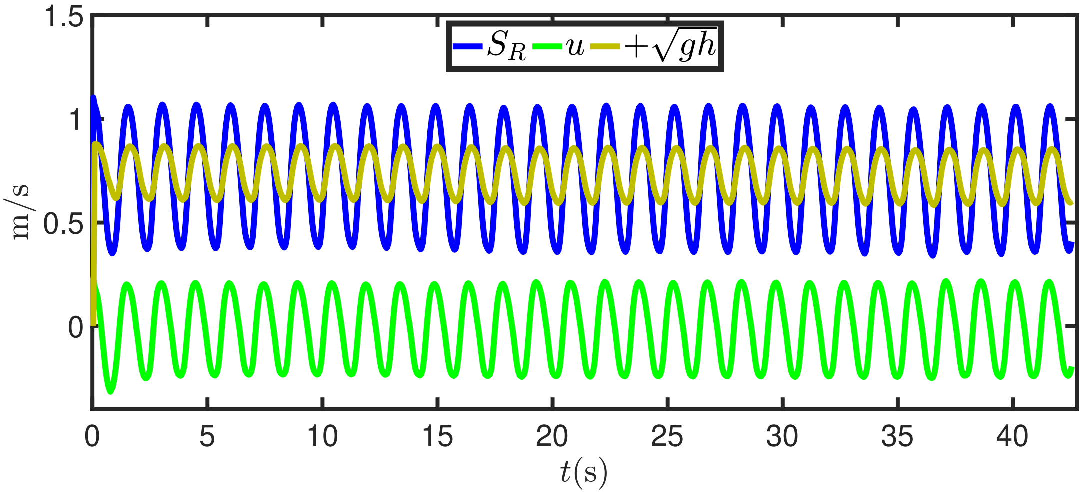
A relatively steep wave front, computed by the FV method for sim1, is observed. Unfortunately, the FE method creates instabilities at those steep wave fronts manifested by the ripples. These ripples are considered to be a consequence of reaching the limit of the irrotational assumption, while the numerical depth at the left wall does not coincide with the measurements ($h(0,t)$).

D.4 Computed wave speeds

In this sub-section, the wave-speeds for cases 149, 165 have been computed.

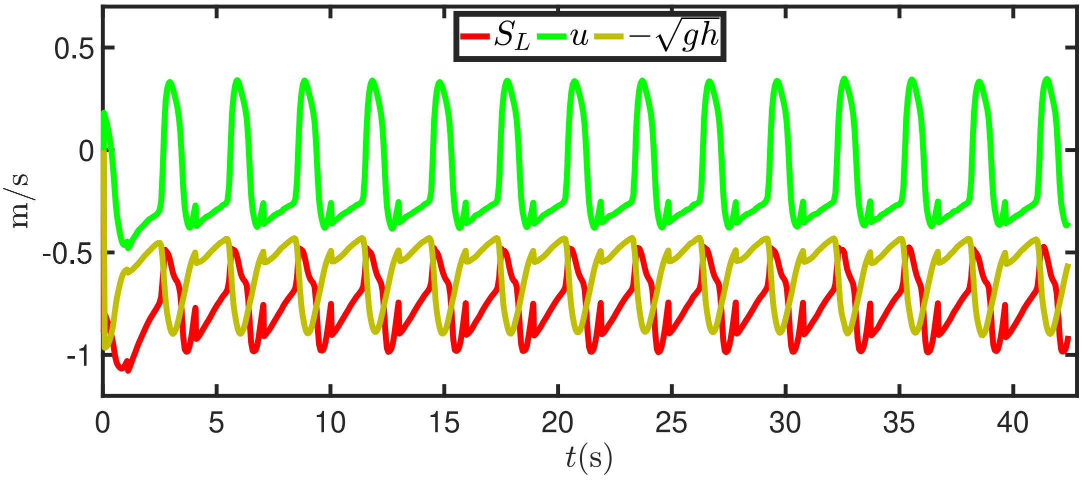


(a) Minimum wave speed $S_L = \min\left(u_L - \sqrt{gh_L}, u_R - \sqrt{gh_R}\right)$ over time.

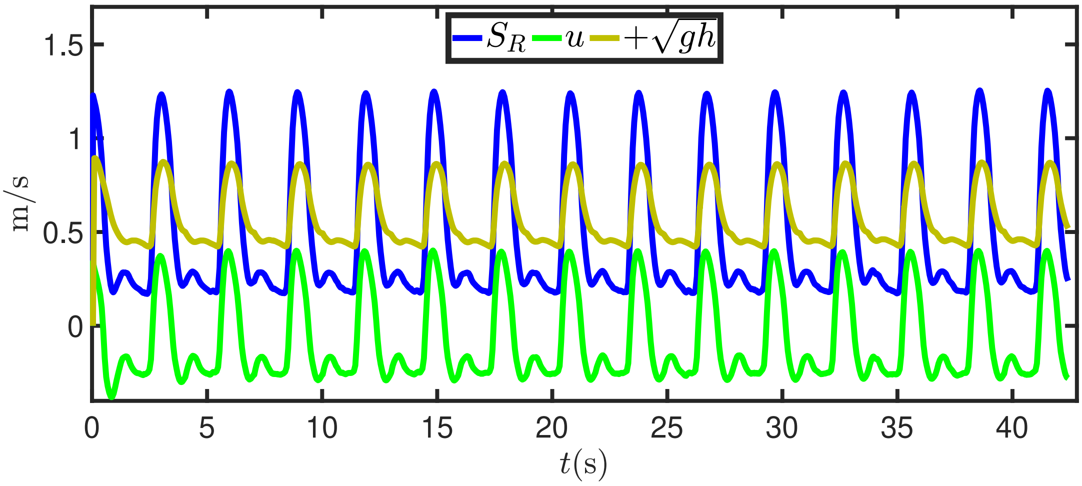


(b) Maximum wave speed $S_R = \max\left(u_L + \sqrt{gh_L}, u_R + \sqrt{gh_R}\right)$ over time.

Figure D.20: Minimum and maximum wave speeds S_L, S_R computed for case 149. It can be seen that throughout the experiment the flow is subsonic ($S_L \leq 0 \leq S_R$) as $\mathrm{Fr} < 1$.



(a) Minimum wave speed $S_L = \min\left(u_L - \sqrt{gh_L}, u_R - \sqrt{gh_R}\right)$ over time.



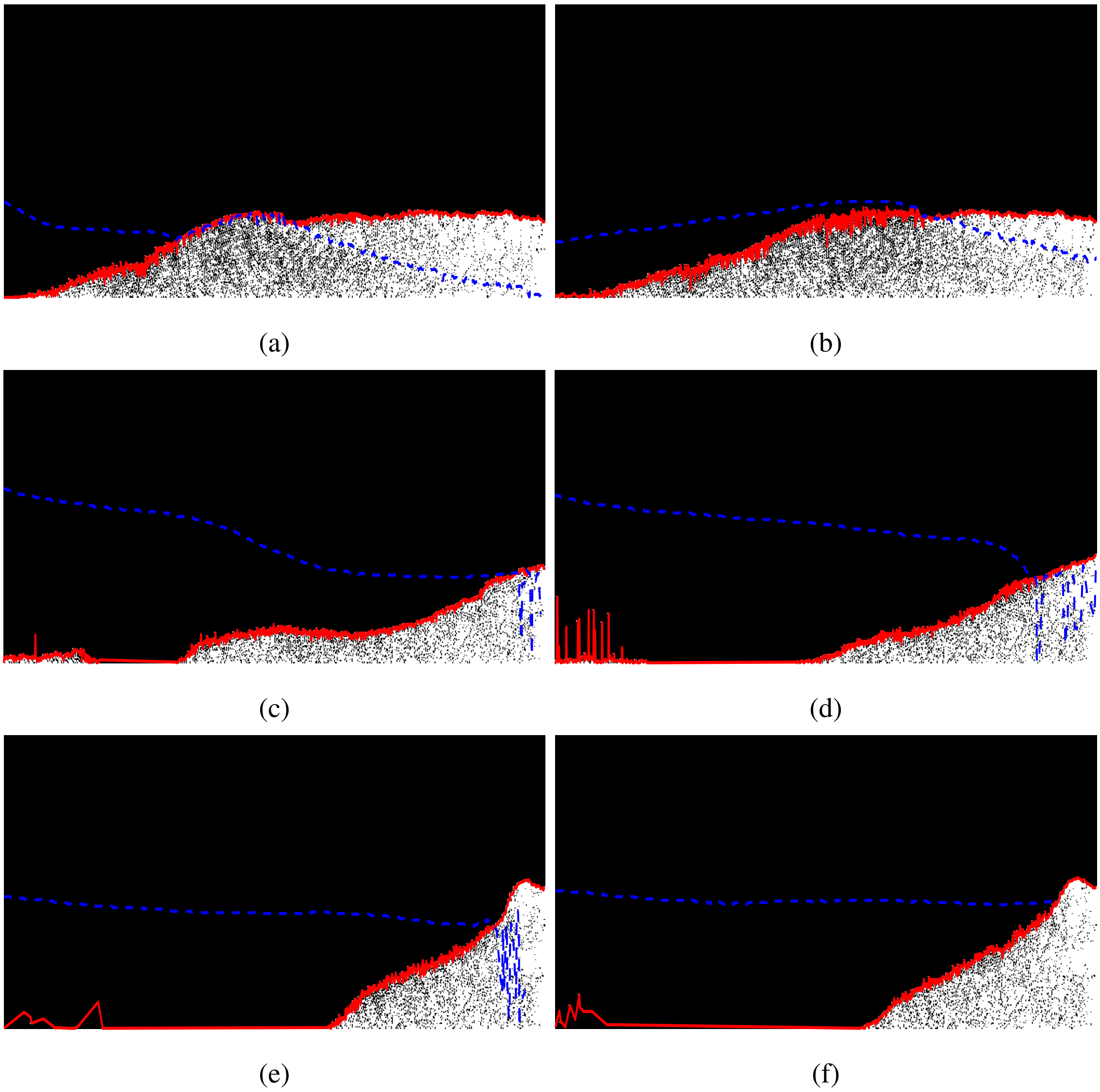
(b) Maximum wave speed $S_R = \max\left(u_L + \sqrt{gh_L}, u_R + \sqrt{gh_R}\right)$ over time.

Figure D.21: Minimum and maximum wave speeds S_L, S_R computed for case 165. It can be seen that throughout the experiment the flow is subsonic ($S_L \leq 0 \leq S_R$) as $\mathrm{Fr} < 1$.

E Water-bed dynamics

E.1 Tracked bed & water binary snapshots

In the figures that follow, the binary images of the tracked water and bed have been illustrated. This "version" of the images was the one that was ultimately tracked from the algorithm.

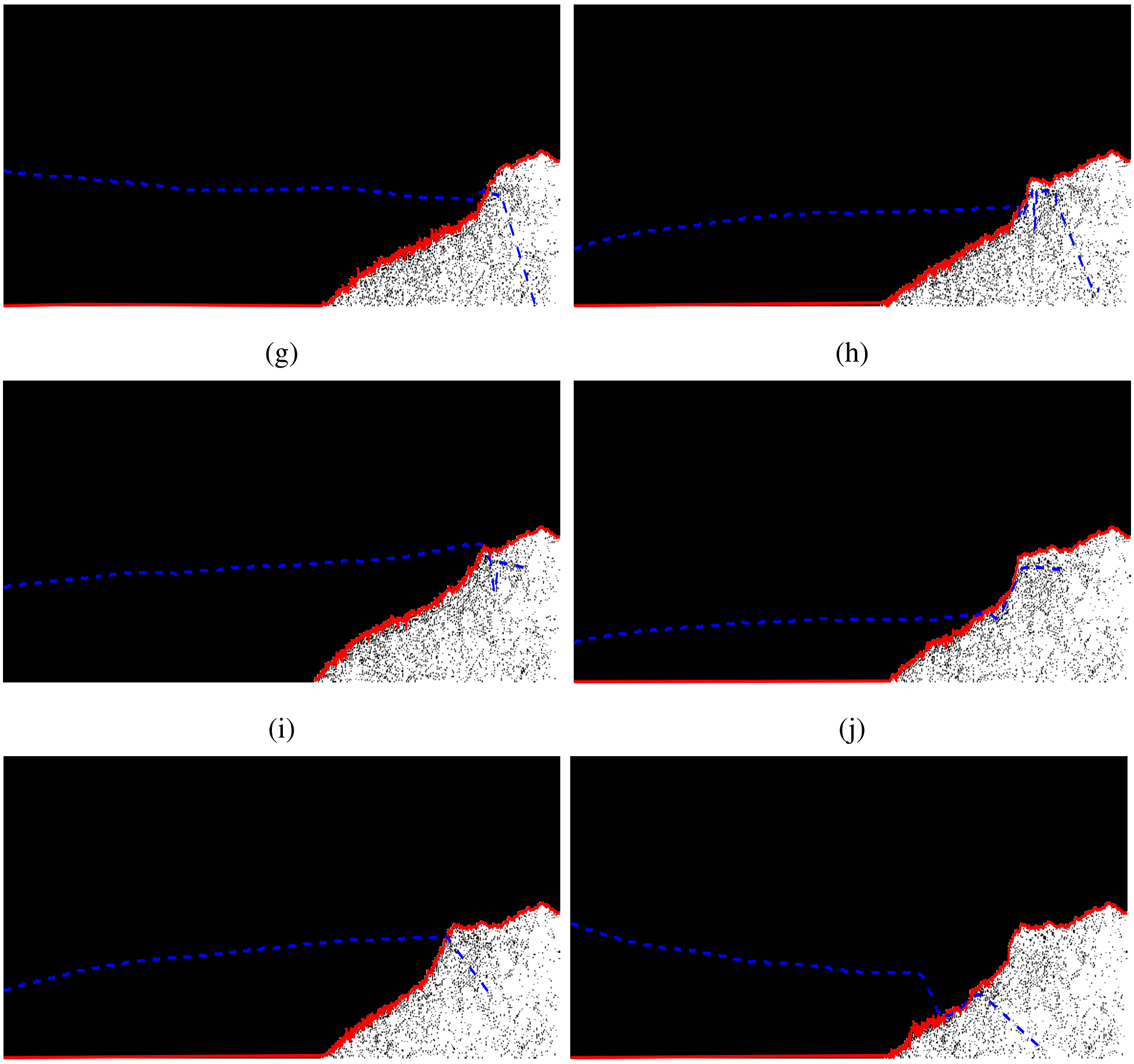


(a) (b)

(c) (d)

(e) (f)

Figure E.22: Tracked binary snapshots for case 219-223, at $t =$ 0min 30s, 43s, 5min, 7min 41s, 11min 20s, 18min 20s.


(g)

(h)

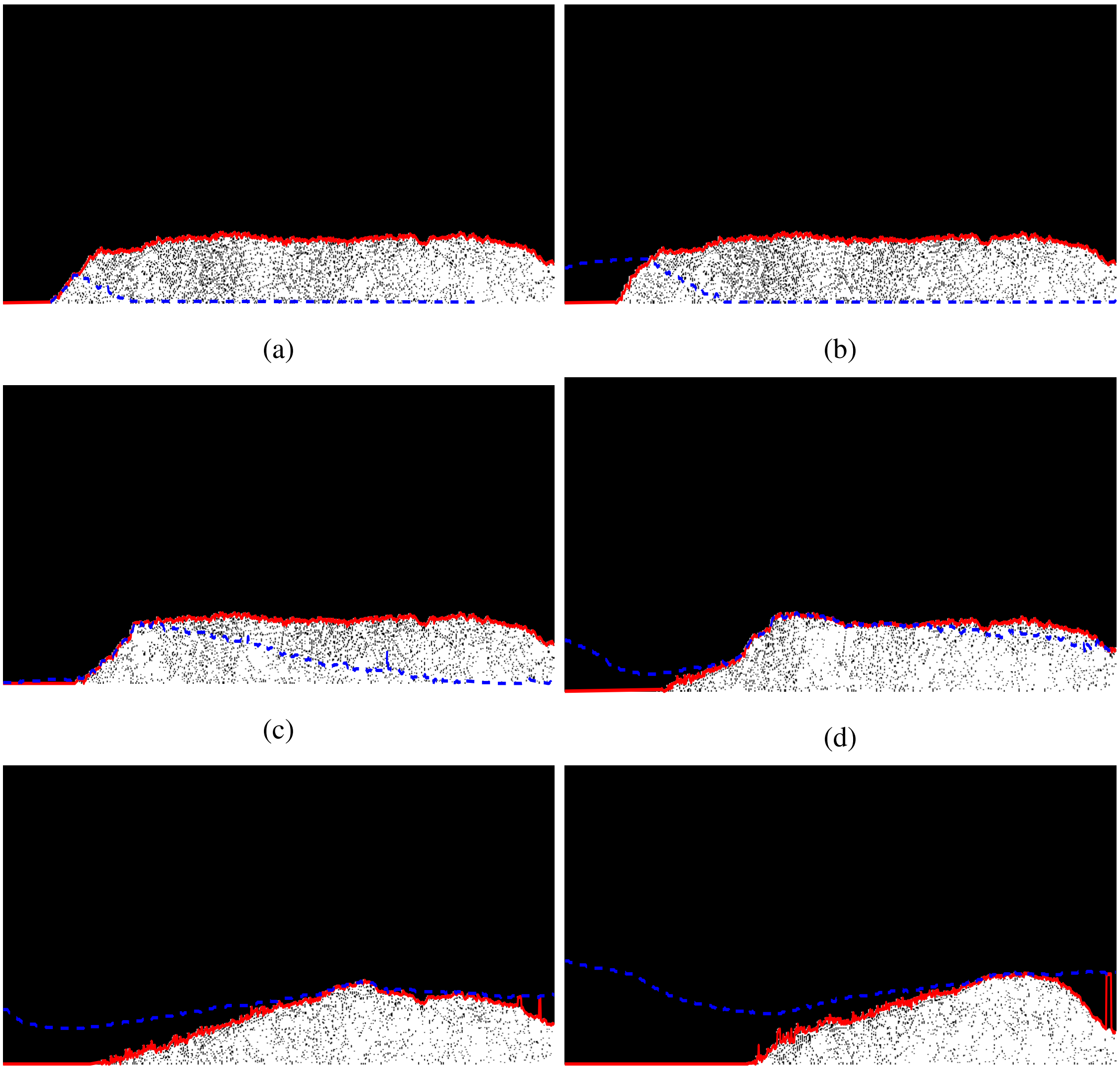
(i)

(j)

(k)

(l)

Figure E.22: Tracked binary snapshots for case 219-223, at $t =$ 22min 12s, 30min, 35min 20s, 50min 30s, 68min 44s, 69min 48s.

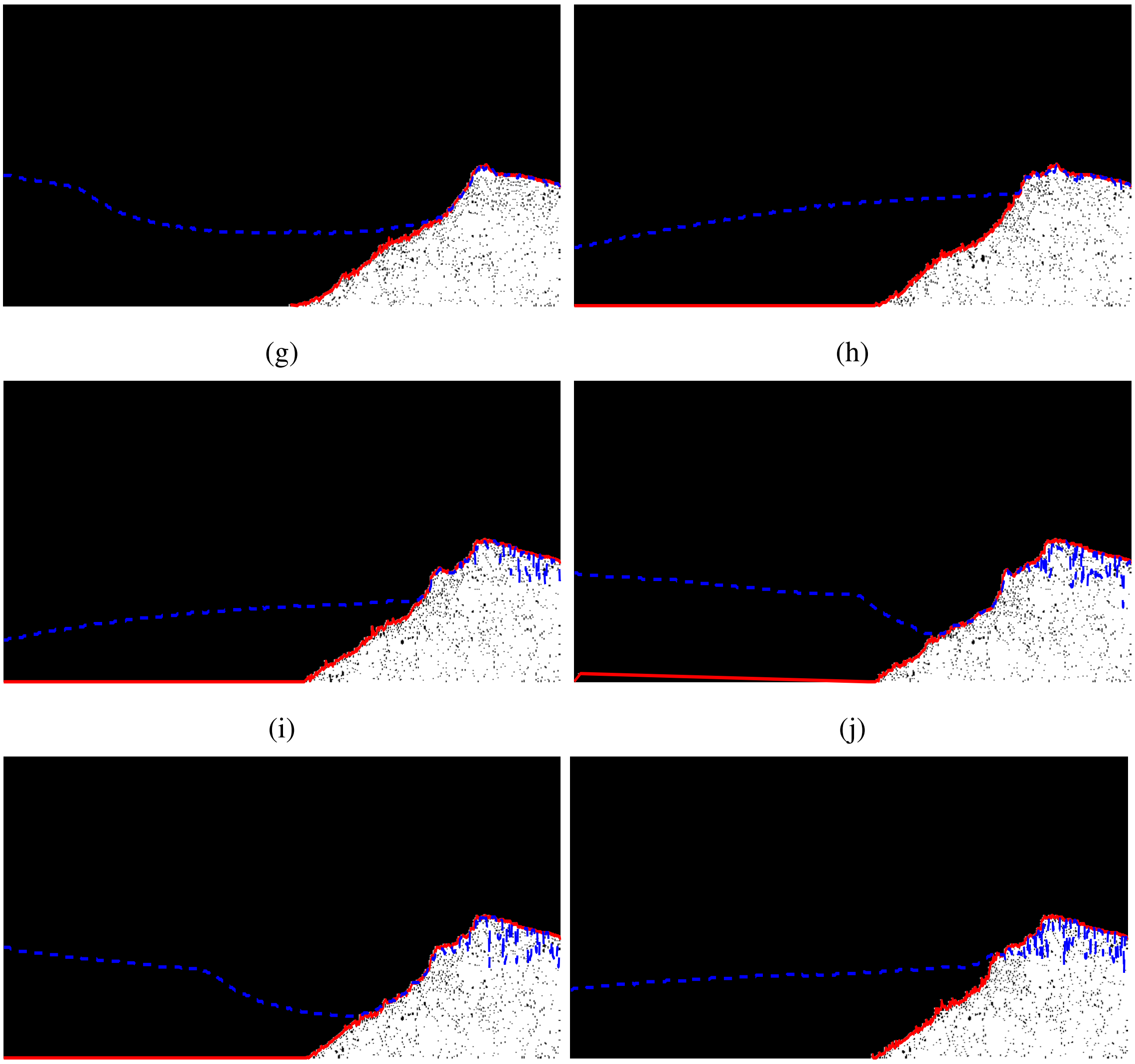


(a) (b)

(c) (d)

(e) (f)

Figure E.23: Tracked binary snapshots for case 232-237, at t = 17s, 19s, 45s, 1min 43s, 2min 45s, 7min 25s.

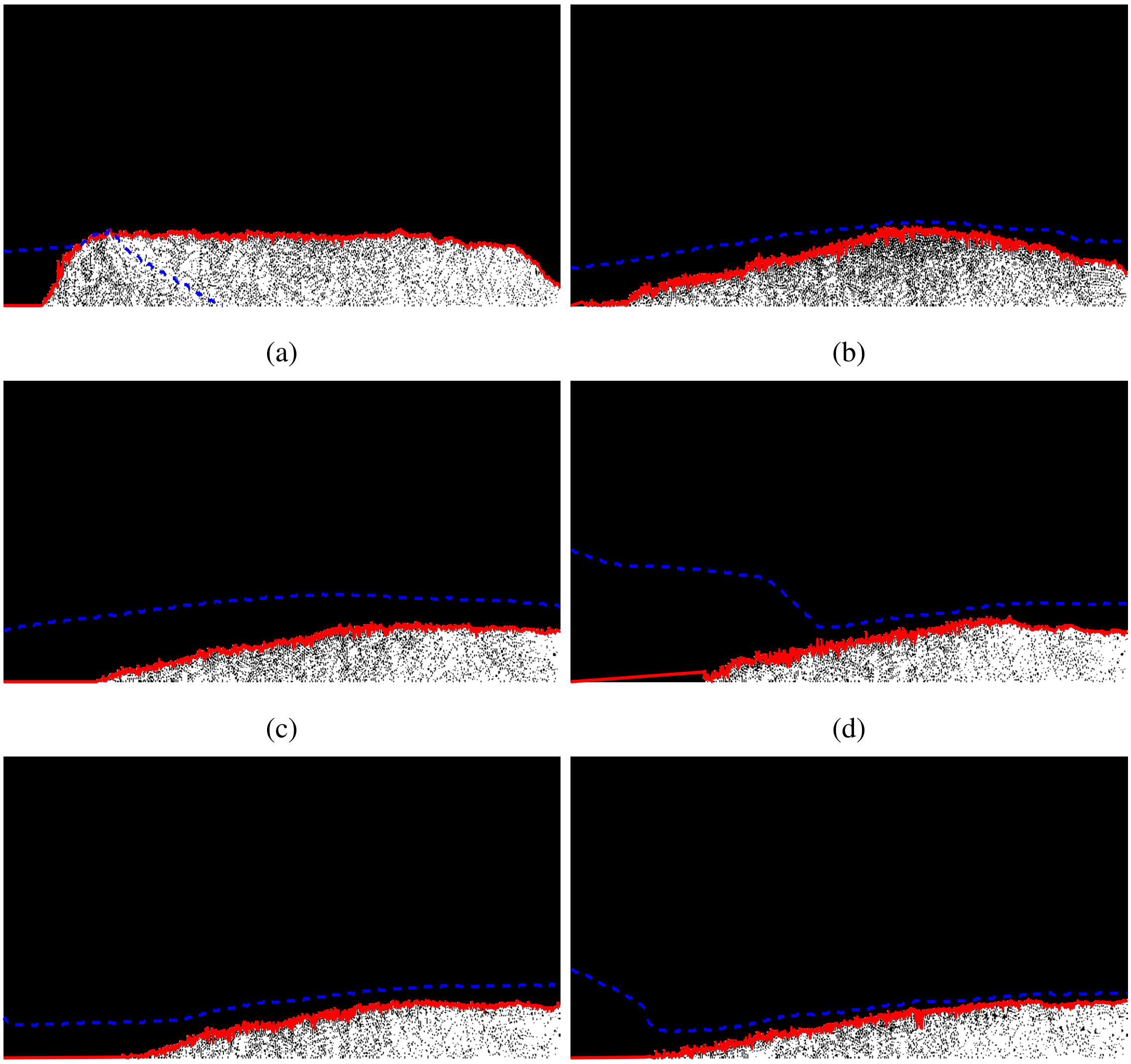


(g) (h)

(i) (j)

(k) (l)

Figure E.23: Tracked binary snapshots for case 232-237, at t = 23min 24s, 50min 6s, 70min 40s, 83min 24s, 101min 40s, 103min 5s.

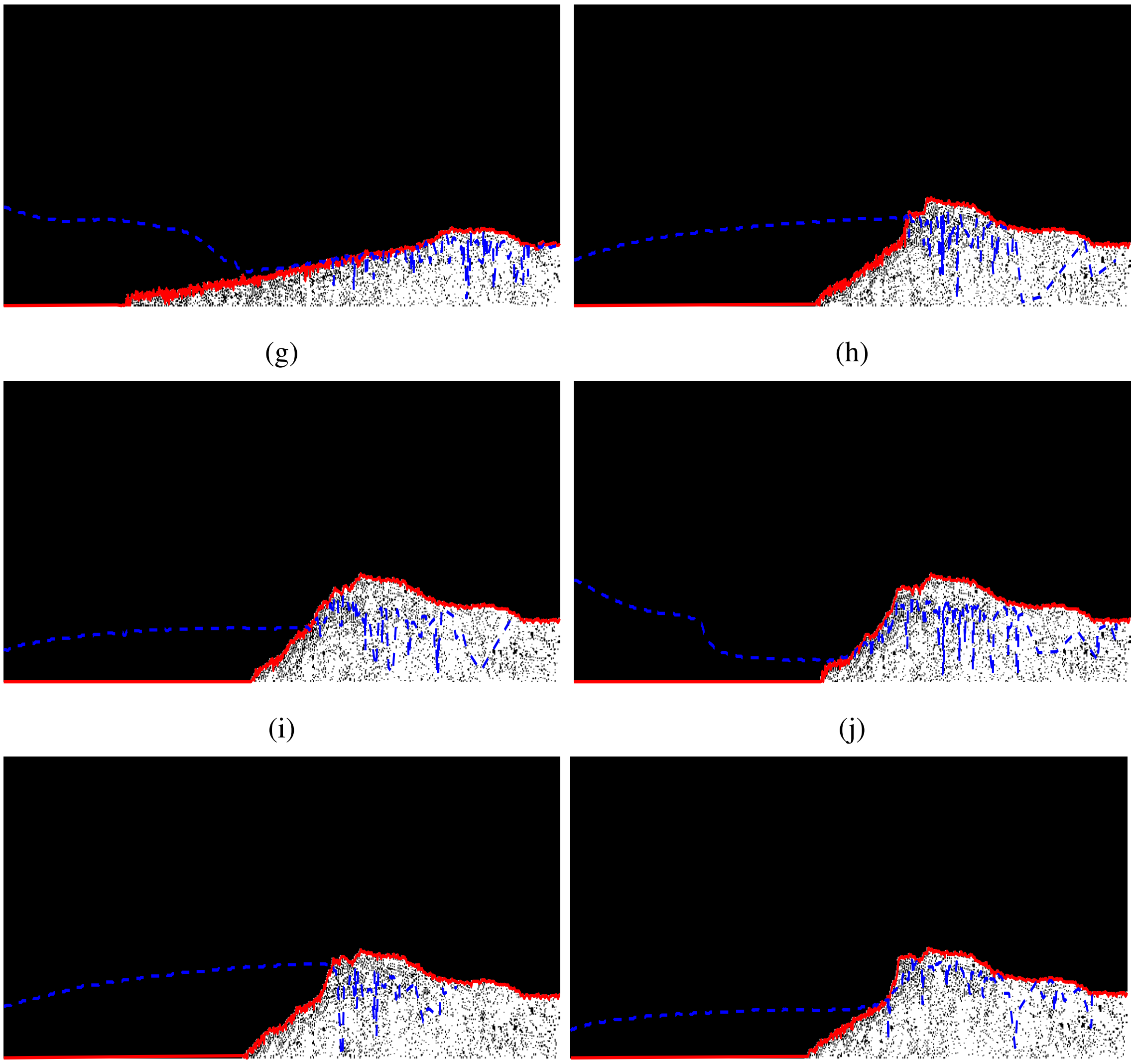


(a) (b)

(c) (d)

(e) (f)

Figure E.24: Tracked binary snapshots for case 248-256 248-256, at $t =$ 16s, 50s, 10min 20s, 20min, 23min 27s, 30min 4s.



(g) (h)

(i) (j)

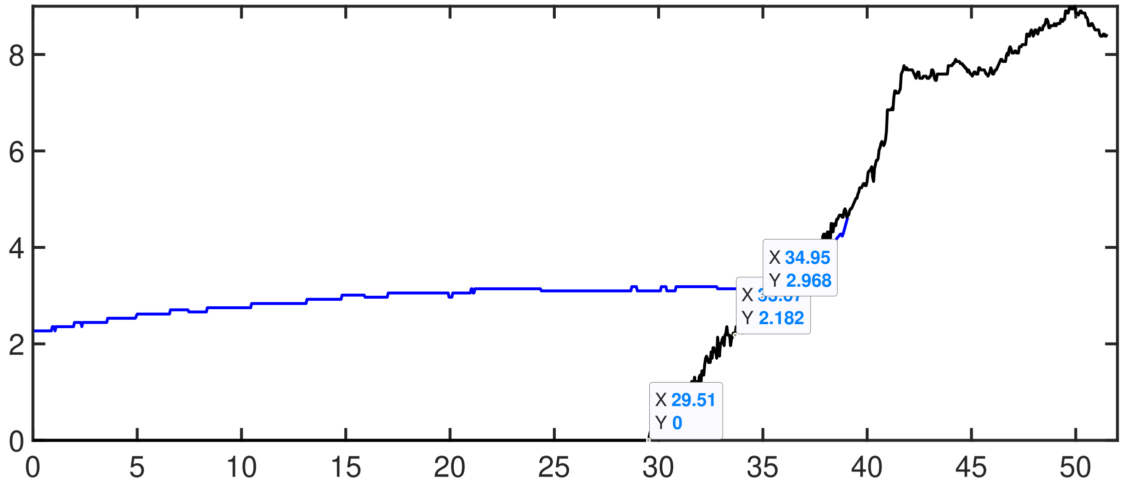
(k) (l)

Figure E.24: Tracked binary snapshots for case 248-256, at $t = 40\mathrm{min}, 50\mathrm{min}\,40\mathrm{s}, 60\mathrm{min}, 62\mathrm{min}\,28\mathrm{s}, 81\mathrm{min}\,20\mathrm{s}, 82\mathrm{min}\,34\mathrm{s}$.

E.2 Final-time bed angle calculations

In this section, the final-bed angles are computed separately for the wet and swash region for each case. Runs of the image analysis algorithm, at the times of interest, provided measurements for the length and the height of the bed in each region. Then using these measurements the slope of the bed was calculated using equation (5.3). Lastly, taking the inverse tangent of the slope and expressing the angles in degrees led to the following results.

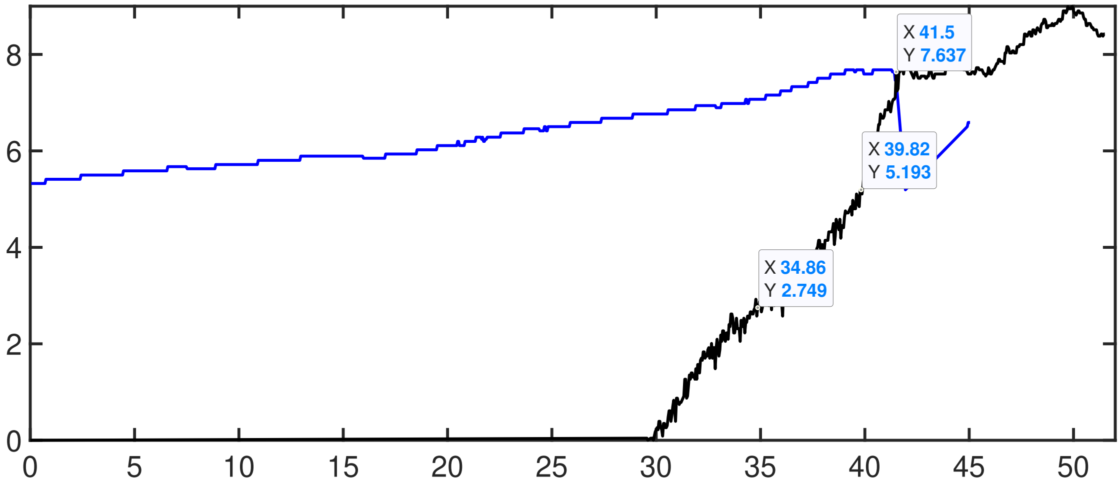
(a) Minimum water depth at $t = 70 : 01\text{min}$.



(b) Slope angle 29 °.

Figure E.25: Data originating from the wet region for videos 219-223. Bed angle computed by the points (29.51,0), (33.67, 2.182), (34.95, 2.968). Units are in centimetres.

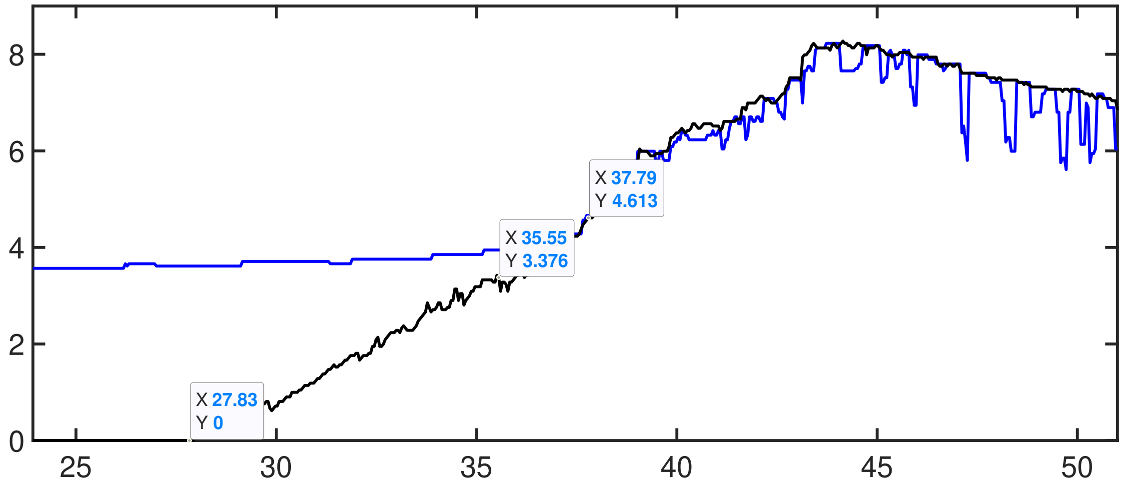
(c) Maximum water depth at $t = 70 : 02$min.



(d) Slope angle 40°.

Figure E.25: Data originating from the swash region for videos 219-223. Bed angle computed by the points (34.86, 2.749), (39.82, 5.193), (41.5, 7.637). Units are in centimetres.

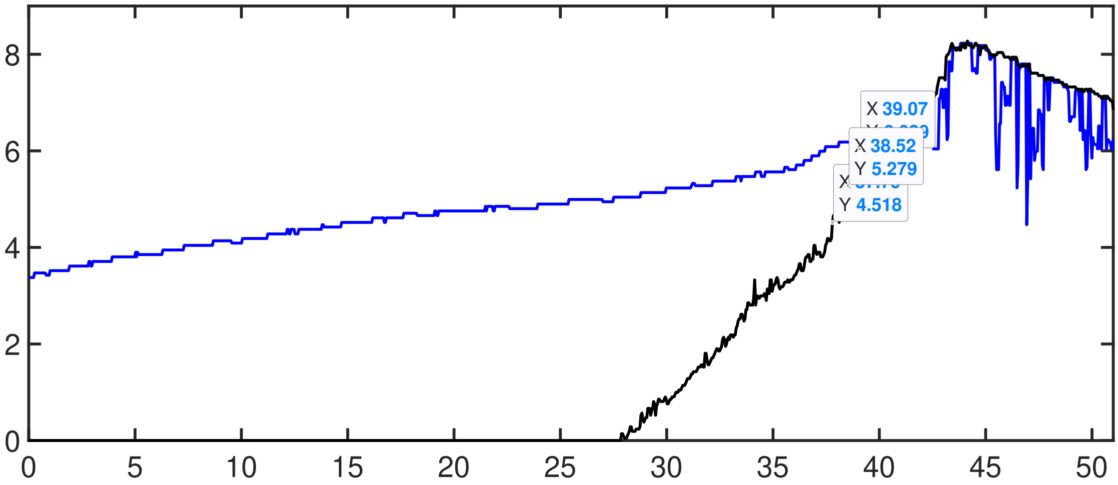
(a) Minimum water depth at $103:04$min.



(b) Slope angle 26°.

Figure E.26: Data originating from the wet zone for videos 232-237. Bed angle computed by the points (27.83, 0), (35.55, 3.376), (37.79, 4.613). Units are in centimetres.

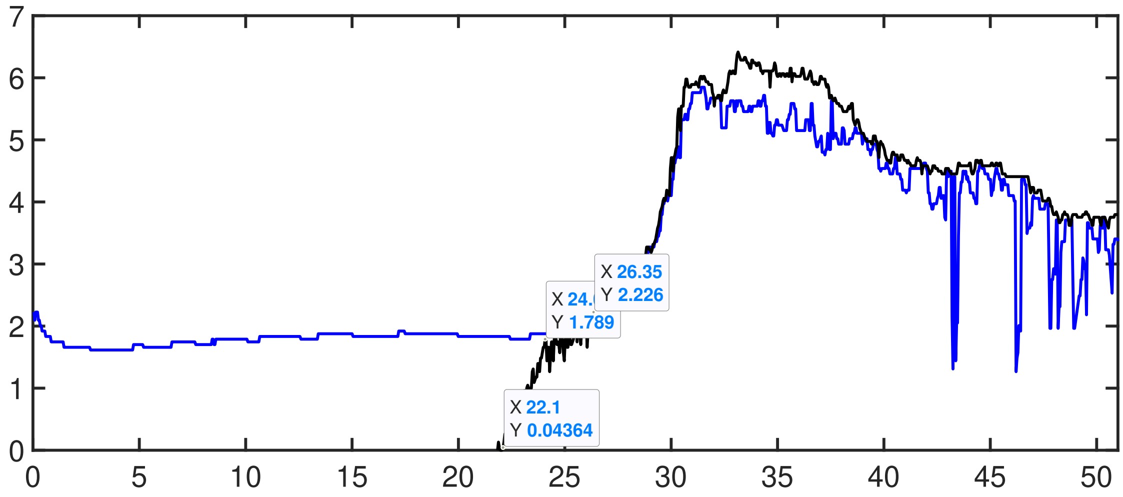
(c) Maximum water depth at $103:05$min.



(d) Slope angle 50°.

Figure E.26: Data originating from the swash zone for videos 232-237. Bed angle computed by the points (37.79, 4.518), (38.52, 5.279), (39.07, 6.039). Units are in centimetres.

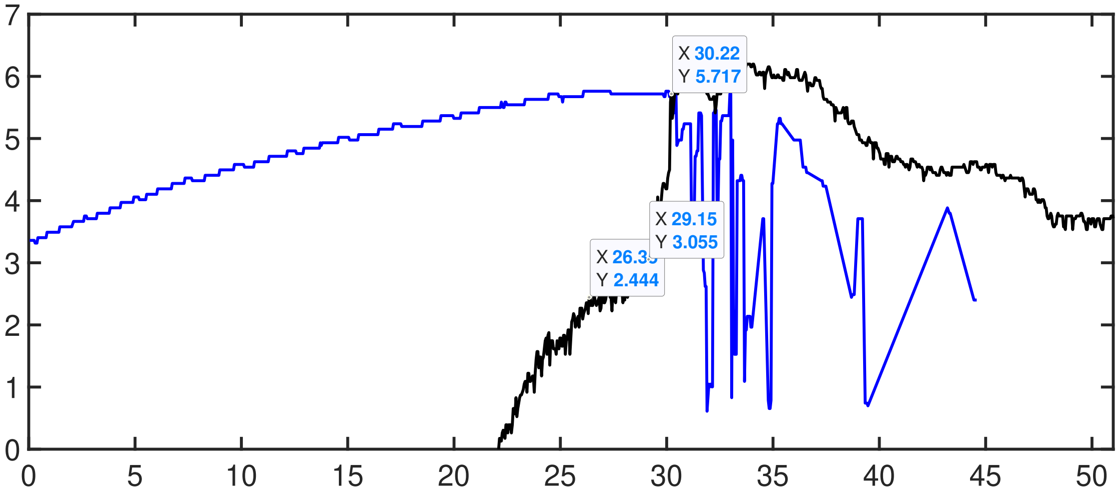
(a) Minimum water depth at 82 : 31min.



(b) Slope angle 26°.

Figure E.27: Data originating from the wet zone for videos 248-256. Bed angle computed by the points (22.1, 0), (24.07, 1.789), (26.35, 2.226). Units are in centimetres.

(c) Maximum water depth at 82 : 33min.

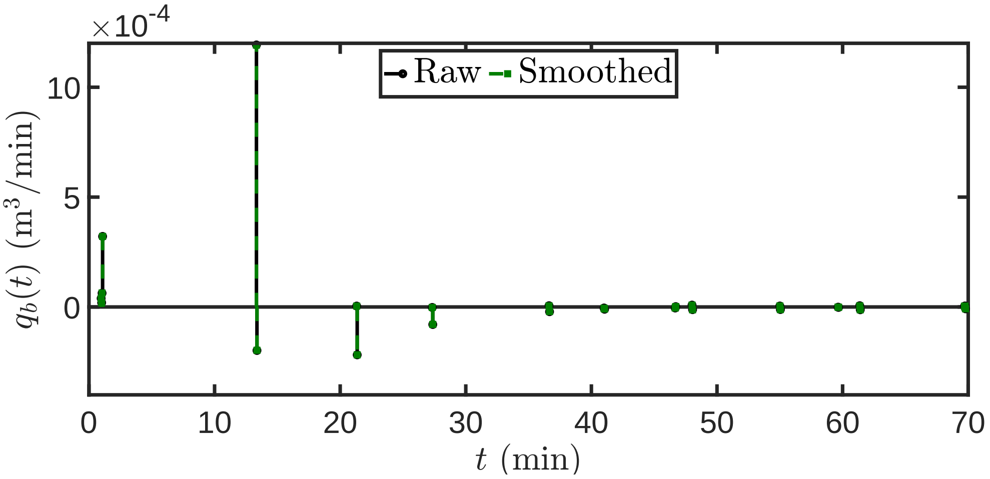


(d) Slope angle 40 °.

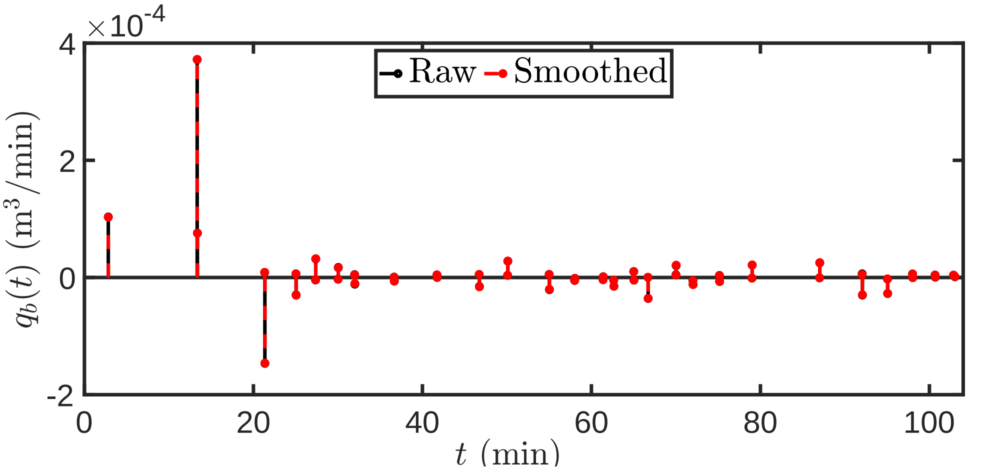
Figure E.27: Data originating from the swash zone for videos 248-256. Bed angle computed by the points (26.35, 2.444), (29.15, 3.055), (30.22, 5.717). Units are in centimetres.

E.3 Bed flux

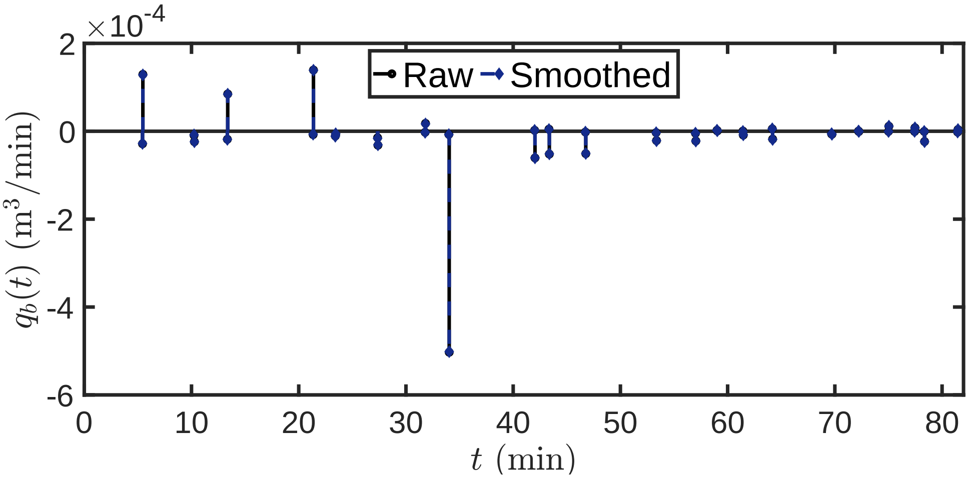
In contrast to Section 5.2.1.4, a moving average smoothing method was applied to the bed height data prior to the flux computation. In this way, potential improvements to the bed flux computation were investigated. Observing the following figures, it can be concluded that smoothing the bed data would only result in changes in bed flux in the order of O(-8)-O(-6) thus rendering the smoothing step unimportant.



(a) Case 219-223.



(b) Case 232-237.



(c) Case 248-256.

Figure E.28: Comparison of the raw and smoothed bed flux for cases 219-223 (top panel), 232-237 (middle panel) and 248-256 (bottom panel). The error between the raw and the smoothed bed flux has been visualised in Figure E.29.

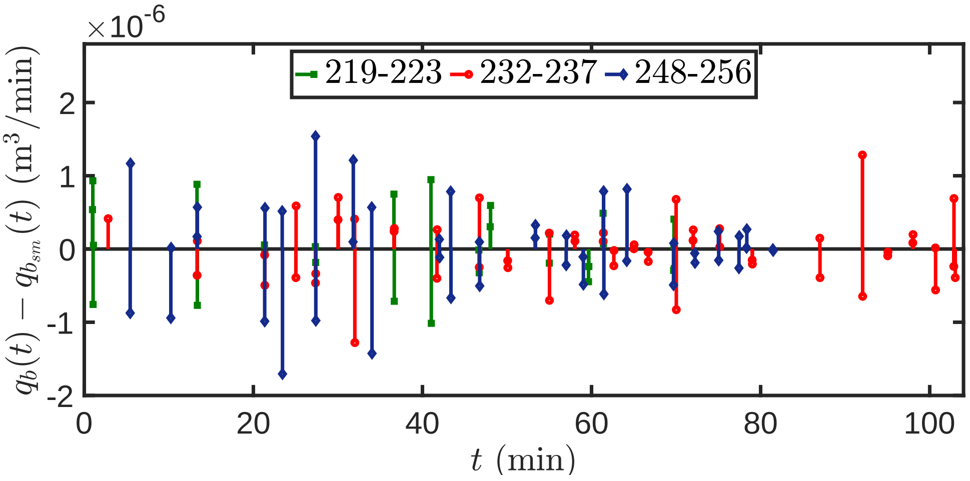
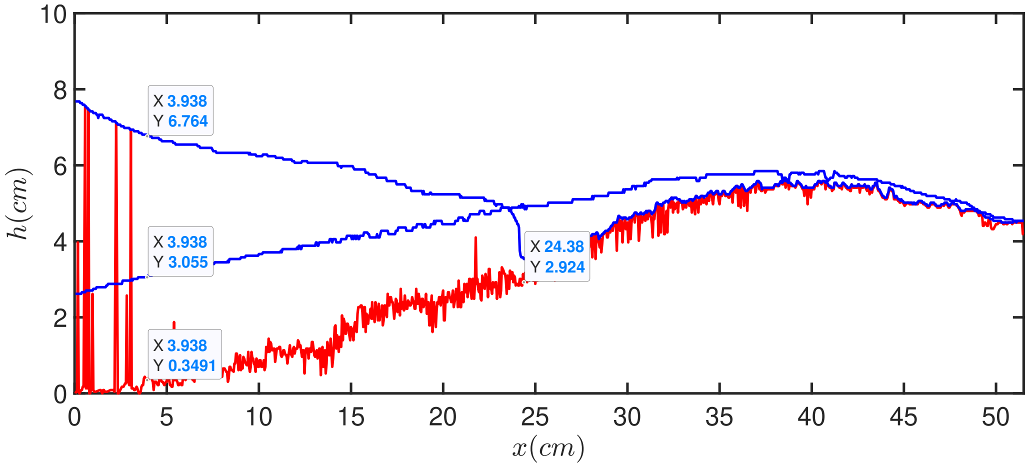


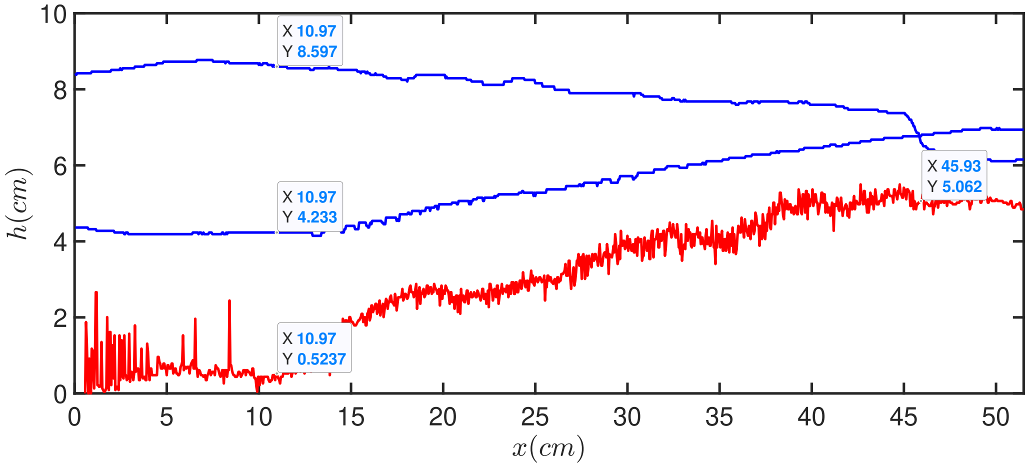
Figure E.29: Bed flux error over individual wave-cycles, computed as the difference between the raw and the smoothed bed flux for case 219 (green squares), 232 (red circles) and 248 (blue diamonds).

E.4 Iribarren number computations

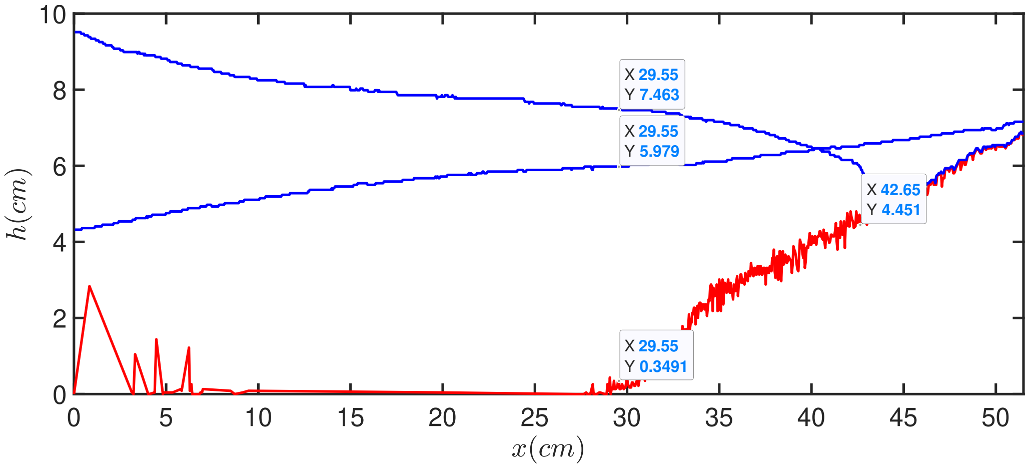
In this section, the computation of the Iribarren number is performed. Essentially, the wave height (H_b) at the toe of the bed and the slope of the bed ($\tan\alpha$) are computed with the help of the data points indicated on the blue and red lines, respectively. Then the wavelength (λ_b) is computed knowing the wave period for each experiment and acceleration due to gravity equal to $9.81\mathrm{m/s}^2$.



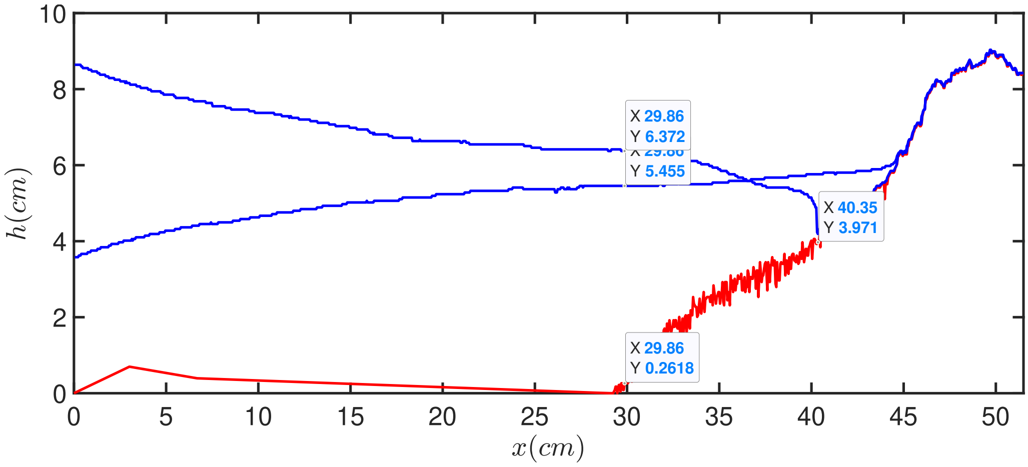
(a) $t = 1\mathrm{min}\,2\mathrm{min}$, $I_b = 2.04$, suggests a plunging breaker at the limit of becoming a collapsing breaker. Here: H_b=0.0371m, $\tan\alpha = 0.126$, $\lambda_b = 9.7582\mathrm{m}$, $g = 9.81\mathrm{m/s}^2$.



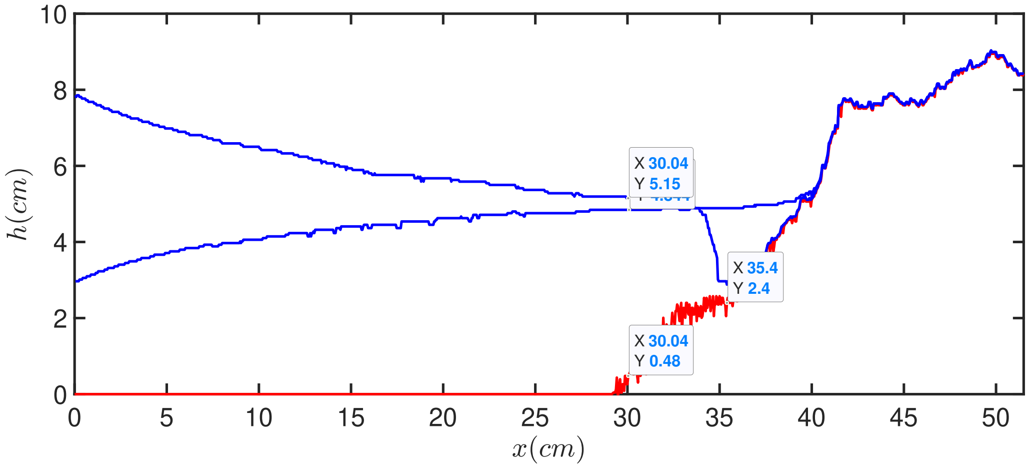
(b) $t = 1\mathrm{min}\,38\mathrm{s}$, $I_b = 1.94$, suggests a plunging breaker. Here: H_b=0.0436m, $\tan\alpha = 0.1298$, $\lambda_b = 9.7582\mathrm{m}$, $g = 9.81\mathrm{m/s}^2$.



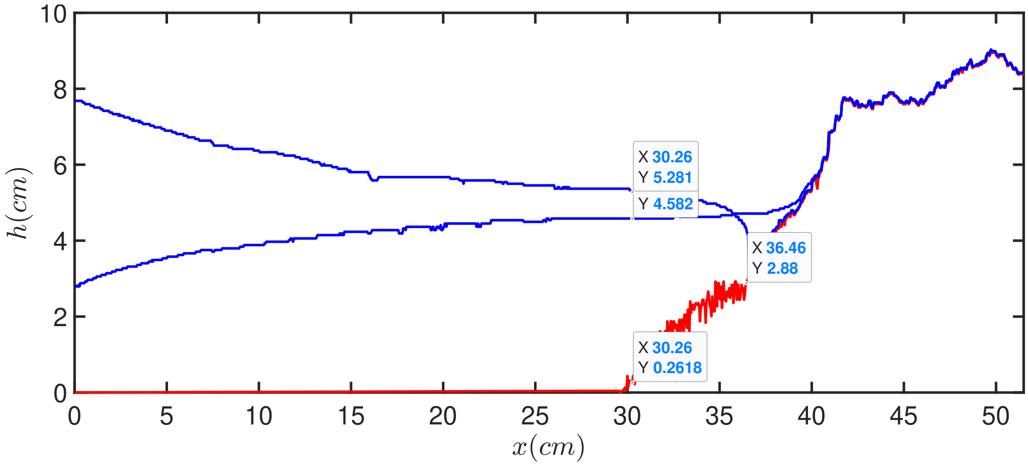
(c) $t = 9\text{min}\,8\text{s}$, $I_b = 8.03$ suggests a surging breaker. Here: H_b=0.0148m, $\tan\alpha = 0.3131$, $\lambda_b = 9.7582\text{m}$, $g = 9.81\text{m/s}^2$.



(d) $t = 21\text{min}\,45\text{s}$, $I_b = 11.53$, suggests a surging breaker. Here: H_b=0.0092m, $\tan\alpha = 0.3536$, $\lambda_b = 9.7582\text{m}$, $g = 9.81\text{m/s}^2$.

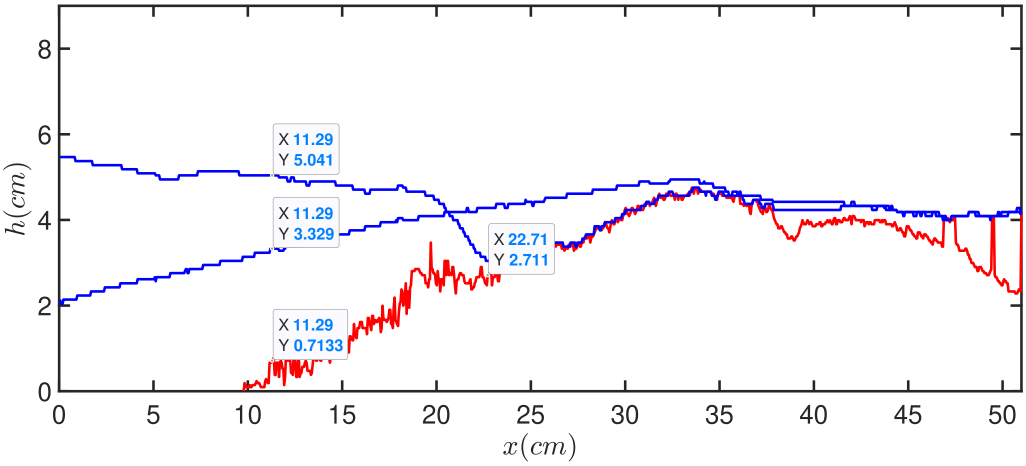


(e) $t = 68\text{min}\,6\text{s}$, $I_b = 20.23$, suggests a surging breaker. Here: H_b=0.0031m, $\tan\alpha = 0.3582$, $\lambda_b = 9.7582\text{m}$, $g = 9.81\text{m}/\text{s}^2$.

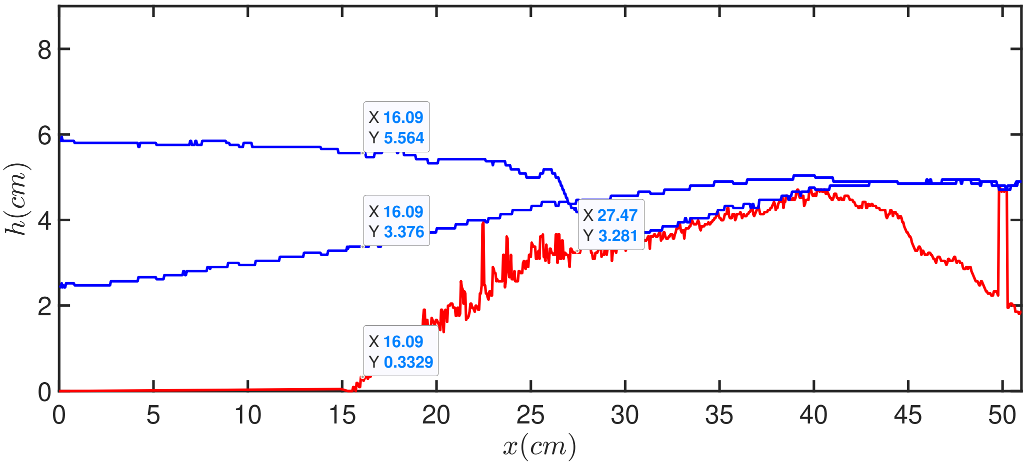


(f) $t = 70\text{min}\,2\text{s}$, $I_b = 15.78$, suggests a surging breaker. Here: H_b=0.007m, $\tan\alpha = 0.4223$, $\lambda_b = 9.7582\text{m}$, $g = 9.81\text{m}/\text{s}^2$.

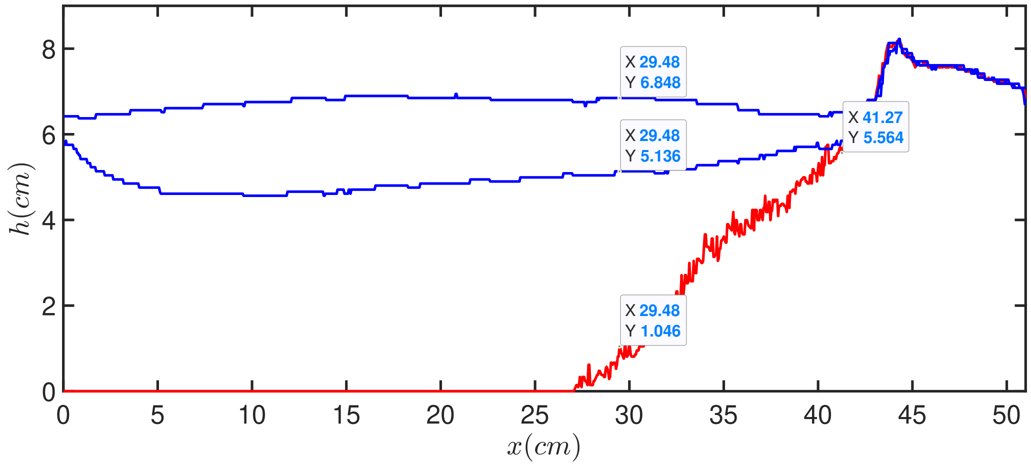
Figure E.30: Iribarren calculation for each of the breakers observed in Fig. 5.24. The Iribarren number is calculated from the provided wave height at the toe of the beach, the beach slope and the wave period is $T_{219-223} = 2.5\text{s}$.



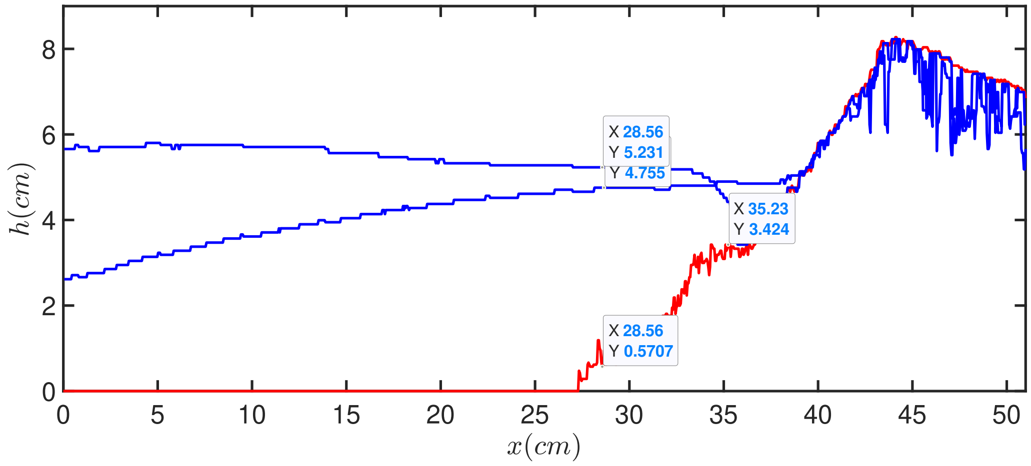
(a) $t = 2\text{min}\,54\text{s}$, $I_b = 2.5$, suggests a collapsing breaker. Here: H_b=0.0171m, $\tan\alpha = 0.1749$, $\lambda_b = 3.5129\text{m}$, $g = 9.81\text{m/s}^2$.



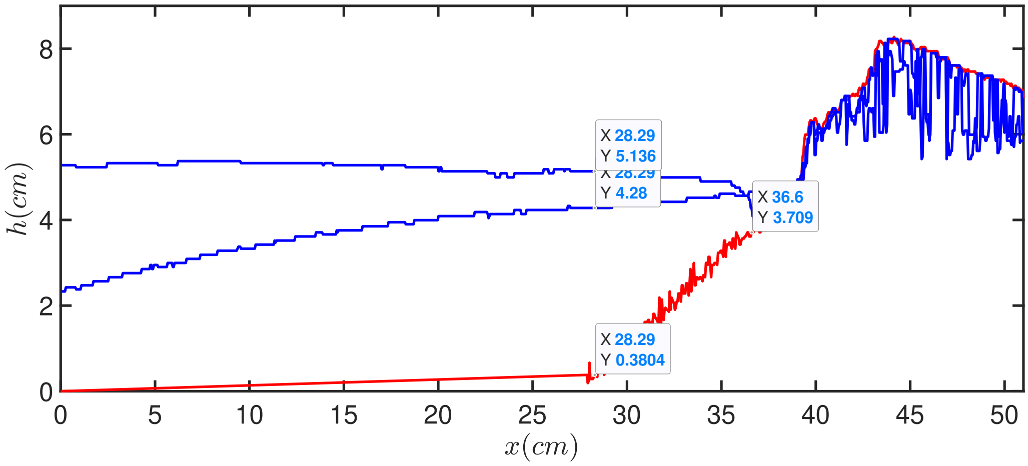
(b) $t = 4\text{min}\,55\text{s}$, $I_b = 3.28$, suggests a collapsing breaker at the limit the limit of becoming a surging breaker. Here: H_b=0.0219m, $\tan\alpha = 0.2591$, $(\lambda_b = 3.5129\text{m}$, $g = 9.81\text{m/s}^2$.



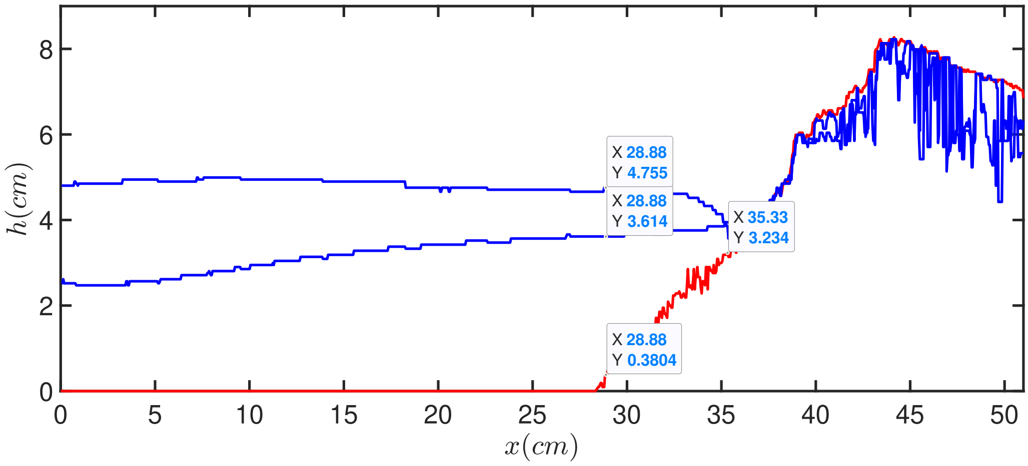
(c) t = 42min 30s, $I_b = 5.49$, suggests a surging breaker. Here: H_b=0.0171m, $\tan\alpha = 0.3832$, $\lambda_b = 3.5129$m, $g = 9.81\text{m/s}^2$.



(d) t = 62min 35s, $I_b = 11.62$, suggests a surging breaker. Here: H_b=0.0048m, $\tan\alpha = 0.4278$, $\lambda_b = 3.5129$m, $g = 9.81\text{m/s}^2$.

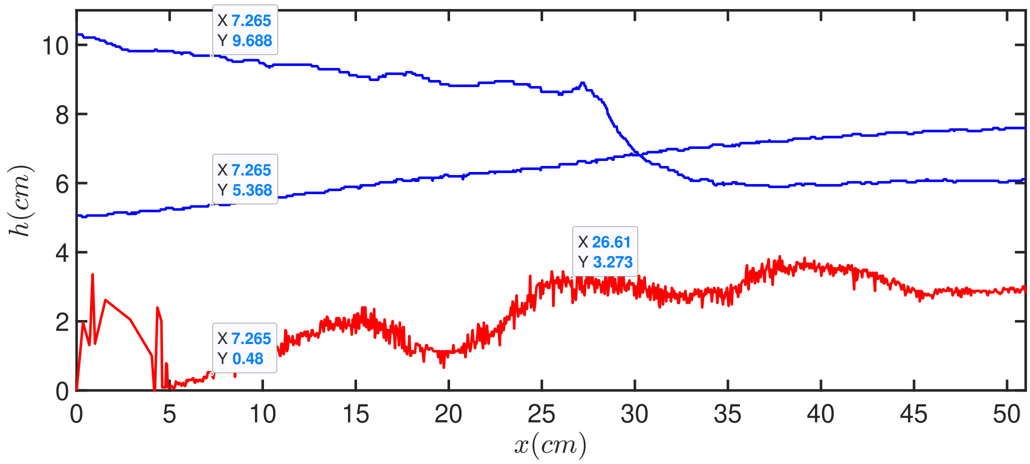


(e) $t = 82\text{min}\,26\text{s}$ $I_b = 8.11$, suggests a surging breaker. Here: H_b=0.0086m, $\tan\alpha = 0.4006$, $\lambda_b = 3.5129\text{m}$, $g = 9.81\text{m/s}^2$.

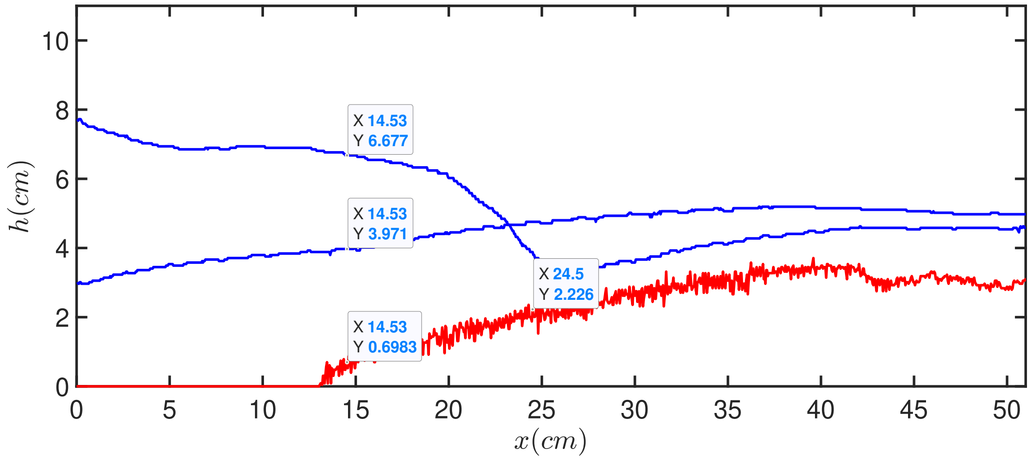


(f) $t = 102\text{min}\,46\text{s}$, $I_b = 7.76$, suggests a surging breaker. Here: H_b=0.0114m, $\tan\alpha = 0.4424$, $\lambda_b = 3.5129\text{m}$, $g = 9.81\text{m/s}^2$.

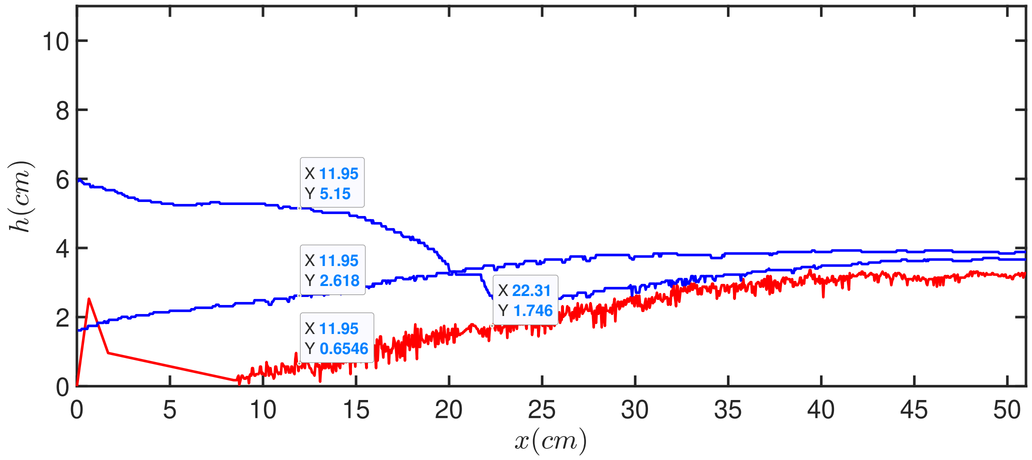
Figure E.31: Iribarren calculation for each of the breakers observed in Fig. 5.26. Wave period is $T_{232-237} = 1.5\text{s}$.



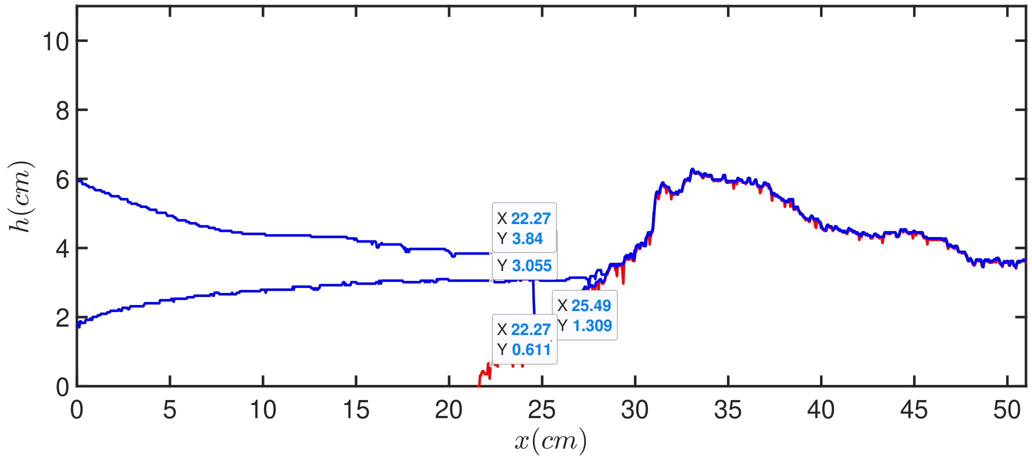
(a) t = 3min 30s, $I_b = 1.74$, suggests a plunging breaker. Here: H_b=0.0432m, $\tan\alpha = 0.1444$, $\lambda_b = 6.2452$m, $g = 9.81\text{m}/\text{s}^2$.



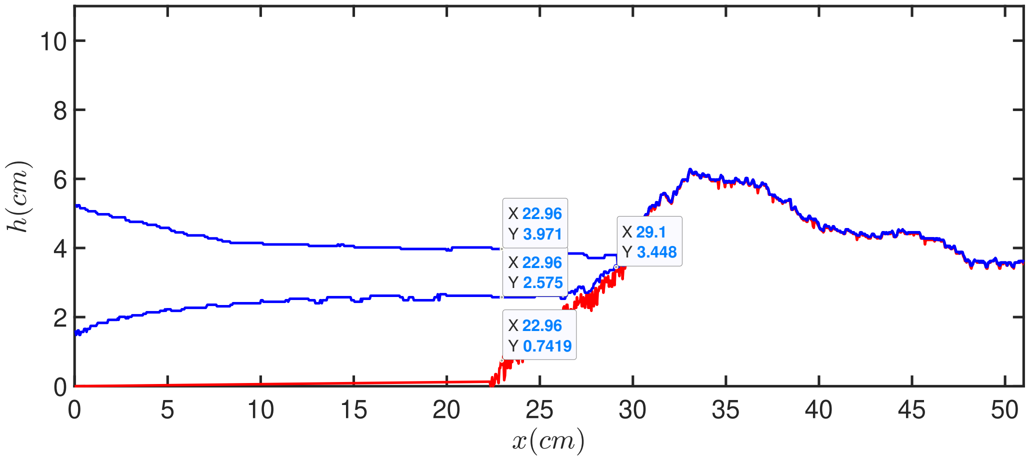
(b) t = 20min 56s, $I_b = 2.33$, suggests a breaker which just transitioned from plunging to collapsing. Here: H_b=0.0271m, $\tan\alpha = 0.1532$, $\lambda_b = 6.2452$m, $g = 9.81\text{m}/\text{s}^2$.



(c) $t = 31\text{min}\,8\text{s}$, $I_b = 1.65$, suggests a plunging breaker. Here: H_b=0.0253m, $\tan\alpha = 0.1053$, $\lambda_b = 6.2452\text{m}$, $g = 9.81\text{m/s}^2$.



(d) $t = 61\text{min}\,26\text{s}$, $I_b = 6.11$, suggests a surging breaker. Here: H_b=0.0078m, $\tan\alpha = 0.2168$, $\lambda_b = 6.2452\text{m}$, $g = 9.81\text{m/s}^2$.



(e) $t = 80\text{min}\,25\text{s}$, $I_b = 9.32$, suggests a surging breaker. Here: H_b=0.014m, $\tan\alpha = 0.4407$, $\lambda_b = 6.2452\text{m}$, $g = 9.81\text{m/s}^2$.

Figure E.32: Iribarren calculation for each of the breakers observed in Fig. 5.28. Wave period is $T_{248-256} = 2\text{s}$.

Bibliography

[1] *Living with coastal erosion in Europe: Sediment and Space for Sustainability Part I - Major findings and Policy Recommendations of the EUROSION project*. EUCC. [Online Edition], May 2004. `uuid:483327a3-dcf7-4bd0-a986-21d9c8ec274e`, [Accessed March 2018].

[2] Shingle beach at Hurst Spit, Hampshire. New Forest District Council, 2015.

[3] Audusse, E., Bouchut, F., Bristeau, M. O., Klein, R. and Perthame, B. A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM J. Sci. Comput.*, 25(6):2050–2065, 2004. doi: https://doi.org/10.1137/S1064827503431090.

[4] Beakawi Al-Hashemi, H.M. and Baghabra Al-Amoudi, O.S. A review on the angle of repose of granular materials. *Powder Technol.*, 330:397–417, 2018. doi: https://doi.org/10.1016/j.powtec.2018.02.003.

[5] Réka A., István, A., Hornbaker, D. and Schiffer, P. and Barabási, A.L. Maximum angle of stability in wet and dry spherical granular media. *Phys. Rev. E*, 56(6):R6271–R6274, Dec 1997. doi: 10.1103/PhysRevE.56.R6271.

[6] Zsamboky M., Fernández-Bilbao, A., Smith, D., Knight, J., Allan, J. *Impacts of climate change on disadvantaged UK coastal communities*. JRF. [Online Edition], 2011. `https://www.jrf.org.uk/report/impacts-climate-change-disadvantaged-uk-coastal-communities`, [Accessed May 2018].

[7] Acharya, T. and Ray, A.K . *Image Processing: Principles and Applications*. Hoboken, N.J: Wiley-Interscience, 2005.

[8] Adrian, R.J. and Westerweel, J. *Particle image velocimetry*. Cambridge: Cambridge University Press, 2011.

[9] Allsop, W. *Sea walls: a literature review*. HR Wallingford, [Online Edition], September 1986. `http://eprints.hrwallingford.co.uk/582/`, [Accessed February 2018].

[10] Balay, S., Abhyankar, S., Adams, M.F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., McInnes, L. C., Rupp, K., Smith, B. F., Zampini, S. and Zhang, H. *PETSc Users Manual*. Number ANL-95/11 - Revision 3.7. Argonne National Laboratory, 2016. `http://www.mcs.anl.gov/petsc`.

[11] Balay, S., Gropp, W. D., McInnes, L. C. and Smith, B. F. Efficient Management of Parallelism in Object Oriented Numerical Software Libraries. In Arge, E., Bruaset, A. M. and Langtangen, H. P., editor, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997. doi: https://doi.org/10.1007/978-1-4612-1986-6_8.

[12] Basco, D.R., and Yamashita, T. Toward a simple model of the wave breaking transition in surf zones. In *Costal Engineering 1986*, pages 955–970. [Online Edition]. doi: https://doi.org/10.1061/9780872626003.072. [Accessed April 2015].

[13] Battjes, J.A . Surf Similarity. *Proc. 14th Coastal Eng. Conf., Copenhagen, Denmark*, pages 466–480, 1974.

[14] Bokhove, O. and Kalogirou, A. Variational water wave modelling: from continuum to experiment. In Bridges, T., Groves, M. and Nicholls, D., editor, *Lectures on the Theory of Water Waves*, volume 426, pages 226–245. Cambridge University Press, Cambridge, 2016.

[15] Bokhove, O., van der Horn, A. J., van der Meer, D., Thornton, A. R. and Zweers, W. On wave-driven “shingle” beach dynamics in a table-top Hele-Shaw cell. In Lynett, P., editor, *Proceedings of ICCE’14*, 2014. doi: http://dx.doi.org/10.9753/icce.v34.sediment.41.

[16] Brocchini, M. . A reasoned overview on Boussinesq-type models: the interplay between physics, mathematics and numerics. *Proc.R.Soc. A: Mathematical, Physical and Engineering Sciences*, 469(2160):20130496, 2013. doi: 10.1098/rspa.2013.0496. URL `https://royalsocietypublishing.org/doi/abs/10.1098/rspa.2013.0496`.

[17] Brun, P.T., Nagel, M., and Gallaire, F. Generic path for droplet relaxation in microfluidic channels. *Phys. Rev. E*, 88(4):043009, 2013. doi: https://doi.org/10.1103/PhysRevE.88.043009.

[18] Burger, W. and Burge, Mark J. *Digital Image Processing: An Algorithmic Introduction Using Java*. London: Springer-Verlag, 2008.

[19] Ceniceros, H. D., Hou, T. Y., and Si, H. Numerical study of Hele-Shaw flow with suction. *Phys. Fluids*, 11(9):2471–2486, 1999. doi: https://doi.org/10.1063/1.870112.

[20] Cheshomi, A., Fakher, A. and Jones, C.J.F.P. A correlation between friction angle and particle shape metrics in Quaternary coarse alluvia. *Q. J. Eng. Geol*, 42(2):145–155, 2009. doi: https://doi.org/10.1144/1470-9236/07-052.

[21] Courant R., Friedrichs K. and Lewy H. On the partial difference equations of mathematical physics. *IBM J. Res. Dev.*, 11(2):215–234, 1967. doi: 10.1147/rd.112.0215.

[22] Dalwadi, M. P., Chapman, S. J., Waters, S. L., and Oliver, J. M. On the boundary layer structure near a highly permeable porous interface. *J. Fluid. Mech.*, 798:88–139, 2016. doi: 10.1017/jfm.2016.308.

[23] Danala, G., Patel, B., Aghaei, F., Heidari, M., Li, J., Wu, T., and Zheng, B. Classification of breast masses using a computer-aided diagnosis scheme of contrast enhanced digital mammograms. *Ann. Biomed. Eng.*, 46(9):1419–1431, 2018. doi: https://doi.org/10.1007/s10439-018-2044-4.

[24] Forsyth, A.J., Hutton, S. R., Rhodes, M.J. and Osborne, C.F. Effect of applied interparticle force on the static and dynamic angles of repose of spherical granular material. *Phys. Rev. E*, 63(3):031302–1 – 031302–5, 2001.

[25] Fowler, A. *Mathematical Geoscience*. Springer, 2011.

[26] Franci, F., Mandanici. E., and Bitelli, G. Remote sensing analysis for flood risk management in urban sprawl contexts. *Geomat. Nat. Haz. Risk*, 6(5-7):583–599, 2015. doi: https://doi.org/10.1080/19475705.2014.913695.

[27] Franz, G., Delpey, M. T., Brito, D., Pinto, L., Leitão, P. and Neves, R. Modelling of sediment transport and morphological evolution under the combined action of waves and currents. *OSJ*, 13(5):673–690, 2017. doi: https://doi.org/10.5194/os-13-673-2017.

[28] Fredsoe, J. and Deigaard, R. *Mechanics of coastal sediment transport*. World Scientific, 1992.

[29] French, P.W. *Coastal defences : Processes, problems and solutions*. London: Routledge, 2001.

[30] Gagarina, E., Ambati, V.R., van der Vegt, J.J.W. and Bokhove, O. Variational space-time (dis)continuous Galerkin method for nonlinear free surface water waves. *J.Comput.Phys.*, 275:459–483, 2014. doi: https://doi.org/10.1016/j.jcp.2014.06.035.

[31] Galvin, C.J. Breaker type classification on three laboratory beaches. *JGR*, 73(12):3651–3659, 1968. doi: https://doi.org/10.1029/JB073i012p03651.

[32] Paul Gillett. Rough seas near Saltdean. `http://www.geograph.org.uk/photo/2541914`, 2011.

[33] Gonzalez, R.C. and Woods, R.E . *Digital Image Processing*. Upper Saddle River, N.J. : Pearson Prentice Hall, 3rd edition, 2008.

[34] Grilli, S.T., Skourup, J., and Svendsen, I.A. An efficient boundary element method for nonlinear water waves. *Eng. Anal. Bound. Elem.*, 6(2):97–107, 1989. doi: https://doi.org/10.1016/0955-7997(89)90005-2.

[35] Hairer, E., Lubich, C., and Wannner, G. *Geometrical numerical integration: structure-preserving algorithms for ordinary differential equations*. Berlin, Great Britain: Springer, 2nd edition, 2006.

[36] Hanson, H., Brampton, A., Capobianco, M., Dette, H.H., Hamm, L., Laustrup, C., Lechuga, A., and Spanhoff, R. Beach nourishment projects, practices, and objectives–a European overview. *Coast. Eng. J.*, 47(2):81–111, 2002. doi: https://doi.org/10.1016/S0378-3839(02)00122-9.

[37] Hele-Shaw, H. S. The Flow of Water. *Nature*, 58:34–36, 1898. doi: http://dx.doi.org/10.1038/058034a0.

[38] Helluy, P., Golay, F., Caltagirone, J.P., Lubin, P., Vincent, S., Drevard, D., Marcer, R., Fraunié, P., Seguin, N., Grilli, S., Lesage, A.C., Dervieux, A., and Allain, O. Numerical simulations of wave breaking. *ESAIM: M2AN*, 39(3):591–607, 2005. doi: https://doi.org/10.1051/m2an:2005024.

[39] Hiscock, K.M. and Bense, V.F. *Hydrogeology: principles and practise*. Chichester, West Sussex: John Wiley & Sons, 2nd edition, 2014.

[40] Hudson, T., Keating, K., and Pettit, A. . *Cost estimation for coastal protection - summary of evidence*. EA, [Online Edition], 2015. `http://evidence.environment-agency.gov.uk/FCERM/Libraries/FCERM_Project_Documents/SC080039_cost_coastal_protection.sflb.ashx`, [Accessed May 2017].

[41] Inch., K.W. Surf Zone Hydrodynamics: Measuring Waves and Currents. In Cook, S., Clarke, L., and Nield, J., editor, *Geomorphological Techniques*. Book Publishing Inc., [Online Edition], 2014. `https://geomorphology.org.ukgeomorph_techniques`, [Accessed April 2018].

[42] Kalogirou, A., Moulopoulou, E.E., and Bokhove, O. Variational finite element methods for waves in a Hele-Shaw tank. *Appl. Math. Model.*, 40(17-18):7493–7503, 2016. doi: http://dx.doi.org/10.1016/j.apm.2016.02.036.

[43] Komar, P.D. *Beach processes and sedimentation*. Englewood Cliffs, N.J: Prentice-Hall, 1976.

[44] Lachaume, C., Biausser, B., Grilli, S.T., Fraunie, P., and Guignard, S. Modeling of breaking and post-breaking waves on slopes by coupling of BEM and VOF methods. In *Proceedings of ISOPE'03*, pages 353–359, 2003.

[45] Lesser, G.R., Roelvink, J.A., van Kester, J.A.T.M., Stelling, G.S. Development and validation of a three-dimensional morphological model. *Coast. Eng. J.*, 51(8):883–915, 2004. doi: https://doi.org/10.1016/j.coastaleng.2004.07.014.

[46] Leveque, R.J. *Numerical methods for conservation laws*. Basel: Birkhäuser Verlag, 2nd edition, 1992.

[47] Leveque, R.J. *Finite Volume Methods for Hyperbolic Problems*. Cambridge: Cambridge University Press, 2002.

[48] López de San Román-Blanco, B., Coates, T.T., Holmes, P., Chadwick , A.J., Bradbury, A., Baldock, T.E., Pedrozo-Acu$\tilde{n}$a, A., Lawrence, J., and Grüne, J. Large scale experiments on gravel and mixed beaches: Experimental procedure, data documentation and initial results. *Coast. Eng. J.*, 53(4):349–362, 2006. doi: https://doi.org/10.1016/j.coastaleng.2005.10.021.

[49] Pevensey Coastal Defence Ltd. Recharge. `http://www.pevensey-bay.co.uk/recharge.html`.

[50] Titan Environmental Surveys Ltd. Coastal erosion at Minehead, Somerset. Teignbridge District Council, 2010.

[51] Luke, J.C. A variational principle for a fluid with a free surface. *J. Fluid Mech.*, 27(2): 395–397, 1967. doi: https://doi.org/10.1017/S0022112067000412.

[52] Marsden, J.E., and Ratiu, T.S. *Introduction to mechanics and symmetry: a basic exposition of classical mechanical systems*. New York: Springer, 2nd edition, 1999.

[53] Masselink G. and Russell P. Impacts of climate change on coastal erosion. *MCCIP*, pages 71–86, 2013. doi: https://doi.org/10.14465/2013.arc09.071-086.

[54] Mayers, D.F. *Numerical Solution of Partial Differential Equations: An Introduction*. Cambridge University Press, 2nd edition, 2005.

[55] McLachlan, N. Cliff erosion at Ulrome, East Riding of Yorkshire. East Riding of Yorkshire Council, 2009.

[56] Mei, C. C. *The Applied Dynamics of Ocean Surface Waves*. Singapore; London: World Scientific, 1989.

[57] Meignin, L., Gondret, P., Ruyer-Quil, C., and Rabaud, M. Subcritical Kelvin-Helmholtz Instability in a Hele-Shaw Cell. *Phys. Rev. Lett.*, 90(23):234502, 2003. doi: 10.1103/PhysRevLett.90.234502.

[58] Miles, J. W. On Hamilton's principle for surface waves. *J. Fluid. Mech.*, 83(1):153–158, 1977. doi: 10.1017/S0022112077001104.

[59] Mimura, N. Sea-level rise caused by climate change and its implications for society. *Proc. Jpn. Acad. Ser. B Phys. Biol. Sci.*, 89(7):281–301, 2013. doi: 10.2183/pjab.89.281.

[60] Montreuil A.L. and Bullard J.E. A 150-year record of coastline dynamics within a sediment cell: Eastern England. *Geomorphology*, 179:168–185, 2012. doi: https://doi.org/10.1016/j.geomorph.2012.08.008.

[61] Muir Wood, A. M. Characteristics of shingle beaches the solution to some practical problems. In *Coastal Engineering 1970 Proceedings*. ASCE, 1970.

[62] Nicholson, J., Broker, I., Roelvink, J.A., Price, D., Tanguy, J.M., Moreno, L. Intercomparison of coastal area morphodynamic models. *Coast. Eng. J.*, 31(1):97 – 123, 1997. doi: https://doi.org/10.1016/S0378-3839(96)00054-3.

[63] Nobuyuki, O. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man. Cybern. B Cybern.*, 9(1):62–66, 1979. doi: 10.1109/TSMC.1979.4310076.

[64] Olver, P.J. *Applications of Lie groups to differential equations*. New York: Springer-Verlag, 2nd edition, 1993.

[65] Tassi P. *Numerical modelling of river processes: flow and river bed deformation*. PhD thesis, University of Twente, 2007.

[66] Park, C.W., and Homsy, G. M. Two-phase displacement in Hele Shaw cells: theory. *J. Fluid. Mech.*, 139:291–308, 1984. doi: https://doi.org/10.1017/S0022112084000367.

[67] Park, C.W., Gorell, S., and Homsy, G. M. Two-phase displacement in Hele-Shaw cells: experiments on viscously driven instabilities. *J. Fluid. Mech.*, 141:275–287, 1984. doi: https://doi.org/10.1017/S0022112084000847.

[68] Peregrine, D.H. Breaking waves on beaches. *Ann. Rev. Fluid. Mech.*, 15(1):149–178, 1983. doi: https://doi.org/10.1146/annurev.fl.15.010183.001053.

[69] Peregrine, D.H., and Svendsen, I.A. Spilling breakers, bores, and hydraulic jumps. In *Coastal engineering 1978*, pages 540–550. [Online Edition], 1978. doi: https://doi.org/10.1061/9780872621909.032. [Accessed April 2015].

[70] Peterson, C.H. and Bishop, M.J. Assessing the environmental impacts of beach nourishment. *BioScience*, 55(10):887–896, 2005. doi: https://doi.org/10.1641/0006-3568(2005)055[0887:ATEIOB]2.0.CO;2.

[71] Pihler-Puzović, D., Peng, G. G., Lister, J. R., Heil, M. and Juel, A. Viscous fingering in a radial elastic-walled Hele-Shaw cell. *J. Fluid. Mech.*, 849:163–191, 2018. doi: https://doi.org/10.1017/jfm.2018.404.

[72] Plataniotis, K. N. and Venetsanopoulos, A. N. *Color Image Processing and Applications*. Berlin; London: Springer, 2000.

[73] Plouraboué, F., and Hinch, E. J. . Kelvin-helmholtz instability in a hele-shaw cell. *Phys. Fluids*, 14(3):922–929, 2002. doi: 10.1063/1.1446884. URL `https://doi.org/10.1063/1.1446884`.

[74] Popinet, S. An accurate adaptive solver for surface-tension-driven interfacial flows. *J. Comput. Phys.*, 228(16):5838–5866, 2009. doi: https://doi.org/10.1016/j.jcp.2009.04.042.

[75] Powell, K. A. *Predicting short term profile response for shingle beaches*. HR Wallingford, [Online Edition], 1990. `http://eprints.hrwallingford.co.uk/1085/1/SR219.pdf`, [Accessed May 2015].

[76] Rajchenbach, J., Leroux, A. and Clamond, D. New standing solitary waves in water. *Phys. Rev. Lett.*, 107(2):024502, 2011. doi: https://doi.org/10.1103/PhysRevLett.107.024502.

[77] Rangayyan, R.M. *Biomedical image analysis*. CRC Press, 2005.

[78] Rao S.S. *The finite element method in engineering*. Amsterdam; Boston, MA: Elsevier/Butterworth Heinemann, 4th edition edition, 2005.

[79] Rathgeber, F., Ham, D. A., Mitchell, L., Lange, M., Luporini, F., Mcrae, A . T. T., Bercea, G. T., Markall, G. R. and Kelly, P. H. J. Firedrake: Automating the Finite Element Method by Composing Abstractions. *ACM Trans. Math. Softw.*, 43(3):24:1–24:27, 2017. doi: http://doi.acm.org/10.1145/2998441.

[80] Rehman, K., and Cho, Y.S. Bed Evolution under Rapidly Varying Flows by a New Method for Wave Speed Estimation. *Water*, 8(212), 2016.

[81] Rhebergen, S., Bokhove O., and van der Vegt, J.J.W. . Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. *J. Comput. Phys.*, 227(3):1887–1922, 2008. doi: https://doi.org/10.1016/j.jcp.2007.10.007.

[82] Richards, J.A., and Xiuping, J. *Remote sensing digital image analysis: An Introduction*. Berlin: Springer-Verlang, 4th edition, 2006.

[83] Roelvink, D. and Reniers, A. *A guide to modeling coastal morphology*. Singapore; London: World Scientific, 2012. doi: 10.1142/7712.

[84] Saffman, P. G., and Taylor G. The Penetration of a Fluid into a Porous Medium or Hele-Shaw Cell Containing a More Viscous Liquid. *Proc. Royal Soc. A*, 245(1242):312–329, 1958.

[85] Santamarina, J.C. and Cho, G.C. Soil behaviour: The role of particle shape. In *Advances in geotechnical engineering: The Skempton conference*, pages 604–617. London: Thomas Telford, 2004.

[86] Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., Tinevez, J.Y., White, D.J., Hartenstein, V., Eliceiri, K., Tomancak, P., and Cardona, A. Fiji: an open-source platform for biological-image analysis. *Nat. Methods*, 9:676–682, 2012. doi: http://dx.doi.org/10.1038/nmeth.2019.

[87] Shields, A. Anwendung der Aehnlichkeitsmechanik und der Turbulenzforschung auf die Geschiebebewegung. 1936.

[88] Soulsby, R. *Dynamics of marine sands*. London: Thomas Telford Publishing, 1997. doi: 10.1680/doms.25844.

[89] Kent T. *An idealised fluid model of Numerical Weather Prediction: dynamics and data assimilation*. PhD thesis, School of Mathematics, Leeds, 2016.

[90] Thornton, A. R., van der Horn, A. J., Gagarina, E., Zweers, W., van der Meer, D. and Bokhove, O. Hele-Shaw beach creation by breaking waves: a mathematics-inspired experiment. *Environ. Fluid Mech.*, 14(5):1123–1145, 2014. doi: http://dx.doi.org/10.1007/s10652-014-9350-7.

[91] Toro, E. F., Spruce, M. and Speares, W. Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves*, 4(1):25–34, 1994. doi: https://doi.org/10.1007/BF01414629.

[92] Toro, E.F. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Berlin: Springer, 2nd edition, 1999.

[93] Toro, E.F. *Shock-Capturing Methods for Free-Surface Shallow Flows*. Chichester: John Wiley, 2001.

[94] Torres-Freyermuth, A., Losada, I. J., Lara, J. L. Modeling of surf zone processes on a natural beach using Reynolds-Averaged Navier-Stokes equations. *J. Geophys. Res. Oceans*, 112(C9), 2007. doi: https://doi.org/10.1029/2006JC004050.

[95] Usami, S., Chen, H.H., Zhao, Y., Chien, S., and Skalak, R. Design and construction of a linear shear stress flow chamber. *Ann. Biomed. Eng.*, 21(1):77–83, 1993. doi: https://doi.org/10.1007/BF02368167.

[96] van der Meer, J.W. *Rock slopes and gravel beaches under wave attack*, volume 396. Delft Hydraulics Publications, [Online Edition], 1988. `uuid:67e5692c-0905-4ddd-8487-37fdda9af6b4`, [Accessed May 2015].

[97] van Rijn, L.C. Coastal erosion and control. *Ocean. Coast. Manag.*, 54(12):867–887, 2011. doi: https://doi.org/10.1016/j.ocecoaman.2011.05.004.

[98] Viola, F., Gallaire, F., and Dollet, B. Sloshing in a Hele-Shaw cell: experiments and theory. *J. Fluid. Mech*, 831:R1, 2017. doi: https://doi.org/10.1017/jfm.2017.650.

[99] Vitousek, S., Barnard, P. L., Fletcher, C. H., Frazer, N., Erikson, L., Storlazzi, C. D. Doubling of coastal flooding frequency within decades due to sea-level rise. *Scientific Reports-Nature*, 7, 2017. doi: https://doi.org/10.1038/s41598-017-01362-7.

[100] Vrieling, A. Satellite remote sensing for water erosion assessment: A review. *CATENA*, 65 (1):2–18, 2006. doi: https://doi.org/10.1016/j.catena.2005.10.005.

[101] Whitham G.B. *Linear and Nonlinear Waves*. New York; Chichester: Wiley, 1999.

[102] Wu, H., Cheng, Z., Shi, W., Miao, Z., and Xu, C. An object-based image analysis for building seismic vulnerability assessment using high-resolution remote sensing imagery. *Nat. Hazards*, 71(1):151–174, 2013. doi: https://doi.org/10.1007/s11069-013-0905-6.

[103] Yang, F.G., Liu, X.N., Yang K.J. and Cao S.Y. Study on The Angle of Repose of Nonuniform Sediment. *J. Hydrodyn. B*, 21(5):685–691, 2009. doi: https://doi.org/10.1016/S1001-6058(08)60200-0.

[104] Zhangn, C., Walters, D., and Kovacs, J.M. Applications of Low Altitude Remote Sensing in Agriculture upon Farmers Requests: A Case Study in Northeastern Ontario, Canada. *PloS One*, 9(11):1–9, 2014. doi: https://doi.org/10.1371/journal.pone.0112894.