Three-Dimensional Numerical Simulations of Subaerial Landslide Generated Waves

by

William Daley Clohan

B. Engineering, McGill University, 2008

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Applied Science

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES
(Civil Engineering)

The University of British Columbia
(Vancouver)

January 2015

Abstract

This research aims to advance the continuing effort of general purpose computational fluid dynamics model validation of subaerial landslide generated wave (SLGW) simulations. Specifically, using the open source program weakly compressible Smooth Particle Hydrodynamics model, DualSPHysics, three-dimensional simulations are quantitatively compared against a combination of physical model data and traditional general-purpose computational fluid dynamics, Flow-3D^{TM}, data.

Many simulations were conducted to determine the effect of both numerical parametrization and numerical scheme prescriptions on SLGW accuracy. A systematic approach was taken to parse out insignificant physical processes using Flow-3D^{TM} - specifically surface tension - and to determine the optimal numerical scheme settings that yield the most accurate results for both Flow-3D^{TM} and DualSPHysics.

From this research, it is found that DualSPHysics is able to accurately simulate both wave generation and wave propagation, but tends to over-predict the maximum wave run-up by about 70%. In contrast, Flow-3D^{TM} was able to accurately simulate wave propagation, but under predicted wave generation by about 25% and over predicted the maximum wave run-up by about 40%.

The question as to why both DualSPHysics and Flow-3D^{TM} both over predict the maximum wave run-up during a SLGW simulation is still open. However, it is speculated that this due to a lack of either energy dissipation through air entrainment or eigenfrequency consideration's.

Preface

This thesis is an original and unpublished work by W.D. Clohan.

Table of Contents

List of Tables

List of Figures

Acknowledgments

I would like to express my gratitude to my supervisor, Professor Bernard Laval. His encouragement to explore my interests made this research both enjoyable and enriching.

I wish to also thank the Natural Sciences and Engineering Research Council of Canada for my Industrial Postgraduate Scholarship. In addition, I would like to express my gratitude to the University of British Columbia, as the UBC Civil Engineering Flood Fund provided generous contributions to my research.

Finally, I would like to acknowledge my wife Jarleen. Without her support, confidence, and patience this thesis would not be what it is.

Chapter 1

Introduction

Landslide generated waves (LWG) are gravity dominated waves that are generated by a sudden disturbance, or impulsive force, in a body of water.

As a landslide rapidly interacts with a body of water, a large volume of water is displaced and then forms a large, fast-moving wave. The generated wave can then travel vast distances with minimal attenuation until it collides with and runs-up a shoreline. This phenomenon is extremely complex and difficult to evaluate. One way to analyze a LGW problem is by breaking it into a wave generation phase, a wave propagation phase, and a wave run-up phase, as shown in Figure 1.1

Physically, the generation phase is a transfer of potential energy (from a slide) to kinetic energy (motion of water) through either a landslide that originates entirely on land (subaerial landslide), originates entirely underwater (subaqueous landslide), or originates partially on land and partially underwater (partially submerged landslide). The amplitude of the generated wave is dependent on the slide's volume, velocity, density, porosity, shape, and hill slope angle[59].

Subaqueous and partially submerged landslides are infamously known for generating destructive Tsunami waves in excess of 20-m[59] high, either in the open ocean (e.g. the 1883 Krakatau Tsunami where 36,000 people died or the 1998 Papua New Guinea Tsunami where 2,200 people died) or inland, on man-made reservoirs (e.g. the 1960 Vajont partially submerged landslide that killed roughly 2,000

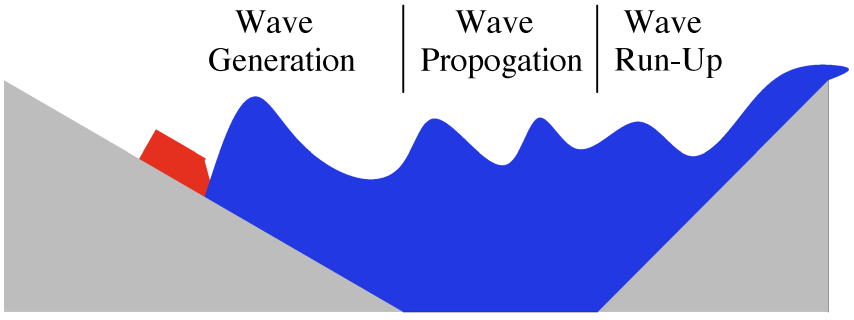

Figure 1.1: Phases of landslide generated waves

people). In contrast, subaerial landslides tend to only be known for their destruction when occurring inland, in natural or man-made reservoirs. For example, of the roughly 15,600[67] people who have died because of subaerial landslide generated waves, only two were killed in a non-inland event in Lituya Bay, Alaska (July 9th, 1958). The remaining causalities all perished in one of seven inland subaerial landslide generated wave (SLGW) events: Five in Norway - one in Rammerfjell (17 perished on January 8th, 1741), one in Tjelle (32 perished February 22nd, 1756), two in Ravenfjell (61 perished January 15th, 1905 and 72 more on September 13th, 1936), and one in Tafjord (41 perished on April 7th, 1934); one in Shimabara, Japan (roughly 15,000 perished on May 21st, 1792); and, one in Yanahuin Lake, Peru (400-600 perished March 18th, 1971).

Despite the fact that subaerial, subaqueous, and partially submerged landslides all generate waves when they interact with a body of water, the underlying physics of each is significantly different. This thesis will focus solely on inland subaerial landslide generated waves - herein referred to as SLGW.

The wave propagation phase encompasses wave energy dispersion[1] that characteristically changes during times of shoaling, diffracting, and refracting. The wave run-up phase looks at when a wave hits a boundary (like a shoreline), and tends to run-up the boundary to an elevation several times that of its original approaching height.

The three phase framework has worked well for analyzing SLGW and there has been a wide breadth of research into subaerial landslide wave generation, wave propagation, and wave run-up. However, there is no "one" way of integrating findings from all the relevant research. Instead, the conscientious engineer needs to use either judgement in applying analytical/empirical relationships between subaerial landslides and a body of water's response, or use scaled physical models or computational fluid dynamics (CFD) models for case specific situations.

Again, this research is limited to waves generated by inland subaerial landslides, and the objective is to correlate the characteristics of a subaerial landslide to the characteristics of the water wave it generates upon impact. More specifically, freely dropping or sliding a rigid or deformable mass (herein refereed to as the forcing mechanism) down a 0-90° slope into still water of finite volume and developing an empirical relation between the motion of the forcing mechanism and the height of the wave(s) it generates - most notably the maximum wave height.

1.1 Literature review

The following literature review is broken down into wave generation research, wave propagation research, wave run-up research, and SLGW research.

1.1.1 Wave generation research

The birth of subaerial landslide wave generation research started in 1845 when John Scott Russell published his "Report on Waves"[65]. Here, Russell discussed his findings on solitary wave generation

[1] The reader is referred to a text (like Sorensen's Basic Coastal Engineering book[69]) on linear, Stokes, Cnoidal, Solitary, and bore wave theory

when a free falling vertical box was dropped into a flume of water[2], and he concluded that a wave's volume is identical to the initial water displacement by the vertical box, which was the first reported relationship between a forcing mechanism and the waves it generates.

Following Russell's 1845 paper, there was a 104-year void in subaerial landslide wave generation research. It was not until 1949, when Johnson and Bermel[35] dropped circular steel discs into a flume of shallow water, did SLGW research pick up again[3]. That said, their research was not overly fruitful. Aside from using data from five tests to show a linear relationship between the fall height of a 472-kg 1.2-m diameter steel disc and the wave length of the first resulting wave, not much can be inferred from their work.

The first comprehensive study of SLGW was undertaken by Kamphuis and Bowering in 1970[36]. Using a two-dimensional (2D) 45-m long and 1-m wide physical model, they released a rigid sled of variable weight from a variety of locations down a roller ramp of variable slope into water depths ranging from 0.23-0.46-m. During each test run, the sled velocity was recorded as well as wave heights at three locations. Analysis allowed them to construct a dimensionless equation (Equation 1.1) that relates the dimensionless maximum wave height at a point, $\eta_{max,dim} = \eta_{max}/h$ (where η_{max} is the maximum wave height and h is the still water depth) , to slide Froude number, $\mathbf{F} = v_s/\sqrt{gh}$ (where v_s is the slide impact velocity), dimensionless slide volume, $\forall = V_s/wh^2$ (where V_s is the slide volume and w is the slide width), specific gravity of the slide, $G = \rho_s/\rho_w$ (where ρ_s is the bulk density of the slide and ρ_w is the density of water), dimensionless propagation distance, $X = x/h$ (where x is the distance from impact), dimensionless time slide spends underwater, $T_s = t_s\sqrt{g/h}$ (where t_s is the underwater motion time), landslide front slope, Φ (in radians), slope angle, α (in radians), and landslide porosity, n:

$$\eta(x)_{max,dim} = \Phi\big(\mathbf{F}, \forall, G, X, T_s, \Phi, \alpha, n\big) \tag{1.1}$$

They found the most influential parameters to be the slide Froude number, $\boldsymbol{F}$, and the dimensionless slide volume, $\forall$.

In the same year as Kamphuis and Bowering, Noda[57] and Wiegel[84] performed their own experiments using a Scott Russell wave generator (SRWG). They also found that the maximum wave height generated was highly influenced by the slide Froude number. Furthermore, Noda[57] suggested that depending on the box width and velocity, the waves generated from the SRWG could be analyzed using either linear or solitary wave theory (see Section 1.1.2)

Expanding on Kamphuis and Bowering's work[36], Huber[33] conducted one of history's most comprehensive SLGW studies. He used granular material to represent a subaerial slide, and carried out over 1000 2D and 3D test runs. During which, he used a slide mass, m_s, ranging from 5-50-kg, a slide impact velocity ranging from 1-5-m/s, a water depth ranging from 0.12-0.36-m, and a slope angle of 28° to 60°. Rounded river rock with a mean diameter of 0.002-m was used for the slide material.

Analysis of Huber's (1980) work by Huber and Hager[34] generated the first relationship between

[2]This method of generating waves has since been coined the "Scott Russell wave generator" and is used by many SLGW researchers.

[3]Johnson and Bermel's motivations were not actually provoked by SLGW research, but rather assisting with the planning of atomic bomb testing near the Bikini Atoll.

a granular subaerial slide and the maximum wave height at a specific point. However, unlike Kamphuis and Bowering[36], Noda[57], and Wiegel[84], they did not take into account the slide Froude number. Instead they suggested the maximum generated wave height at a point was mainly dependent on the non-dimensional landslide volume, and could be described in 2D as:

$$\eta(x)_{max} = 0.88 \cdot h^4 \forall^{1/2} G^{1/4} X^{-1/4} sin\alpha \tag{1.2}$$

or in 3D (note, r is the radial distance away from the slide impact):

$$\eta(x)_{max} = 1.76 \cdot sin\alpha \cdot cos^2 \left[\frac{2\gamma \forall^{1/2} G^{1/4}}{3} \left(\frac{r}{h} \right)^{-2/3} \right] \tag{1.3}$$

Slingerland and Voight[68], used the slide's dimensionless kinetic energy to relate the maximum wave generated at a location $r/h \approx 4$ from the slide impact as:

$$log\left(\frac{\eta(x = r/h \approx 4)_{max}}{h} \right) = -1.25 + 0.71 \cdot log\left(0.5 \frac{\rho_s}{\rho_w} \frac{V_s}{h^3} \frac{v_s^2}{gh} \right) \tag{1.4}$$

or

$$log\left(\frac{\eta(x = r/h \approx 4)_{max}}{h} \right) = -1.25 + 0.71 \cdot log\left(\frac{0.5 w \mathbf{F}^2 \forall G}{h} \right) \tag{1.5}$$

where they arrived at Equation 1.5 through empirical of data generated by a scaled physical model study of subaerial landslides into the Mica dam reservoir in British Columbia, Canada[81]. They then used Equation 1.5 to reasonably reproduce the Lituya Bay maximum generated wave at $r/h \approx 4$.

Walder et al.[80], felt that the maximum wave generated may be influenced by a slide's Froude number (Kamphuis and Bowering[36], Noda[57], and Wiegel et al.[84]), dimensionless volume(Kamphuis and Bowering[36],Huber and Hager[34]), or dimensionless kinetic energy(Slingerland and Voight[68]), but these were not the dominating relationships. Instead Walder et al.[80] argue the key relationship is the ratio of the dimensionless submerged slide travel time to the dimensionless slide volume, $T_s/\forall$. Where the basis for their argument is founded on 2D results from a physical experiment they carried out in a 1-m long, 0.285-m wide flume with 0.051-m to 0.13-m of water, and a rigid variably weighted box that slid down a concave slide. They also strengthened their argument by "reasonably" using this ratio to approximate the Lituya Bay event[47].

Panizzo[60] also advocated that the dominate parameter for the maximum SLGW was the dimensionless submerged slide travel time, but he also valued the dimensionless slide volume. Having said that, his experiments were not 2D, but actually 3D experiments in a 12-m long, 6-m wide, and 0.8-m deep tank, and his conclusion differs from Atai-Ashtiani and Nik-Khah's[2] 3D experimental conclusions.

Atai-Ashtiani and Nik-Khah's[2] used a 25-m long, 2.5-m wide, and 1.8-m deep tank with six uniformly placed wave measuring devices along the tank centreline. They then slid a variety of rigid shapes (i.e. square, short rectangle, tall rectangle, triangle, short/narrow hemisphere, etc.) and deformable

shapes (i.e. granular material in fabric bags) down a variable sloped ramp into water. According to them, the maximum generated 3D wave is "strongly affected" by the slide Froude number, ramp slope, and slide rigidity, but "weakly affected" by ramp slope. For slide rigidity, they found a 35% decrease in the maximum generated wave when using a deformable shape, as opposed to a rigid object. They also suggest the following dimensionless relationship:

$$\frac{\eta_{max}}{h} = \left(0.405 + 0.078(\forall \mathbf{F}^2)^{1.28}\right)\left(\frac{T_s}{\forall}\right)\left(\frac{l}{s}\right)^{-0.12}\left(\frac{x}{h}\right)^{-0.48} \tag{1.6}$$

Where l is the slide length and s is the slide thickness and Equation 1.6 is based on the aggregate data from Kamphuis and Bowering in 1970[36], Huber[33], Walder et al.[80], as well as Atai-Ashtiani and Nik-Khah[2], which greatly improves the reliability of the above equation. That said, what they - and the many other researcher's before them - neglected to focus on, was the underlying physics of a subaerial landslide impacting water. That is, SLGW research was considered a black box type of research. In that, researcher's only focused on a slides pre-impact velocity, shape, volume, kinetic energy, etc. (i.e. "black box" input) and the generated wave(s) occurring post-impact (i.e. "black box" output). However, the PhD. work of Fritz[19] attempted to fill that void.

Fritz[19] constructed an ingenious flexible 2D physical experiment to pragmatically evaluate granular SLGW events, the foundation of which was based on his pneumatic wave maker. He recognized that changes in the slide Froude number would not always generate a 2D wave in his model. That is, assuming a constant volume, the higher the material is placed up on his slide, the more it will tend to thin out and generate a "tongue" like structure where the central portion would impact the water much earlier then the sides. This implies, waves would also be generated radial and inevitably reflect off the model walls. Not only that, for each variation of slide Froude number, the shape of the slide would always change. Using his pneumatic wave maker, Fritz was able to accelerate granular material at different rates while maintaining the front shape of the material[4]. Using particle image velocimetry, Fritz[19] was able to quantify the impact zone physics and show once again that the slide Froude number is one of the dominating parameters, which he showed by reproducing the Lituya Bay event using dynamic similitude [20, 21, 47].

As an extension of Fritz's work[19], Zweifel[85] and Heller[29, 29, 30] used Fritz's experimental setup to conduct their own research. Zweifel showed that the higher a slide's density, the higher the maximum generated wave would be, also that the slide Froude number was the dominant parameter for "slow" slides whereas the water depth and slide thickness were the dominant parameters for "fast" slides.

Heller[27, 28, 30] used Fritz's[19] 223 test runs and his 211 runs to generate (what he calls) a universal impact parameter, **P**, to describe the maximum generated SLGW within $\pm 30\%$:

$$\mathbf{P} = \mathbf{F}S^{1/2}M^{1/4} \cdot \sqrt{cos[(6/7)\alpha]} \tag{1.7}$$

[4]The granular material was placed in a rigid box that was pneumatically accelerated, then at a given location, the front of the box would open and the material would be shot out[19]

With Heller's impact parameter taking into account the slide Froude number, the slide dimensionless mass, $M = m_s/(\rho_w w h^2)$, and the slide dimensionless thickness, $S = s/h$, he argues this parameter is the most valid parameter for estimating the maximum generated wave height caused by subaerial landslides. However, he does indicate that this parameter is only valid for 2D and in his own words states 'the future in terms of an efficient and accurate prediction of the wave features for a specific bathymetry lies in numerical simulations'.

Following Heller's advice (albeit fourteen years before Heller said it!), Monaghan and Kos[53] used a robust 2D weakly-compressible Smoothed Particle Hydrodynamics (SPH) numerical model[22, 44, 51] (discussed in detail in Section 2.3) to investigate waves generated using both a numerical and physical SRWG. The results indicated their SPH model overshot their physical model data by 3-18%; however, they did find that the amplitude of the leading wave, A_1, could be described as:

$$A_1 = 3h^{-1/3}\left(\frac{m_s}{40\rho_w w}\right)^{2/3} \tag{1.8}$$

Later in 2008, Ataie-Ashtiani[3] conducted the same type of experiment with a SRWG, but this time they used a 2D incompressible model. Unfortunately, they did not compare their physical experimental data with their numerical data, but rather their analytical solution to the SRWG problem (i.e. solitary wave theory - see Section 1.1.2).

Monaghan et al.[55] conducted a 2D physical experiment where a weighted rigid box slid down a concave slope into water. They then ran 2D SPH simulations and compared the two results. They found an error between the two to be between 15-50%. This led them to concluded the 2D SPH model qualitatively compared well with the physical experiment, instead of quantitatively.

Das et al.[14] qualitatively looked at a 2D numerical simulation of a triangular wedge sliding into water of 1-m in depth. They used both Flow-3D^{TM} and an open source SPH code to conclude that Flow-3D^{TM} "appears to better predict" experimental free surface elevations. Qiu[61] and Viroulet et al.[79] also evaluated a 2D SPH numerical simulation of a triangular wedge sliding into water, and found the SPH code to be reasonable.

Finally, using an experiment similar to Atai-Ashtiani and Nik-Khah[2], Liu et al.[43] constructed a 3D 104-m long, 3.7-m wide, and 4.6-m deep flume and ran either a triangular or hemispherical sled into the water. They then used the physical data to "validate" a numerical code of theirs, the results of which appeared qualitatively accurate.

Despite much research into SLGW, there appears to be no real consensus on what the dominating parameters are. Nevertheless, most researches agree that the important parameters are the slide Froude number, **F**, dimensionless volume, $\forall$, and dimensionless mass, M, which suggests the likely most suitable parameter is Heller's universal impact parameter[30], **P**. That said, even Heller cautions this parameter is really only suitable for 2D situations and that 3D numerical site specific research needs to be carried out.

For the CFD community, this is an invitation to numerically investigate 3D SLWG events. However, only a handful of researchers have conducted qualitative research, and even less have done quantitative

research. As such, there is a void that needs to be filled.

1.1.2 Wave propagation research

The field of wave propagation research is one in which an attempt is made to mathematically describe (either analytically or empirically) how waves move through water of varying depths, h. Specifically it focuses on quantifying the speed at which waves move through water, c, and the energy/power they carry with them.

Generally speaking, there are many formulations based on relating wave height, H, length, L, and period, T, to wave celerity and energy; however, the main theories used for SLGW research are linear wave theory, solitary wave theory, Stokes wave theory, and cnoidal wave theory. The following discussion will focus on these four theories.

In 1841, George Biddell Airy published the first mathematical construct to evaluate waves propagating through water (known today as the linear wave theory) in his highly influential paper titled "Tides and Waves"[1, 9]. Airy assumed pressure is hydrostatic and flow is irrotational, inviscid, and incompressible. And he suggests that the temporal free surface, $\eta(x,t)$, of a linear periodic wave be expressed as a function of wave height, length, and period as it propagates through water. That is:

$$\eta(x,t) = \frac{H}{2} cos(\frac{2\pi}{L}x - \frac{2\pi}{T}t) \tag{1.9}$$

Although Airy's linear wave theory was a giant leap forward in the field of fluid dynamics, the theory had many limitations and caveats. Namely, the theory is a sinusoidal first-order accurate small amplitude wave theory that is only valid, according to Dean[15], for $H/h < 0.3$, $H/L < 0.006$, and $HL^2/h^3 << 1$[27]. Furthermore, the theory is linear, meaning it negates higher order non-linear terms which become important when a wave height relative to the flow depth and wavelength increases past 0.3 and 0.006, respectively.

Joseph Valentin Boussinesq, however, did account for non-linear effects in his 1871 mathematical construct of solitary waves[5]. Here, Boussinesq assumed a non-hydrostatic[5] rotational flow condition to mathematically confirm John Scott Russell's solitary wave generation observations made in 1845[65].

Boussinesq's theory implies that the non-linear temporal free surface of a solitary wave could be expressed as a function of wave amplitude, a, and speed in the following manner:

$$\eta(x,t) = asech^2\left(\sqrt{\frac{3a}{4h^3}}(x-ct)\right) \tag{1.10}$$

Where c is defined by Russell[65] as:

$$c = \sqrt{g(h+a)} \tag{1.11}$$

The major caveat to the solitary wave theory is that it is based on the assumption that there exists only one wave peak and no wave trough, which implies an infinite wave length. This of course is not

[5] Implying pressure is not simply the specific gravity of water times its depth (or height above a given point).

practical for engineering applications and arguably has no real use for real world applications.

In 1847, George Gabriel Stokes developed his own theory on non-linear oscillatory waves[71], which today is called "Stokes wave theory". Fundamentally, this theory is based on irrotational flow and non-hydrostatic pressure distributions, and is generally only appropriate for deep water waves - specifically for $H/h < 0.1$[69], $H/h < 1$, and $HL^2/h^3 \approx 1$ [74]. A second-order formulation, according to Dean[16], of the free surface for a Stokes wave is (using the above noted notation):

$$\eta(x,t) = \frac{H}{2}cos\left(\frac{2\pi}{L}x - \frac{2\pi}{T}t\right) + \frac{\pi H^2}{8L}\frac{cosh\left(\frac{2\pi}{L}h\right)}{sinh^3\left(\frac{2\pi}{L}h\right)}\left(2 + cosh\left(\frac{4\pi}{L}h\right)\right)cos\left(2\left(\frac{2\pi}{L}x - \frac{2\pi}{T}t\right)\right) \quad (1.12)$$

In 1895, Diederik Johannes Korteweg and Gustav de Vries published their work "*On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves*"[38]. Here they developed a non-linear wave theory that allows for periodic waves to exist in shallow-water[16]. Korteweg and de Vries coined their theory the "Cnoidal Wave Theory" due to their inclusion of a Jacobian elliptical function, cn.

The Korteweg and de Vries' Cnoidal Wave Theory is most useful for shallow water wave investigations - specifically when $HL^2/h^3 > 26$. But due to it's complexity, most authors choose not to use this theory, which this author has chosen to do as well, and the reader is directed to Wiegel's paper on "A presentation of condial wave theory for practical application"[83] for a comprehensive cnodial wave theory discussion.

Following the works of Airy, Boussinesq, Russell, Stokes, and Korteweg most research in the field of wave probation focused on refining these original works by increasing their order of accuracy[16, 69], or by evaluating when one theory breaks down and another takes over[16, 46]. For example, as briefly mentioned above, Stokes theory is appropriate for deep water conditions, while Cnoidal Waver Theory is better for shallow water conditions. However, the underlying principals remain the same.

From inspection of the linear wave, solitary wave, Stokes wave, and Cnoidal wave theories, it is apparent that these all assume the initial wave height is known, wave trains are infinity long and periodic, and water depths are constant. Sound judgment is required when applying them. As such, for SLGW research these theories can be problematic, namely because SLGW are highly non-linear, finite wave trains that may or may not be periodic, and they can rapidly transition from deep to intermediate to shallow depths (though most are shallow water) during their travels. Therefore, it is extremely difficult to soundly use an analytical theory without suitable expertise. Lastly, as discussed above (Section 1.1.1), predicting the exact wave height due to a subaerial landslide impact is not feasible.

1.1.3 Wave run-up research

Wave run-up is defined as the vertical height above the still water level an incident wave reaches when it collides with a boundary. For coastal waters, this could either be a typical wind-generated ocean wave breaking and running a meter or so up a beach, or a tsunami wave running $\geq$20-m[59] up a densely populated shoreline. For inland subaerial landslide generated waves (SLGW), this could be a wave

running up and overtopping a dam face.

Although the focus for this research is on inland SLGW, the physical concept of wave run-up is similar for coastal or inland waters. Rather it is dependent on the shape and roughness of the boundary the wave strikes, the depth of water and bed slope adjacent to the boundary toe, the permeability of the boundary, and the speed, length, and period of the incident wave[76]. As such, because of the breadth of variables, there is no complete analytical description of this phenomenon, and research is limited to empirical relations for linear monochromatic[6] wave run-up.

One of the first comprehensive studies on wave run-up was carried out by the US Army Corps of Engineers (USACE) from 1973-1984[75, 76]. Here the USACE presents graphical relationships between wave run-up on a variety of sloped surfaces (e.g. impermeable rigid flat/stepped/curved walls and permeable rubble slopes) and incident wave heights and periods, with different still water depths at the toe of a variably sloped boundary. The results indicated that the larger the wave height, the smaller the waver period, and/or the steeper the sloped boundary is, the higher the run-up will be. However, it is noted that this work is limited to linear monochromatic waves and does not account for complex sloped surfaces, non-linear effects, or 2D/3D waves.

During the same time as the USACE study, Chue[6] developed a "universal" wave run-up formula (Equation 1.13) that is useful for nonbreaking wave run-up. Here, Chue relates the run-up, R, to the still water depth, h, the incident wave period, L, and the angle of the sloped boundary the incident wave impacts, θ , as follows:

$$\frac{R}{h} = 1.8\left(1 - 3.111\frac{h}{L}\right)\left(\frac{\pi}{2\theta}\right)^{1/2} \tag{1.13}$$

Costas Synolokis[72], on the other hand, used a series of laboratory experiments to validate a simplified analytical solution he developed for non-breaking solitary wave run-up (Equation 1.14). Here, Synolokis relates wave run-up, R, to wave height, η, still water depth, h, and the angle of the sloped boundary the wave impacts, θ, as follows:

$$\frac{R}{h} = 2.831\sqrt{cot\theta}\left(\frac{\eta}{h}\right)^{\frac{5}{4}} \tag{1.14}$$

The simple form of Synolokis equation makes it attractive for general engineering applications; however, because it is only valid when $(\eta/h)^{\frac{1}{2}} \gg 0.288tan(\theta)$, and does not account for non-linear terms, its use is limited for complex situations.

To combat the need for estimating wave run-up on breaking waves, Hajime Mase[45] suggests relating wave run-up, R, to wave height, η, and wave period, L, as follows:

$$\frac{R}{\eta} = 2.32\left(\frac{tan\theta}{\sqrt{\eta/L}}\right)^{0.77} \tag{1.15}$$

However, Mase's equation is only valid for $2^\circ \leq \theta \leq 11^\circ$ and $0.007 \leq \eta/L$, meaning it is only valid for

[6] Waves having only one wave length (i.e. only one wave signal)

boundaries with gentle slopes and waves with heights that are relatively much smaller then their periods. Consequently, Mase's equation is not overly practical for SLGW, as these wave's have heights that are relatively much larger then their periods.

In fact, inspecting the equations developed by the USACE, Synolokis, and Mase, it is noted that non of these are valid for inland SLGW problems. Instead, only scaled physical models or validated computational fluid dynamics (CFD) models should be used.

1.1.4 Integrated subaerial landslide generated wave research

Although individual research into wave generation, propagation, and run-up phases has been fruitful, it is still necessary to look at subaerial landslide generated wave (SLGW) events as a whole, and integrate these three phases into one investigation. This is easily justified by simply looking at the difficulty associated with coupling the analytical/empirical relations between the three phases. For example, how is the maximum SLGW best described? By Kamphuis and Bowering[36], Noda[57], Wiegel et al.[84], Slingerland and Voight[68], Huber and Hager[34], Atai-Ashtiani and Nik-Khah's[2], or Heller[27, 28, 30]? Then, what is the best method to use for wave propagation? Is the wave a linear wave, a Stokes wave, a Cnoidal wave, a solitary wave, or an amalgamation of these four wave types? Lastly, once a wave hits a boundary, how can the maximum wave run-up be estimated? Here, likely the situation is such that the wave is non-linear and the boundary it is running up is much more complex then a gently slopped flat impermeable wall.

Therefore, because of the vast number of variables and alternative methods for evaluating each phase, an integrated approach using either scaled physical models (SPM) or computational fluid dynamics (CFD) models are used. The preferred approach is to use a SPM due to its ability to nearly reproduce prototype conditions. But due to their cost, both in terms of money and time, and scale effects, CFD models are increasingly being used.

However, the main issue with CFD models, is that they need to be properly validated before general application to SLGW work is accepted within the hydrotechnical engineering community. Where validation ultimately stems from the "exact" reproduction of real world SLGW events or high fidelity physical experimentation data, which is unfortunately rare to come by.

One of the first attempts to validate a CFD model for SLGW was carried out by Harbitz, Pedersen, and Gjevik[26]. They tried to reproduce a real world SLGW event (the Tafjord, Norway slide mentioned above) using their own CFD[7] model, but they under predicted the run-up by over -100%.

Liu et. al.[43] developed their own CFD model[8] and attempted to validate it against physical experiments they conducted. Here, they were specifically investigating the run-up generated behind the slide as it entered a body of water. The results of which reproduced the general trend of the temporal run-up, but did not accurately capture run-up magnitudes.

One of the first commercially available software packages used for SLGW validation was Flow-3D^{TM} by Basu[4]. Here a 2D Flow-3D^{TM} model was used to numerically re-create Fritz's Lituya bay

[7] linearized hydrostatic shallow water approximation discretized using finite differences

[8] Three-dimensional "filtered momentum and continuity" equations discretized using a finite volume scheme with a volume of fluid free surface tracker and LES turbulence model.

physical model experiments[20], which included a deformable slide material. To do this, Basu employed Flow-3D's^{TM} two-phase drift-flux model to represent a sediment-water slurry that he approximated as the slide[9]. The results of this simulation were then looked at from a wave generation point of view and a wave run-up point of view.

Basu's Flow-3D^{TM} model qualitatively reproduces the impact crater Fritz describes in his physical experiments, but tends to over predict the maximum wave run-up of the actual Lituya bay event by 28-56%. This, according to Basu, could be due to his choice of using a 2D approximation, and/or the physical approximation of the slide as a sediment-water slurry.

Using the same two-phase drift-flux model in Flow-3D^{TM}, Kim[37] also shows that the Flow-3D^{TM} over predicts the maximum wave run-up by 0-78% (depending on location). However, because of lack of physical model data, he could not parse out what was causing the discrepancy. That is, was it a wave generation, propagation, or run-up issue. Consequently, as with Basu's results, it is very difficult to say that Flow-3D^{TM} has been adequately validated for SLGW simulation.

Lastly, it is noted that there exists no quantitative 3D numerical simulation/validation of DualSPHysics for SLGW.

1.2 Aim of present work

The aim of this research is to conduct the first DualSPHysics simulations for three-dimensional inland subaerial landslide generated wave (SLGW) events, and to validate these simulations using high fidelity physical model data. In addition, this research aids in the continuing effort of validating the general purpose computational fluid dynamics model Flow-3D^{TM} for SLGW events.

1.3 Thesis outline

This thesis is has been broken into five chapters. Chapter 1 presents a comprehensive literature review of all relevant subaerial landslide generated wave research conducted to date. Chapter 2 discusses the numerical models used in this research, but mainly focuses on the mathematical backbone of the Smoothed Particle Hydrodynamics model called DualSPHysics. Chapter 3 goes over the physical experiment used to evaluate both Flow-3D^{TM} and DualSPHysics. Chapter 4 quantitatively compares the Flow-3D^{TM} and DualSPHysics simulations to the data generated by the physical experiment. Lastly, Chapter 5 completes this thesis with a detailed summary of this work, some concluding remarks, and a brief outlook for future numerical subaerial landslide generated wave research.

[9] Basu himself notes this as not being an exact representation of Fritz's slide, but it was/is the only option provided by Flow-3D^{TM} .

Chapter 2

Computational Methods

2.1 Background information

Describing subaerial landslide generated waves (SLGW) using a computational fluid dynamics (CFD) program is no trivial task. Not only does the CFD code have to account for moving (and possibly deforming) boundaries (i.e. the slide), but it also must capture the free surface, which for SLGW is a highly complex and multi-faceted problem. Wave generation during impact causes a turbulent crashing wave that is difficult to capture numerically, while wave run-up involves a rapid decrease in flow depth.

Fortunately, there are several general purpose/specialized CFD packages available that can perform SLGW simulations. Flow-3D^{TM} in particular has been shown to yield reasonable results[4, 14, 37, 56], which makes it a suitable tool to compare against new numerical schemes.

In the case of this research, a relatively new numerical scheme, Smoothed Particle Hydrodynamics, will be evaluated against both physical experimental data and Flow-3D^{TM} simulations of SLGW.

This section will provide a description of both Flow-3D^{TM} and DualSPHysics , and how their underling numerics make use of the governing equations of motion.

2.1.1 Governing equations

In the field of hydrotechnical computational fluid dynamics, two of the many relevant equations are the continuity and the Navier-Stokes equations. When coupled together, they describe the physical velocity and pressure fields of a fluid. These are collectively called the equations of motion. Table 2.1 presents them in their differential form for both Eulerian and Lagrangian descriptions.

Despite their compact form, these equations are wrought with mathematical richness and may therefore be discussed in great detail. Such a large discussion, however, is outside the context of this thesis and the reader is directed to comprehensive texts by Landau and Lifshitz[41], Tritton[73], Currie[12], and Kundu et. al[39, 40] for further information. That said, relevant details about these equations are discussed in the following sections.

Table 2.1: Differential form of continuity and Navier-Stokes equations

	Eulerian Form	Lagrangian Form
Continuity	$\frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} \rho = -\rho(\boldsymbol{\nabla} \cdot \mathbf{u})$	$\frac{D\rho}{Dt} = -\rho(\boldsymbol{\nabla} \cdot \mathbf{u})$
Navier-Stokes	$\rho(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} \mathbf{u}) = -\boldsymbol{\nabla} P + \rho \mathbf{g} + \boldsymbol{\Gamma}$	$\rho(\frac{D\mathbf{u}}{Dt}) = -\boldsymbol{\nabla} P + \rho \mathbf{g} + \boldsymbol{\Gamma}$

Note: ρ=density; t=time; $\boldsymbol{p}$=pressure; $\boldsymbol{u}$=velocity vector; g=gravity; $\boldsymbol{\Gamma}$=dissipative terms

2.2 Flow-3D^{TM}

Flow-3D^{TM} is a commercially available software produced by FlowScience. The program was develop by Dr. C.W. Hirt in 1980 after he left the Los Alamos National Laboratory. The method uses a rectilinear mesh to discretize a computational domain using the Finite Volume Method, and solves the Eulerian three-dimensional Navier-Stokes and Continuity equations.

At the core of the Flow-3D^{TM} program is what FlowScience has coined the "True Volume of Fluid", a modified version of Hirts original construct[32]. This allows for efficient tracking of complex free-surface's and the reproduction of solid boundary interfaces. As such, many consulting engineering firms and research teams use Flow-3D^{TM} to investigate hydrotechncial problems (eg. spillway performance, hydraulic jumps, weir design, etc.).

The intent of this section is not to re-construct the Finite Volume Method or present the underlying numerics of Flow-3D^{TM}[1], but rather to qualitatively focus on some of the relevant features that are specific to this research and give the reader a feel for how Eulerian CFD works. Specifically, this section will focus on spatial discretization, free surface tracking, and momentum advection and viscosity.

Spatial discretization: In the field of CFD, the computational domain can be spatial discretized using either a unstructured or structured mesh. The mesh is a discrete set of connected nodes that construct thousands to hundreds of millions of tetrahedral or hexahedral cells used to solve the equations of motion.

Unstructured meshes, are a random construct of cells that range in both size and shape and have no direct connectivity[2] - this connectivity comes from clever bookkeeping. A structured mesh, on the other hand, uses a logical construct to generate cells that can be either uniform in size, or non-uniform; however, the general shape remains constant throughout the domain.

Both meshes are fixed in space - and hence called a Eularian frame of reference. Computations are subsequently evaluated at fixed points using a Finite Volume method.

In general terms, the Finite Volume method uses a flux integral to approximate the rate of "things" (i.e. mass, momentum, velocity, etc.) travelling through a given volume (i.e. a cell). A more detailed description can be found in Versteeg[78].

[1] Which can be difficult when working with proprietary algorithms

[2] Cell one is not necessary connected to cell two, etc..

Free surface tracking: Of particular interest to hydrotechnical engineers is the time evolution of the free surface. Unfortunately, this is rather difficult to do in Eularian CFD and requires special treatment. Flow-3D^{TM} uses it's proprietary Volume of Fluid method to track the free surface. It does this by finding the volume of water occupied in each cell and prescribing the free surface as the cell with the smallest fractional volume that is above zero.

The Volume of Fluid method has been shown to be quite robust and computationally efficient. However, it does lack the ability to completely track complex non-linear free surfaces, like for example a wave.

Momentum advection and viscosity: Viscosity is implemented in Flow-3D^{TM} through the Navier-Stokes equations. However, additional dissipation (i.e. numerical viscosity) is encountered when momentum is advected between cells. Here, momentum is carried from one cell to another and the result is a combination of the current and previous values. This process smooths out momentum gradients between cells, and when performed over many steps a diffusion of momentum is observed[31]

Flow-3D^{TM} uses a first-order momentum advection scheme as the default. However, when their second-order scheme is used the simulation is less dissipative and performs well for free surface waves[18].

Turbulence: Flow-3D^{TM} allows a user to select from one of the following five turbulence models[18]:

1. Prandtl mixing length model: Turbulence is taken into account by spatially "enhancing" the kinematic viscosity in such a way viscous losses resemble TKE dissipation.

2. One-equation turbulence model: In this model, the kinetic energy associated with turbulence (i.e. turbulent kinetic energy - TKE) is approximated as the half the summation of the squared component velocity fluctuations (u', v', and w')

$$TKE = 0.5(u'^2 + v'^2 + w'^2) \tag{2.1}$$

   Equation 2.1 is then dispersed using Flow-3D's^{TM} "FAVORized" equations of motion, and the rate of TKE dissipation is approximated as:

$$\varepsilon = 0.09\sqrt{\frac{3}{2}}\left(\frac{TKE^{1.5}}{0.07 * L_{min}}\right) \tag{2.2}$$

   where L_{min} is the smallest domain dimension.

3. Two-equation (k-ε) turbulence model: The two-equation (k-ε) model is essentially a more sophisticated extension of the one-equation model. Instead of computing the rate of TKE dissipation based on a purely empirical relationship, it is instead computed using a empirical-numerical (i.e. using Flow-3D's^{TM} "FAVORized" equations of motion with some empirical coefficients) approach, but is limited to a minimum value defined by Equation 2.2.

4. Renormalization-Group (RNG) model: "The RNG model uses equations similar to the equations for the k-ε model. However, equation constants that are found empirically in the standard k-ε mode are derived explicitly."

5. Large Eddy Simulation (LES) model: The LES method models all turbulent flow structures that can be resolved by the numerical mesh and only approximates features that are smaller than the grid.

From inspection, the simplest of methods is the Prandtl method. Here it is noted that it is rather simple to implement and requires the least amount of CPU resources. That said, one must use cation when using this method, as it is only really suitable for fully developed steady flows[18].

The one-, two-equation and RNG models are all derivatives of the same fundamental assumptions; however, they vary in sophistication. The one-equation being the least, while the RNG being the most sophisticated. As such, they also each influence the time to complete a simulation by varying degrees. Where the-one equation is the quickest, while the RNG model is the slowest.

Finally the LES model is considered the most advance and accurate model of them all. However, it is noted that not only is it the slowest, it is also plagued by numerical instability issues that are highly dependent on the users mesh building skills.

From a practical point of view, the most widely used model is the RNG model. It yields a suitable balance been both simulation accuracy and practical simulation run times.

Computational hardware: With this research residing in the field of CFD, some description of the hardware Flow-3D^{TM} uses is needed.

For the standard[3] version of Flow-3D^{TM} , the hardware used is simply a computer's central processing unit (CPU). The CPU may include several cores that can be used for computations. Intuitively, the speed at which a Flow-3D^{TM} simulation comes to completion would linearly increase with each additional CPU core; however, this is not the case, as the more cores used for a simulation, the more overhead (in the form of communication) between cores is required. In fact, it is observed that Flow-3D^{TM} simulation speed plateaus at 16-cores. The significance of this may not initially be visible, but when factored in with decreased rates of CPU speed growth (i.e. the industry no longer follows Moore's Law), one is forced to ask: has the speed of CFD modelling reached its limit with traditional CPUs?

2.3 Smoothed particle hydrodynamics

Smoothed particle hydrodynamics is a mathematical construct that approximates spatial derivatives without the need for a computational mesh. It does this by prescribing a Lagrangian description of motion to a discrete set of randomly spaced computational points (or particles) and interpolating between neighbouring particles using an appropriate function.

This, however, is a vastly different approach from traditional CFD constructs; as such, the following sections outline the fundamental mathematics of SPH and will focus on how DualSPHysics[11, 24, 25]

[3]There is also a parallel version of Flow-3D^{TM} that allows simulations to be solved on a Linux cluster

takes advantage of this construct.

2.3.1 Integral representation

At the core of SPH, lies the integral interpolation method (or smoothing kernel, or kernel). Here a continuous field function, f, that depends on a three-dimensional position vector, $\mathbf{x}$, within a given domain, Ω, that is bounded by the surface, $\partial\Omega$, (see Figure 2.1) can be exactly represented as the convolution product with the Dirac delta function, δ:

$$f(\mathbf{x}) = \int f(\boldsymbol{x}')\delta(\boldsymbol{x}\text{-}\boldsymbol{x}')d\mathbf{x}' \tag{2.3}$$

where $\boldsymbol{x}'$ is a neighbouring position vector, $d\boldsymbol{x}'$ is an elemental volume that is located within Ω, and $\delta(\boldsymbol{x}\text{-}\boldsymbol{x}')$ is given by:

$$\delta(\boldsymbol{x}\text{-}\boldsymbol{x}') = \begin{cases} 1 & for \quad \boldsymbol{x} = \boldsymbol{x}' \\ 0 & for \quad \boldsymbol{x} \neq \boldsymbol{x}' \end{cases} \tag{2.4}$$

However, Equation 2.3 only holds for continuous functions as the Dirac delta function is infinitely narrow. To accommodate discrete functions, the Dirac delta function can be replaced with a suitable approximation, say $W(\boldsymbol{x}\text{-}\boldsymbol{x}', h)$, that is of a finite width, h. In the context of SPH, such functions are typically called kernels (or smoothing kernel) and are discussed below. Equation 2.3 can be written as follows (note, angular brackets, $\langle\rangle$, are used to define the approximated representation of f):

$$f(\boldsymbol{x}) \approx \langle f(\boldsymbol{x})\rangle = \int f(\boldsymbol{x}')W(\boldsymbol{x}\text{-}\boldsymbol{x}', h)d\mathbf{x}' \tag{2.5}$$

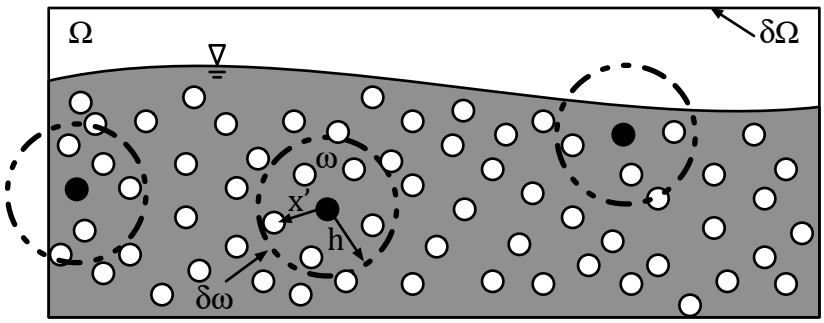

Figure 2.1: Smoothed particle hydrodynamics domain with three types of kernel support. The centre kernel has complete support. The right kernel has a truncated support, but is within the domain. And the left kernel has truncated that extends past the main domain.

Smoothing kernels

The kernel function is a "distribution" function that is evaluated using neighbouring particles. The closer the particle is, the more influence it has on the the kernel function. Theoretically, this influence tends to zero as the distances between two particles approach infinity. However, this is impractical for numerical schemes. As such, the kernel function is forced to be zero at a prescribed distance h, called the smoothing length.

The smoothing length defines the radial distance away from a particle that other particles will influence that particle. In three-dimensions, this would be a volumetric construct, and is typically called the kernel support, ω, that is bounded by the surface $\partial\omega$. The larger the smoothing length, the more particles will influence a given particle and yield a more accurate results. However, with more particles supporting a kernel more computational time is required. Oger[58] suggests using a normalized smoothing length (i.e. $\frac{h}{\Delta x}$, where Δx is the particle size) of 3 to limit errors in spatial derivatives, but admits this yields excess computational time. As such, he indicates a normalized value of 1.2-1.5 is an acceptable balance between time and error.

Kernel functions can theoretically be any distribution function; however, the following caveats limit what kernel functions can be used[42]:

1. The kernel must be normalized for convergence[17]: $\int_\omega W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = 1$
2. The kernel must approach the Dirac delta function as the smoothing length tends to zero, or else the integral interplant will not converge[17] : $\lim_{h\to 0} W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = \delta(\boldsymbol{x}\text{-}\boldsymbol{x}')$
3. The kernel must be symmetric, so that two particles an equal distance away will have the same influence: $W(-(\boldsymbol{x}\text{-}\boldsymbol{x}'),h) = W(\boldsymbol{x}\text{-}\boldsymbol{x}',h)$
4. The kernel must have compact support: $W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = 0$ for $|\mathbf{x}\text{-}\mathbf{x}'|>$h

With this in mind, the so called "golden" kernel is a Gaussian kernel. It is stable and accurate for higher order spatial derivatives[42]. However, a Gaussian kernel does not have compact support because it asymptotes to zero with increasing x. Consequently, several alternative kernel functions have been developed. For example,

- Modified Gaussian: $W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = \frac{1}{\pi^{3/2}h^3} exp(-q^2)$
- Quadratic: $W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = \frac{5}{4\pi h^3}[\frac{3}{16}q^2 - \frac{3}{4}q + \frac{3}{4}]$ for $\leq q \leq 2$
- Cubic spline[54]: $W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = \begin{cases} 1-\frac{3}{2}q^2+\frac{3}{4}q^3 & for \quad 0\leq q\leq 1 \\ \frac{1}{4}(2-q)^3 & for \quad 1\leq q\leq 2 \\ 0 & for \quad q\geq 2 \end{cases}$
- Wendland[82]: $W(\boldsymbol{x}\text{-}\boldsymbol{x}',h) = \frac{21}{16}(1-\frac{q}{2})^4(2q+1)$ for $0 \leq q \leq 2$

(note: q is the non-dimensional distance between particles)

Generally speaking, any of these can be implemented in SPH; however, in DualSPHysics only the Cubic spline and Wendland kernels are implemented, and of these, the Cubic spline is the most widely used since it most closely resembles a Gaussian distribution. That said, numerical stability[42] and particle clumping[63] issues have been noted. The Wendland kernel, is said to be more accurate than the cubic spline, but increases computational cost[25].

2.3.2 Derivative representation

Using Equation 2.5, a spatial derivative, $\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$, can now be written as:

$$\left\langle \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right\rangle = \int_{\omega} \frac{\partial f(\mathbf{x}')}{\partial \mathbf{x}} W(\boldsymbol{x}\text{-}\boldsymbol{x}', h) d\mathbf{x}' \tag{2.6}$$

Applying the product rule, Equation 2.6 is re-cast as:

$$\left\langle \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right\rangle = \int_{\omega} \frac{\partial}{\partial \mathbf{x}} [f(\mathbf{x}') W(\boldsymbol{x}\text{-}\boldsymbol{x}', h) d\mathbf{x}'] - \int_{\omega} f(\mathbf{x}') \frac{\partial W(\boldsymbol{x}\text{-}\boldsymbol{x}', h)}{\partial \mathbf{x}'} d\mathbf{x}' \tag{2.7}$$

If f is a vector or scalar field, then applying Gauss's theorem[39, 66] to the first term on the right-hand side of Equation 2.7 yields:

$$\left\langle \frac{\partial f(\mathbf{x})}{\partial x} \right\rangle = \int_{\partial \omega} f(\boldsymbol{x}') W(\boldsymbol{x}\text{-}\boldsymbol{x}', h) \cdot \mathbf{n} dS - \int_{\omega} f(\mathbf{x}') \frac{\partial W(\boldsymbol{x}\text{-}\boldsymbol{x}', h)}{\partial x'} d\mathbf{x}' \tag{2.8}$$

Equation 2.8 is the complete SPH expression to represent a spatial derivative in either a vector or scalar field, respectively. However, to enforce point 4 above (i.e. the kernel must have compact support), the first term on the right-hand side in Equation 2.8 must be zero. Therefore, a spatial derivative in either a vector or scalar field can be approximated as:

$$\left\langle \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right\rangle = -\int_{\omega} f(\mathbf{x}') \frac{\partial W(\boldsymbol{x}\text{-}\boldsymbol{x}', h)}{\partial \mathbf{x}'} d\mathbf{x}' \tag{2.9}$$

For the case where the kernel has a truncated support (i.e. at a boundary), the approximation of the spatial derivative falls apart. That said, special techniques (as discussed in Section 2.3.7) can be used to rectify this problem.

Equation 2.9 now allows the spatial derivative of a function to be determined using values of the function and spatial derivatives of the smoothing kernel, rather than from spatial derivatives of the function itself[42]. This approach will now be used to transform the differential from of the equations of motion (see Table 2.1) into the discrete Smoothed Particle Hydrodynamics approximation of the equations of motion.

2.3.3 Particle approximation of continuity

Referring back to the SPH integral representation of a functions derivative (Equation 2.9), when the kernel is symmetric and the elemental volume, $d\mathbf{x}'$, is replaced by the mass of a particle over its density

(i.e. a discrete volumetric construct), it can be re-written in discrete form as:

$$\left\langle \frac{\partial f(\mathbf{x}_\mathrm{a})}{\partial \mathbf{x}_\mathrm{a}} \right\rangle \approx -\sum_b \frac{m_\mathrm{b}}{\rho_\mathrm{b}} f(\mathbf{x_b}) \frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b, h)}{\partial \mathbf{x}_\mathrm{b}} \tag{2.10}$$

where b denotes the particle label up to the number of particles in support of particle a; and m_b, ρ_b, and $\boldsymbol{x}_\mathrm{b}$ are the mass, density, and position of particle b, respectively.

However, it is noted that this derivative does not vanish if $f(\mathbf{x})$ is a constant. To rectify this, Monaghan[52] suggests using the product rule of differentiation as follows:

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \frac{1}{\Phi}\left[\frac{\partial(\Phi f(\mathbf{x}))}{\partial \mathbf{x}} - f(\mathbf{x})\frac{\partial \Phi}{\partial \mathbf{x}}\right] \tag{2.11}$$

where Φ is any differentiable function. Re-casting in SPH form with $\Phi = \rho$, we get the following:

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \frac{1}{\rho_\mathrm{a}}\left[-\sum_b \frac{m_\mathrm{b}}{\rho_\mathrm{b}} \rho_\mathrm{b} f(\mathbf{x_b}) \frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b, h)}{\partial \mathbf{x}_\mathrm{b}} - f(\mathbf{x_a}) \frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b, h)}{\partial \mathbf{x}_\mathrm{b}}\right] \tag{2.12}$$

or, if re-arranged and simplified

$$\frac{\partial f(\mathbf{x}_\mathrm{a})}{\partial \mathbf{x}} = -\frac{1}{\rho_\mathrm{a}} \sum_b \left[m_\mathrm{b} (f(\mathbf{x}_\mathrm{b}) - f(\mathbf{x}_\mathrm{a})) \frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b, h)}{\partial \mathbf{x}_\mathrm{b}} \right] \tag{2.13}$$

Equation 2.13 now vanishes when $f(\mathbf{x})$ is a constant.

Now, replacing $\boldsymbol{\nabla} \cdot \mathbf{u}$ with Equation 2.13 in the Lagrangian description of continuity (Table 2.1), the discrete SPH continuity equation can be constructed as:

$$\frac{D\rho_\mathrm{a}}{Dt} = -\sum_b m_\mathrm{b} (\mathbf{u}_\mathrm{a} - \mathbf{u}_\mathrm{b}) \frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b, h)}{\partial \mathbf{x_b}} \tag{2.14}$$

Equation 2.14 is how DualSPHysics generally exploits the continuity equation using SPH; however, alternative codes may set Φ, in Equation 2.11, equal to one[52] or some other differentiable function.

2.3.4 Particle approximation of Navier-Stokes

In Lagrangian form, the Navier-Stokes equation can be written as (Table 2.1)

$$\rho\left(\frac{D\mathbf{u}}{Dt}\right) = -\boldsymbol{\nabla} P + \rho \mathbf{g} + \boldsymbol{\Gamma} \tag{2.15}$$

If an inviscid fluid is assumed, Equation 2.15 can be written as:

$$\rho\left(\frac{D\mathbf{u}}{Dt}\right) = -\boldsymbol{\nabla} P + \rho \mathbf{g} \tag{2.16}$$

Using the discrete SPH equation for first derivatives (Equation 2.13), Equation 2.16 can now be expressed as:

$$\rho_{\mathrm{a}}\left(\frac{D\mathbf{u}_{\mathrm{a}}}{Dt}\right)=\sum_{b}\frac{m_{\mathrm{b}}}{\rho_{\mathrm{a}}}(P_{\mathrm{b}}-P_{\mathrm{a}})\frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)}{\partial \mathbf{x_b}}+\rho\mathbf{g} \tag{2.17}$$

However, this does not conserve neither linear or angular momentum since the the force on particle a from b is not the same as the force on b from a (i.e. $P_{\mathrm{a}} \neq P_{\mathrm{b}}$). Instead, Monaghan[50] suggests to symmetrize the pressure gradient term as follows:

$$\frac{\boldsymbol{\nabla} P}{\rho}=\boldsymbol{\nabla}\left(\frac{P}{\rho}\right)+\frac{P}{\rho^2}\boldsymbol{\nabla}\rho \tag{2.18}$$

Casting Equation 2.18 into SPH form (using Equation 2.10), we get the following:

$$\frac{\boldsymbol{\nabla} P}{\rho}=-\sum_{b}\frac{m_{\mathrm{b}}}{\rho_{\mathrm{b}}}\left(\frac{P_{\mathrm{b}}}{\rho_{\mathrm{b}}}\right)\frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)}{\partial \mathbf{x}_{\mathrm{b}}}+\frac{P_{\mathrm{a}}}{{\rho_{\mathrm{a}}}^2}\left[-\sum_{b}\frac{m_{\mathrm{b}}}{\rho_{\mathrm{b}}}\rho_{\mathrm{b}}\frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)}{\partial \mathbf{x}_{\mathrm{b}}}\right] \tag{2.19}$$

Simplifying Equation 2.19 and placing it into into Equation 2.16, we get:

$$\frac{D\mathbf{u}_{\mathrm{a}}}{Dt}=\sum_{b}m_{\mathrm{b}}\left(\frac{P_{\mathrm{a}}}{{\rho_{\mathrm{a}}}^2}+\frac{P_{\mathrm{b}}}{{\rho_{\mathrm{b}}}^2}\right)\frac{\partial W(\mathbf{x}_{\mathrm{a}}-\mathbf{x}_{\mathrm{b}},h)}{\partial \mathbf{x_b}}+\mathbf{g} \tag{2.20}$$

Equation 2.20 is now spatially discretized for an inviscid fluid. To include viscous effects, Monaghan[50] suggests using an artificial viscosity term, Π, in Equation 2.20:

$$\frac{D\mathbf{u}_{\mathrm{a}}}{Dt}=\sum_{b}m_{\mathrm{b}}\left(\frac{P_{\mathrm{a}}}{{\rho_{\mathrm{a}}}^2}+\frac{P_{\mathrm{b}}}{{\rho_{\mathrm{b}}}^2}+\Pi_{\mathrm{ab}}\right)\frac{\partial W(\mathbf{x}_{\mathrm{a}}-\mathbf{x}_{\mathrm{b}},h)}{\partial \mathbf{x_b}}+\mathbf{g} \tag{2.21}$$

Where Π_{ab} is equivalent to the dissipative term, Γ in Table 2.1 and can be any suitable construct. However, in DualSPHysics, Π_{ab} is defined as:

$$\Pi_{\mathrm{ab}}=\begin{cases}-\alpha\frac{0.5(C_{\mathrm{a}}-C_{\mathrm{a}})h(\mathbf{u}_{\mathrm{a}}-\mathbf{u}_{\mathrm{b}})\cdot(\mathbf{x}_{\mathrm{a}}-\mathbf{x}_{\mathrm{b}})}{(\rho_{\mathrm{a}}-\rho_{\mathrm{b}})[(\mathbf{x}_{\mathrm{a}}-\mathbf{x}_{\mathrm{b}})^2+0.01h^2]} & for \quad (\mathbf{u}_{\mathrm{a}}-\mathbf{u}_{\mathrm{b}})\cdot(\mathbf{x}_{\mathrm{a}}-\mathbf{x}_{\mathrm{b}})<0\\ 0 & for \quad (\mathbf{u}_{\mathrm{a}}-\mathbf{u}_{\mathrm{b}})\cdot(\mathbf{x}_{\mathrm{a}}-\mathbf{x}_{\mathrm{b}})>0\end{cases} \tag{2.22}$$

Where α is the coefficient of viscosity; u is the particle velocity; x is the particle position vector; ρ is the particle density; and C is the particle speed of sound.

From Equation 2.22 it is noted that α and h are user defined variables, where α has the most influence on the viscous term.

2.3.5 Moving the particles

As the SPH method is a Lagrangian description, the computational points must move with the fluid. This can be achieved by simply saying $\frac{d\mathbf{x}_{\mathrm{a}}}{dt}=\mathbf{u}_{\mathrm{a}}$ or by Monaghan's[49] suggestion of an "XSPH variant":

$$\frac{d\mathbf{x}_{\mathrm{a}}}{dt}=\mathbf{u}_{\mathrm{a}}+\varepsilon\sum_{b}m_{\mathrm{b}}\left(\frac{\mathbf{u}_{\mathrm{b}}-\mathbf{u}_{\mathrm{a}}}{0.5(\rho_{\mathrm{a}}+\rho_{\mathrm{b}})}\right)W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h) \tag{2.23}$$

In this case, the XSPH variant moves a particle with the average velocity of its neighbours.

DualSPHysics uses both methods, where ε is a definable parameter ranging from zero to one.

2.3.6 Equation of state

With three equations (Equation 2.14, Equation 2.21, Equation 2.23) and four unknowns (velocity vector, position vector, pressure, and density) there is a closure problem.

Analytically, this closure problem can be solved by assuming that water is an incompressible fluid, and the continuity equation reduces to $\nabla \cdot \mathbf{u} = 0$ and the pressure can be expressed as a non-linear function of velocity that can be solved using a traditional Poisson solver. However, this can become rather complex to solve numerically, so a "weakly compressible" assumption can be made. This allows for an efficient closure scheme.

DualSPHysics uses the following equation of state to relate pressure to density:

$$P(\rho) = \frac{\rho_o {c_o}^2}{7}\left[\left(\frac{\rho}{\rho_o}\right)^7 - 1\right] \tag{2.24}$$

Where ρ_o is the reference density (usually 1000kg/m^3 for water) and c_o is the speed of sound.

Of particular note, Equation 2.24 indicates that a small change in density, ρ, will change the pressure. For weakly compressible fluids, this change does not occur instantaneously. Rather, it takes time for a density change to effect pressure elsewhere in the SPH domain. The time it takes to make this change is governed by the speed of sound. The faster the speed of sound, the "stiffer" the fluid, and the more dissipative the fluid will be[63].

In DualSPHysics c_o is defined as the "coefficient of sound" multiplied by the maximum particle velocity. The significance of which was just discussed, but has other implications with time integration (Section 2.3.8).

Furthermore, it is well known in the SPH community that pressure oscillations about a particle can occur. This may not be significant for certain cases where pressure or force terms are not explicitly needed, but it can cause significant heartache when coupled with a structural solver. Molteni[48] suggest smoothing out these pressure oscillations using what he calls a "Delta-SPH" term. Essentially, this term filters out high frequency density oscilations prior to solving for pressure. This research did not require the use of the Delta-SPH term.

2.3.7 Boundary conditions

Until now, it has been noted that the smoothing kernel requires full support. However, when a particle is near a boundary, the support becomes truncated and the kernel falls apart. For astrophysical computations where the problem has no boundaries, this poses no problem. But in hydrotechnical engineering problems boundaries are extremely important. As such, the kernel support is required to be supplemented so that it is no longer truncated.

Several attempts have been made to implement suitable boundary conditions. Monaghan[51] used a set of particles along a boundary that used a repulsive force to keep fluid particles at bay (analogous to repulsive forces encountered between molecules). Randles[62] completed the kernel support for particles that encroached on a boundary by taking the mirror image of truncated support along the boundary line - this they called ghost particles. Crespo[10] on the other hand, used two layers of staggered fixed

particles to implement what he called a "dynamic boundary particle". All three boundary conditions have their pros and cons and can be discussed in detail; however, as DualSPHysics exclusively uses dynamic boundary particles, this research will focus on them.

Physically, dynamic particles work by generating a density gradient along a boundary, which in turn generates a pressure gradient that creates a repulsion mechanism along the boundary interface.

Numerically, dynamics boundary particles are implemented by using an alternative SPH momentum equation:

$$\frac{d\mathbf{u}_\mathrm{a}}{dt} = -\left(2c^2 \frac{W(\mathbf{x}_\mathrm{a}-\mathbf{x}_\mathrm{b},h)}{(W(\mathbf{x}_\mathrm{a}-\mathbf{x}_\mathrm{b},h)+W(\mathbf{x}_\mathrm{a}-\mathbf{x}_\mathrm{a},h))^2} + m_\mathrm{b}\Pi_\mathrm{ab}\right)\frac{\partial W(\mathbf{x}_\mathrm{a}-\mathbf{x}_\mathrm{b},h)}{\partial \mathbf{x_a}} + \mathbf{g} \tag{2.25}$$

It is noted that boundary particles are easy to implement "since they can be calculated inside the same loop as fluid particles" and hence "save considerable computational time"[10]. But like most things, this easy of implementation comes at a cost. It has been noted by Crespo and observed in this research that using this method can cause an unphysical repulsion mechanism along a boundary with a thin layer of fluid. As such, a gap between the flow and the boundary can be seen. In the case of this research, this causes a moving boundary to be artificially larger in volume (see Section 4.2.2).

For the case where the kernel support is truncated due to a free surface, there is no cause for alarm. This is actually where SPH excels, because a lack of support at the free surface simply means there are no particles influencing that particle, which is a natural way of tracking the free surface. Consequently, SPH is well poised for tracking complex non-linear free surfaces, and why many coastal hydrodynamic researchers have used it[7, 13, 23, 64, 70].

In addition to dynamic boundary particles, DualSPHysics can also prescribe periodic boundaries. That is, should the kernel support of a particle extend past a boundary, it can complete its support using particles from the opposite boundary. For example, when simulating flow through a rectangular flume, the outflow particles would support the inflow particles.

2.3.8 Time integration

With the Navier-Stokes equations reduced to a set of ordinary differential equations, any stable time integration scheme may be applied[52]. In DualSPHysics , the two available schemes are the Verlet and Symplectic schemes[25]:

Verlet scheme: The Verlet scheme is a second-order integrator, that Verlet[77] discusses in detail; however, in essence, it uses data from the previous time (i.e. time equals n-1) and the current time (i.e. time equals n) to find the density, ρ, velocity, u, and position of a particle at the next time step (i.e. time equal n+1):

$$\left\{\begin{array}{ll} (a) & \rho_a^{n+1} = \rho_a^{n-1} + 2\Delta t\left(\sum_b \frac{m_\mathrm{b}}{\rho_\mathrm{b}}(\mathbf{u}_\mathrm{a}-\mathbf{u}_\mathrm{b})\frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)}{\partial \mathbf{x_a}}\right)^n \\ (b) & \mathbf{u}_a^{n+1} = \mathbf{u}_a^{n-1} + 2\Delta t\left(-\sum_b m_\mathrm{b}\left(\frac{P_\mathrm{a}}{{\rho_\mathrm{a}}^2}+\frac{P_\mathrm{b}}{{\rho_\mathrm{b}}^2}+\Pi_\mathrm{ab}\right)\frac{\partial W(\mathbf{x}_\mathrm{a}-\mathbf{x}_\mathrm{b},h)}{\partial \mathbf{x_a}}+\mathbf{g}\right)^n \\ (c) & \mathbf{x}_a^{n+1} = \mathbf{x}_a^n + \Delta t\mathbf{u}_a^n + 0.5\Delta t^2\left(\mathbf{u}_\mathrm{a}+\varepsilon\sum_b m_\mathrm{b}\left(\frac{\mathbf{u}_\mathrm{b}-\mathbf{u}_\mathrm{a}}{0.5(\rho_\mathrm{a}+\rho_\mathrm{b})}\right)W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)\right)^n \end{array}\right\} \tag{2.26}$$

However, because these equations are not coupled, this scheme tends to diverge if every 50th time step is not updated in the following manner[25]:

$$\left\{\begin{array}{ll} (a) & \rho_a^{n+1} = \rho_a^{n} + \Delta t\left(\sum_b \frac{m_\mathrm{b}}{\rho_\mathrm{b}}(\mathbf{u}_\mathrm{a}-\mathbf{u}_\mathrm{b})\frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)}{\partial \mathbf{x_a}}\right)^n \\ (b) & \mathbf{u}_a^{n+1} = \mathbf{u}_a^{n} + \Delta t\left(-\sum_b m_\mathrm{b}\left(\frac{P_\mathrm{a}}{{\rho_\mathrm{a}}^2}+\frac{P_\mathrm{b}}{{\rho_\mathrm{b}}^2}+\Pi_\mathrm{ab}\right)\frac{\partial W(\mathbf{x}_\mathrm{a}-\mathbf{x}_\mathrm{b},h)}{\partial \mathbf{x_a}}+\mathbf{g}\right)^n \\ (c) & \mathbf{x}_a^{n+1} = \mathbf{x}_a^n + \Delta t\mathbf{u}_a^n + 0.5\Delta t^2\left(\mathbf{u}_\mathrm{a}+\varepsilon\sum_b m_\mathrm{b}\left(\frac{\mathbf{u}_\mathrm{b}-\mathbf{u}_\mathrm{a}}{0.5(\rho_\mathrm{a}+\rho_\mathrm{b})}\right)W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)\right)^n \end{array}\right\} \tag{2.27}$$

Symplectic scheme: The symplectic scheme is a second-order accurate integrator that is a slightly more complex scheme than the Verlet scheme. Particle positions (Equation 2.28a) and densities (Equation 2.28b) are computed mid-step. After which, momentum (Equation 2.29a) and particle positions are updated at end of the step (Equation 2.29b) using mid-step values, while the $\frac{d\rho^{n+1}}{dt}$ is updated using $\mathbf{u}^{n+1}$ and $\mathbf{x}^{n+1}$[52].

$$\left\{\begin{array}{ll} (a) & \rho_a^{n+1/2} = \rho_a^{n} + \frac{\Delta t}{2}\left(\sum_b \frac{m_\mathrm{b}}{\rho_\mathrm{b}}(\mathbf{u}_\mathrm{a}-\mathbf{u}_\mathrm{b})\frac{\partial W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)}{\partial \mathbf{x_a}}\right) \\ (b) & \mathbf{x}_a^{n+1/2} = \mathbf{x}_a^n + \frac{\Delta t}{2}\left(\mathbf{u}_\mathrm{a}+\varepsilon\sum_b m_\mathrm{b}\left(\frac{\mathbf{u}_\mathrm{b}-\mathbf{u}_\mathrm{a}}{0.5(\rho_\mathrm{a}+\rho_\mathrm{b})}\right)W(\boldsymbol{x}_a\text{-}\boldsymbol{x}_b,h)\right) \end{array}\right\} \tag{2.28}$$

$$\left\{\begin{array}{ll} (a) & \mathbf{x}_a^{n+1} = \mathbf{x}_a^{n+1/2} + \frac{\Delta t}{2}\mathbf{u}^{n+1} \\ (b) & (V\rho\mathbf{u})_a^{n+1} = (V\rho\mathbf{u})_a^{n+1/2} + \frac{\Delta t}{2}(\frac{d(V\rho\mathbf{u})}{dt})_a^{n+1} \end{array}\right\} \tag{2.29}$$

Where V is the volume of a particle.

Both schemes are dependent on variables already discussed, but they are also dependent on an incremental time step, Δt. In DualSPHysics, the time step is evaluated using the minimum time step computed by either a forcing condition (so that particles do not get too close to each other over a time-step) or a combined viscous diffusion and Courant-Friedrichs-Lewy[8] (CFL) condition[49]:

$$\Delta t = min \begin{cases} min[0.3(\sqrt{h/|f_a|})] \\ \\ min\left[\left(\frac{0.3h}{c_a + max\left|\frac{h((\mathbf{u}_a-\mathbf{u}_b)\cdot(\mathbf{x}_a-\mathbf{x}_b))}{(\mathbf{x}_a-\mathbf{x}_b)^2}\right|}\right)\right] \end{cases} \tag{2.30}$$

Where f_a is the force per unit mass on a particle, c_a is the speed of sound evaluated for each particle, u is a particle's velocity, and x is a particle's position.

From inspection of the second constraint in Equation 2.30, it is noted that the dominating variable is the speed of sound. The smoothing length is O(0.01), the difference in velocity between particles is assumed O(1), the difference in particle position is related to particle size, O(0.001-10), and the speed of sound in water is O(1000). As such, Δt will be impractically small for numerical simulations, so an approximation to this is to assume the speed of sound is at least ten times the maximum velocity. This assumption limits the density fluctuations on DualSPHysics to within 1%. For example, if a particle's pressure is proportional to a reference density multiplied by the square of a particle's velocity ($p \sim \rho_o u_a^2$), then Equation 2.24 can be re-written as:

$$\frac{\rho_a}{\rho_o} = \left(\frac{7u_a^2}{c_o^2} + 1\right)^{1/7} \approx 1 + \frac{u_a^2}{c_o^2} + O\left(\left[\frac{7u_a^2}{c_o^2}\right]^2\right) \tag{2.31}$$

therefore,

$$\frac{\Delta\rho_a}{\rho_o} \sim \frac{u_a^2}{c_o^2} \tag{2.32}$$

and if $c_o = 10u_a$, Equation 2.32 goes to:

$$\frac{\Delta\rho_a}{\rho_o} \sim \frac{u_a^2}{100u_a^2} = 1\% \tag{2.33}$$

Equation 2.33 validates approximating the speed of sound as ten time the particle velocity.

2.3.9 Computer hardware

Instead of using CPUs to execute CFD code in serial or parallel with several CPU cores, DualSPHysics uses graphical processing units (GPUs)[11], to compute things in parallel using hundreds of low cost GPU cores.

For example, simulations performed in this thesis were carried out using a NVIDIA^{TM} Tesla-C2075 GPU with 448 cores, where the decrease in simulation run times were orders of magnitude lower when compared with a serial execution.

Unfortunately, not all CFD codes are well suited for GPU execution[4]. Hence, not many commercial codes use GPUs. That said, because of the open question stated earlier (i.e. has the speed of CFD modelling reached its limit with traditional CPUs), software developers are now trying to utilize GPUs for cheaper and faster simulations.

[4]This is mainly due to how matrices are solved

Chapter 3

Physical Experiment

3.1 Model setup

The physical experiment, Figure 3.1, was constructed and tested at Northwest Hydraulic Consultants, North Vancouver, Canada laboratory. It consisted of a 0.61-m wide flume with a 30-deg sloped west wall, a 45-deg slopped east wall, two vertical walls on the north and south sides of the model, and a 1.02-m horizontal section connecting the toe of both the east and west slopes.

Subaerial land slide generated waves (SLGW) were created by releasing a 0.18x0.31x0.31-m weighted acrylic box with a 45-deg sloped nose down the west slope of the model. When released, the slide travelled (by gravity) 0.77-m before impacting the 0.61-m deep water; after which, it travelled another 1.05-m, through the water, before hitting a stop block.

Using National Instruments (NI) LabVIEW^{TM} 2012 with two NI USB-6008 four channel 12-bit data acquisition boards, slide acceleration and displacement, as well as wave heights were recorded for a period of six-seconds at a frequency of 100Hz. The six instruments used to measure this data are as follow:

Slide acceleration: The acceleration of the slide as it travelled down the west slope was measured using a Memsic CXL04GP1 single axis accelerometer. Manufacture specifications are outlined in table Table 3.1. The overall accuracy, E, of this device can be inferred as $E = \sqrt{e_1 + e_2 + ... + e_6} = 3.2\%$, where e is the individual error of each component in the accelerometer.

Slide displacement: A Celesco SG1-120-3 potentiometer was used to measure the slide displacement over time as it travelled down the west slope. Accuracy of this device is stated by the manufacture as 0.01-m over 3.05-m. Or $\pm$ 0.35% full scale (FS).

Wave probes 1, 2, & 3: RBR WG-50 capacitance wave-probes, which work by measuring the change in capacitance of the probe as the water level changes with time. They have an accuracy (stated by the manufacture) of $\pm$ 0.4%, and react instantaneously.

The probes were installed vertically in the model so that half their length was located within the water, while the other half was outside the water (still water conditions). Output data consisted of temporal

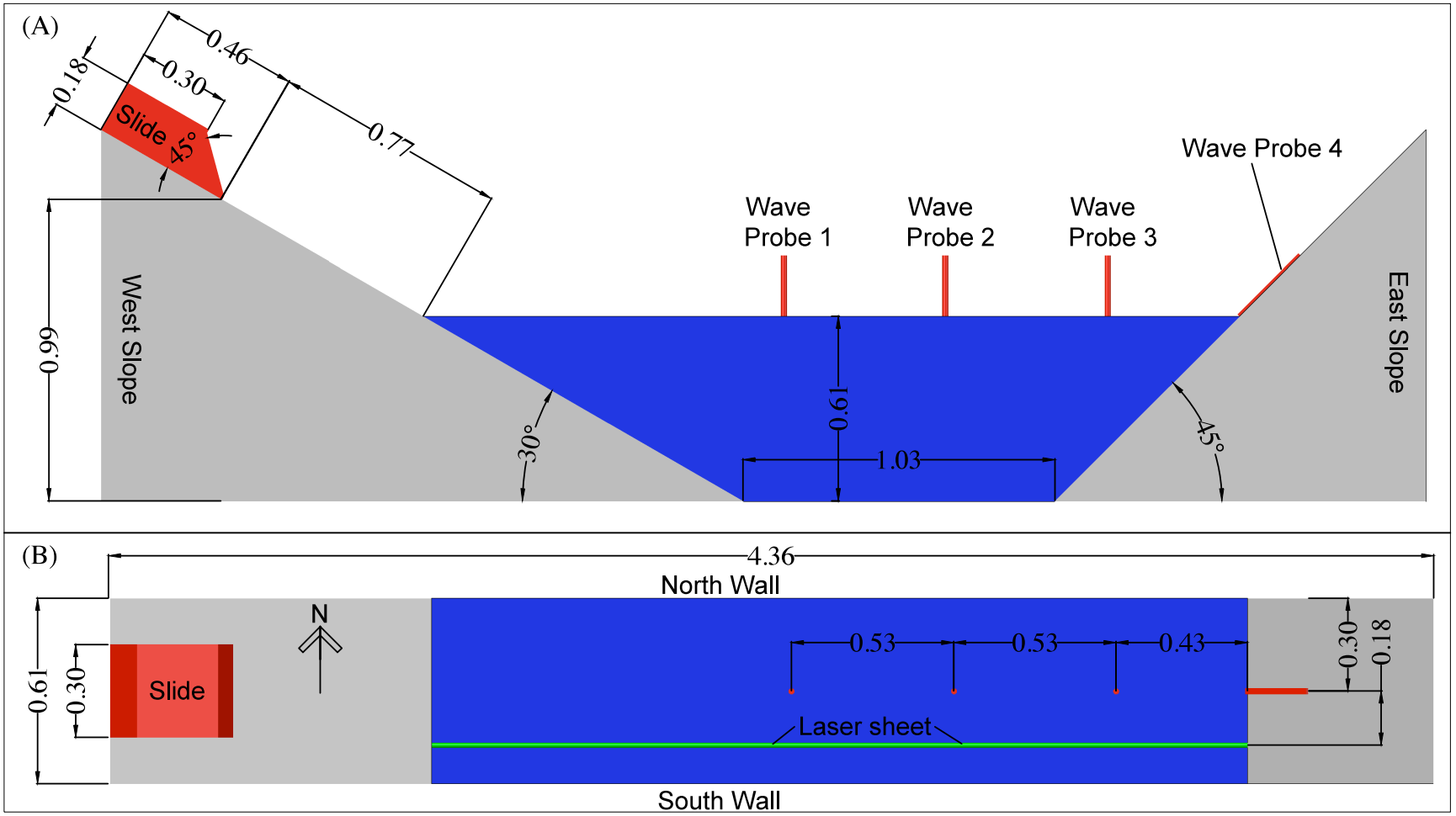

Figure 3.1: Subaerial Landslide Generated Wave Test Stand. Elevation view shown in panel (a) and plan view shown in panel (b). (Note: all units are in meters)

Table 3.1: Accelerometer technical specifications

Specification	Value	Units
Input Range (e_1)	$\pm$ 4	G†
Zero Drift (e_2)	$\pm$ 0.1	G†
Sensitivity (e_3)	500 $\pm$ 15	mV/G†
Transverse Sensitivity (e_4)	$\pm$ 5	% of Sensitivity
Non-Linearity (e_5)	$\pm$ 0.2	% FS‡
Alignment Error (e_6)	$\pm$ 2	Degrees

† Multiples of Gravity; ‡ Full Scale

voltages that were converted to wave heights above the still water level.

Wave probe 4: A Milone PN-6573TC-24 continuous fluid level probe was used. It reacts instantaneously and has a resolution of $\pm$ 0.25-mm with an accuracy of 0.006-m over its 0.607-m length ($\leq 1\%$ FS). The probe laid flat along the east slope with half of its length residing in the water, and the other half outside the water. Output data consisted of temporal voltages that were converted to wave heights above the still water level.

In addition to the above mentioned instrumentation, three Allied Manta G-125 video cameras with Theia MY110M machine vision lenses (i.e. no optical distortion) were used to record the water's free surface for each test. Theoretically, this setup could capture 30 frames per second; however, in reality, it only achieved 15-18 frames per second.

During each test run, the model would operate in darkness while a thin sheet of laser light was cast along a plane offset 0.18-m from the centreline of the flume. This, combined with using luminescent dye, allowed for enhanced visualization of the water's free surface.

3.2 Model results

The physical experiment, discussed above, consisted of releasing the slide from a single fixed location on the west ramp and recording data for a period of 6-sec. Experimental random error was reduced by repeating this five times.

The data recovered from all six instruments during each of the five test runs can be seen in Figure 3.2. Here, it is observed that this data is highly repeatable. The maximum coefficient of variation (standard deviation over the mean value) is 0.33, 0.02, 3.81, 1.02, 0.21, and 0.16 for the slide acceleration, slide displacement, wave probe 1, wave probe 2, wave probe 3, and wave probe 4, respectively.

From Figure 3.2, it is found that after an instantaneous release, the slide travelled down the west slope with an average acceleration of 3.8m/s^2 until it impacted the water, thereafter decelerating at a average rate of -3.5m/s^2.

The maximum observed generated wave was the wave caused by the slide upon initial impact with the water. This wave travelled the length of the flume and rode up the east wall to a height of 0.08-

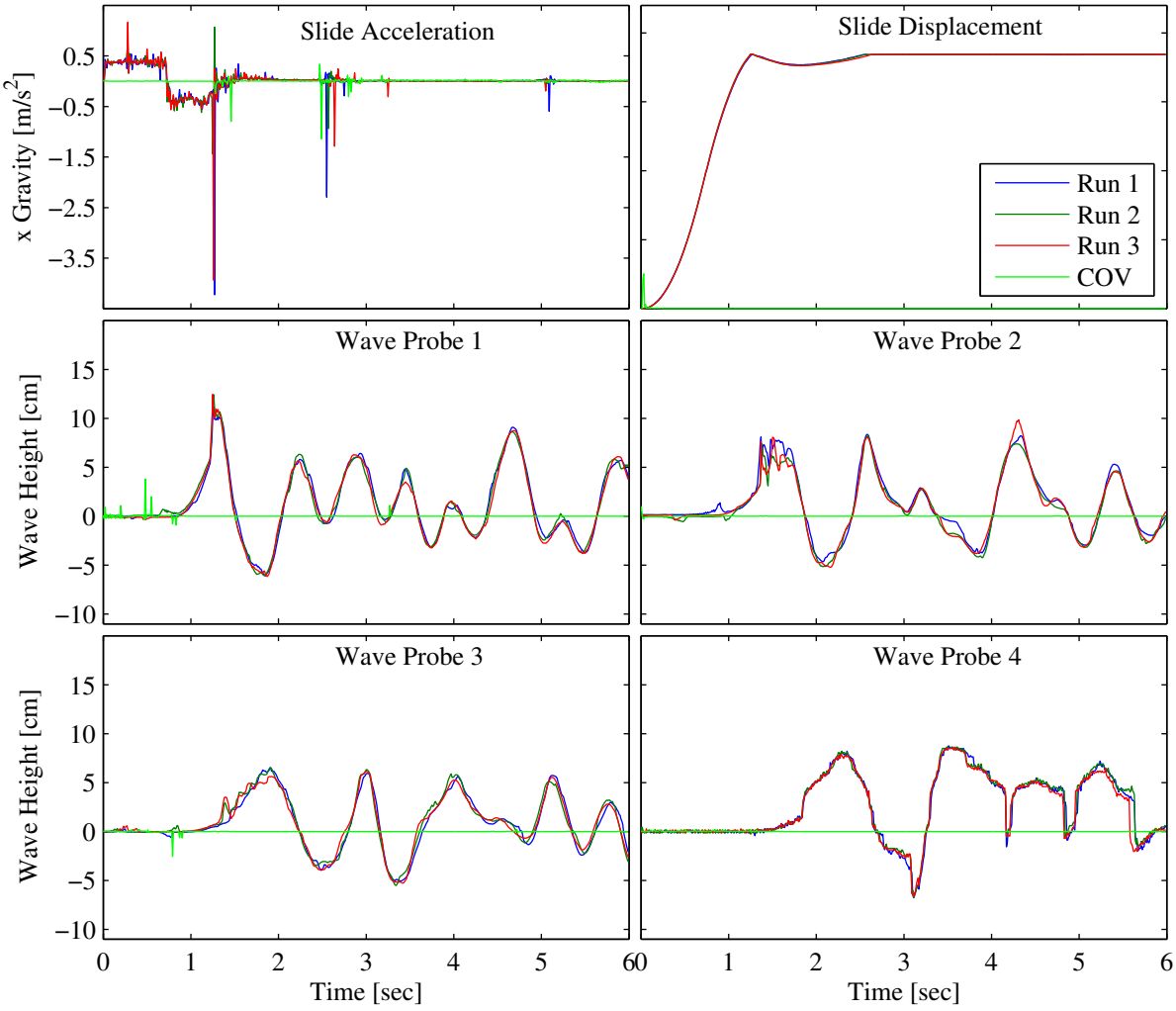

Figure 3.2: Results of the five physical experiment test runs. Note, slide displacement y-axis ticks are at 0 0.5 1.0 1.5 2.0-meters.

m above the still water elevation. However, this was not the observed maximum wave run-up. This occurred during the next run-up cycle, when the wave reached 0.086-m above the still water elevation. At this point, the wave running up the east wall was not simply a result of east-west reflection of the first wave, but also a combination of additional waves being generated post impact behind the slide, and north-south wave reflections. After the slide came to a rest, the waves in the flume gradually dissipated due to boundary friction, and viscosity.

3.2.1 Slide motion

Although slide acceleration and displacement were recorded in the physical experiment, they are not useful input parameters for the numerical models. Both Flow-3D^{TM} and DualSPHysics require velocity profiles for prescription of motion. As such, the acceleration data was integrated and the displacement

data was differentiated to get two separate velocity profiles for all five tests.

Figure 3.3 shows the ensemble average of the slide velocity using both acceleration and displacement data during a six second period. Here, it is noted that between time = 0 and time = 1.25-sec the computed slide motion using either the potentiometer or accelerometer data yields similar results (root-mean-square-difference for this period is 0.05-m/s). However, after time=1.25-sec the slide motion computed using the potentiometer and accelerometer differs considerably. The potentiometer data indicates that at 1.25-sec, the slide rapidly changes from travelling down the west slope (positive velocity) to travelling up the west slope (negative velocity) for a brief period of time before again travelling down the west slope and ultimately stopping. This data matches visual observations, as it was noted the slide did not completely stop when it hit a stop block (located at the toe of the slide), but rather bounced off the stop block once before coming to a complete stop. The accelerometer data, however, does not tell the same story. It suggests something occurred at time=1.25-sec and the slide velocity changes by first decreasing, then increasing before stabilizing at a constant velocity of approximately 0.75-m/s. This of course is not physical, and suggests something was wrong with the data after 1.25-sec. As such, the slide motion was taken as the average of the computed motion from the potentiometer and accelerometer data sets between time = 0 to time = 1.25-sec, and then set to 0-m/s, thereafter. This motion was then used as an input file for numerical simulations and classified as a 'simple' description of slide motion. Subsequently, the entire potentiometer data set was used as a 'complex' slide description of motion.

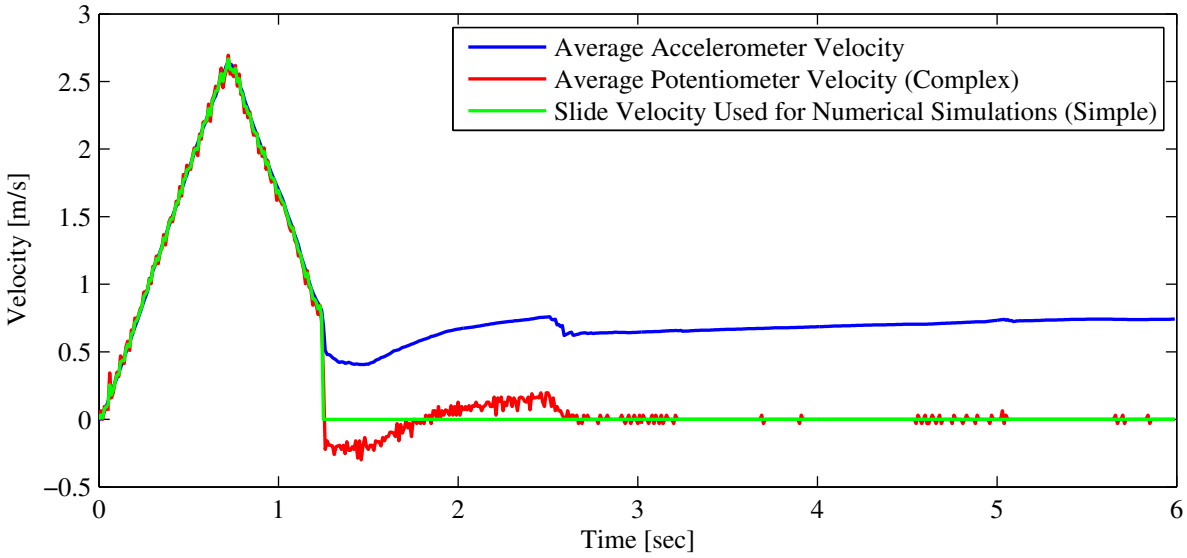

Figure 3.3: Slide motion computed from the average of five accelerometer and potentiometer data sets. A simple slide motion used for numerical simulations is the average between the average accelerometer and average potentiometer data from Time=0 to Time=1.25 sec, and then set to zero thereafter. A complex slide motion was also used, and is defined as the entire potentiometer data set.

Chapter 4

Numerical Experiment

Regardless of the robustness and maturity of a numerical scheme, there will always be a need to evaluate the scheme's default settings. This is especially true for general purpose schemes where the default settings are the appropriate settings for the majority of scenarios, not necessarily for the desired one.

As such, the physical data discussed earlier (see Chapter 3) is used to validate both the Flow-3D^{TM} and DualSPHysics models using their respective default settings, thereafter referred to baseline modelling. Subsequently, the physical model data is used to calibrate both models through a series of developmental modelling until an optimal set of parameters is found.

In this chapter, the sections on baseline modelling, developmental modelling, and final results represent the typical path of a simulation sequence used to find the optimal settings for a given scenario (in this case, aerial landslide generated waves). These sections have further been split into two subsections to accommodate both Flow-3D^{TM} and DualSPHysics models.

4.1 Baseline modelling

Baseline modelling consisted of running both Flow-3D^{TM} and DualSPHysics models with their default settings, and comparing their output with the physical experiment data described in the previous chapter (see Chapter 3). For both numerical models this is described in detail below.

4.1.1 Eulerian computational fluid dynamics

Using the same dimensions as the physical model, a computerized three-dimensional model was constructed and imported into Flow-3D^{TM} as a sterolithography (STL) file. Once in Flow-3D^{TM} , 5.8 million 0.010-m sized square cells in a 4.39(L)x0.60(W)x2.20(H)-m rectangular mesh were used to discretize the 3D-model.

The slide was prescribed a simple velocity profile based on data collected from the physical model (as discussed above, Section 3.2.1). Surface roughness for the east and west sloped faces was set to 0-m (i.e. hydraulically smooth). The modelled fluid was water at 293k, and a dynamic Renormalization-Group (RNG) turbulence model with the default settings were used in Flow-3D^{TM} v10.1[18].

Similar to the physical model, Flow-3D^{TM} simulated a period of six seconds and recorded wave

heights at wave probe locations 1-3 (Figure 4.1). For wave probe 4, Flow-3D^{TM} does not explicitly compute the temporal wave run-up heights. Instead, it outputs a text file containing a longitudinal centreline slice of the free surface. This output however, is not the free surface between fluid and air, but rather between fluid/solid and air. Flow-3D^{TM} does not differentiate between fluid or solid when computing the free surface. As such, two different post processing schemes were used to capture the wave run-up height.

1. Threshold operator: Using a rotated frame of reference that aligns with the east wall (i.e. the east wall is the new zero reference line), the first free surface elevation above a given threshold (0.00001-m) was considered the location at which the free surface intersected the east wall.

2. Maximum second-derivative: Here, the maximum second-derivative of the free surface profile near the east wall is considered the location at which the free surface intersected the east wall.

Both methods appear to yield reasonable results (Figure 4.1), but they also have their issues. The threshold operator tends to work very well when fluid is travelling up the east wall; but when travelling down, this method under predicts the free surface due to significant noise in the data. The maximum second-derivative generally works better then the threshold operator method, but its output is noisy due to sudden changes in the location of maximum second-derivative. For example, at one time step the maximum second-derivative may be at the free surface/east wall interface, but then it may jump to a random section where the slope of the free surface is quite large. To alleviate these issues, results from both methods are presented.

Frames of the Flow-3D^{TM} simulation can be seen in Figure 4.2 and Figure 4.3. Here, the slide can be seen impacting the fluid at about 0.8-sec and generating the initial impact wave. Figure 4.4b shows that during impact, Flow-3D^{TM} does not capture the splash nor the non-linear shape of this impact wave. As such, Flow-3D^{TM} under predicts the impact wave height (at time equal to about 1.2sec) by 36%, Figure 4.1a.

At 1.0sec, the fluid starts to wrap around behind the slide before hitting itself at 1.2-sec (Figure 4.3). After this, the initial impact wave is observed to travel from west to east (Figure 4.4c) until time equal to about 2.20-sec, where it runs 0.092-m (15% higher than the physical experiment) up the east slope before reflecting back. During which, the secondary wave generated behind the slide at 1.2-sec is observed to travel west to east and reflect off the north and south walls, which can seen in Figure 4.3.

After the initial impact wave reflects off the east slope, it starts to interact with the wave generated behind the slide and the waves reflected off the north and south walls before running back up the east slope to a height of 0.172-m at 3.8-sec (Figure 4.2 and Figure 4.4d), or 96% higher than the physical experiment.

Following the maximum wave run-up, subsequent run-ups were again over-predicted and Flow-3D^{TM} wave probe 1-3 (Figure 4.1) data became less accurate. Table 4.1 presents the associated error with the baseline testing. The relative maximum wave amplitude error is the relative error between the physical and Flow-3D^{TM} data pertaining to the maximum amplitude in each wave probe data set.

Based on the above noted observations, a series of developmental tests were carried out to try and reduce the error between Flow-3D^{TM} and the physical experiment - specifically the maximum wave run-up. The description of these tests and their results can be found in Section 4.2.1.

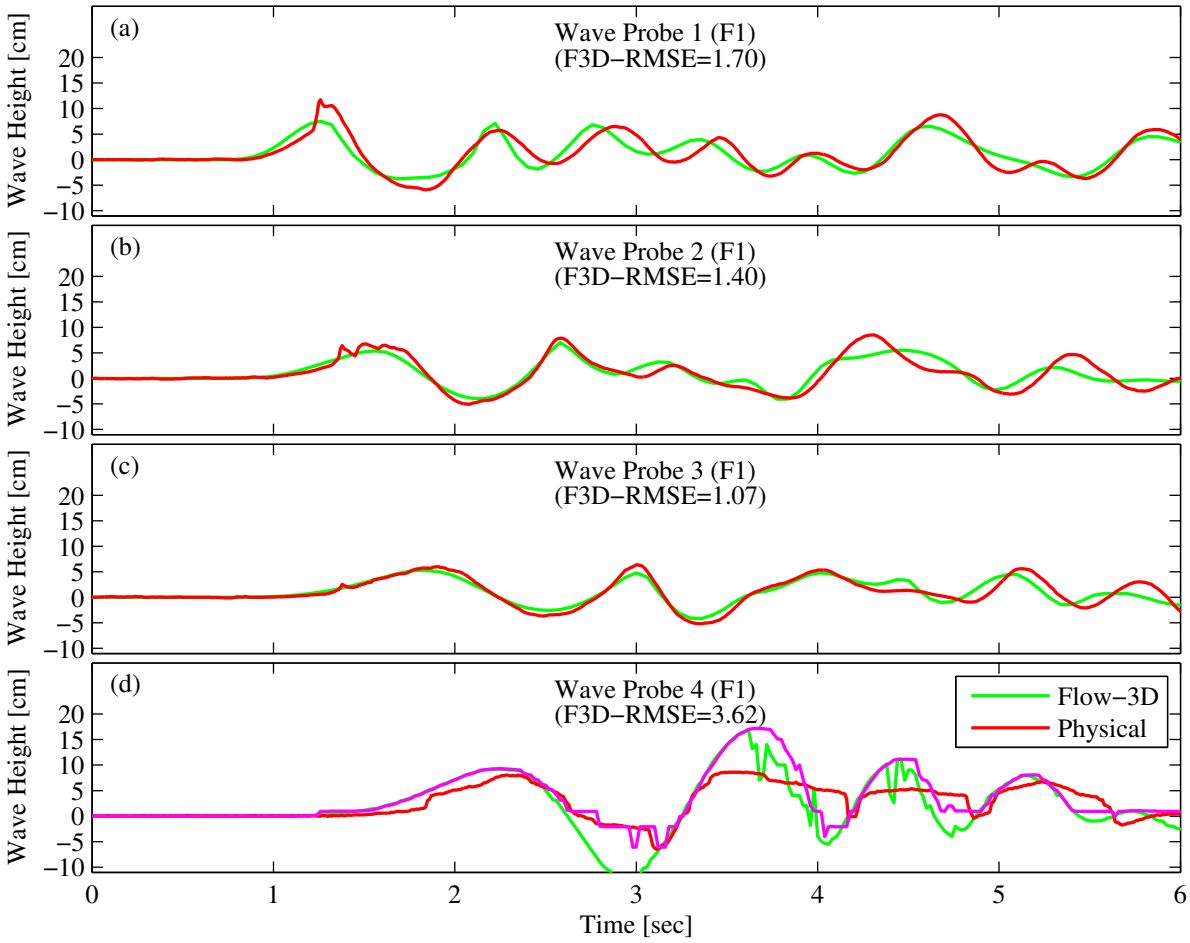

Figure 4.1: Baseline Flow-3D^{TM} and physical model data of the free surface measured at wave probe 1-4 locations. Panel d represents the evaluated wave run-up using the maximum second derivative method (green line) and the threshold operator method (magenta line).

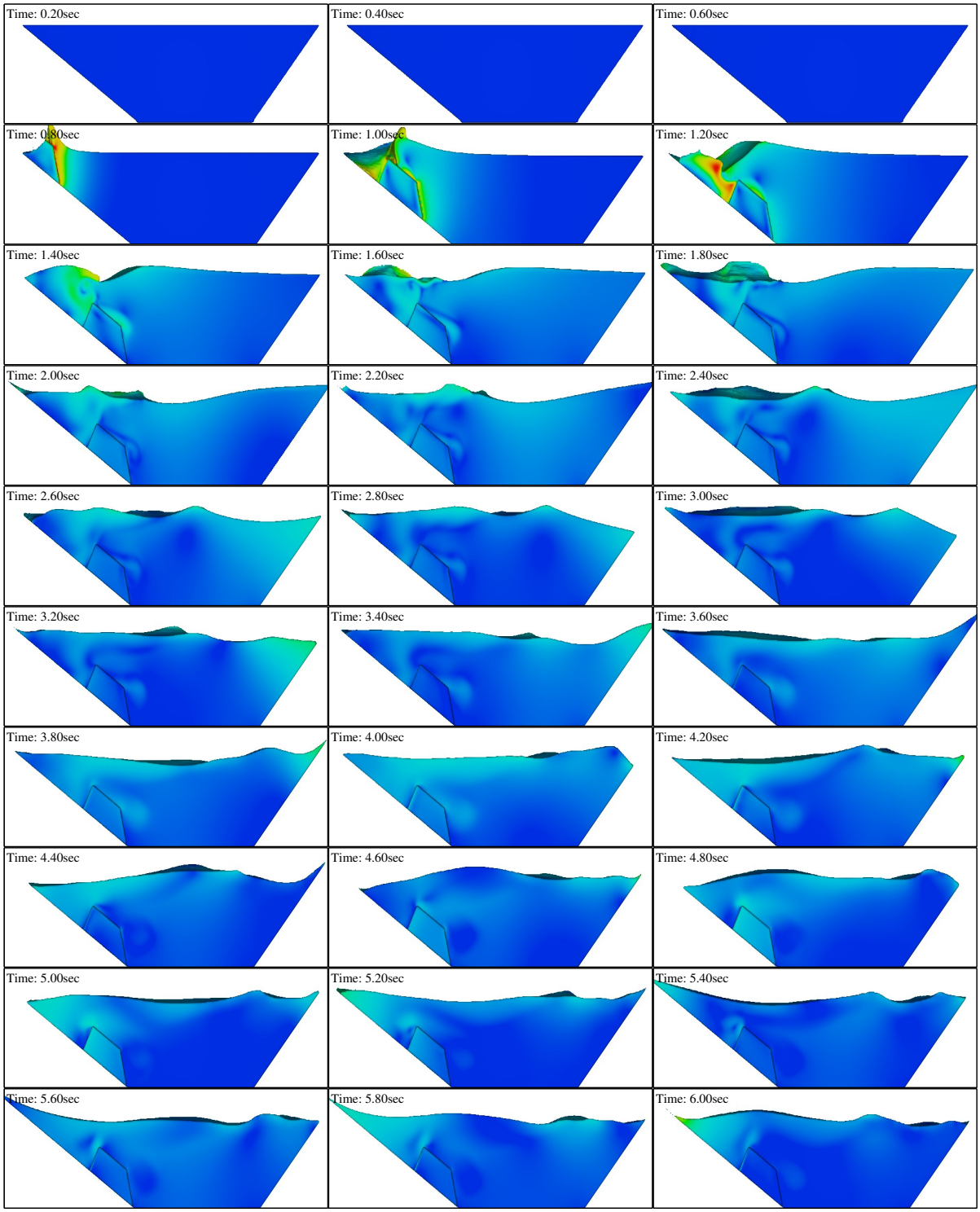

Figure 4.2: Baseline elevation view of Flow-3D^{TM} frames (Test- #F1). Note, colour gradient represent velocity magnitude, where blue is 'slow' and red is 'fast'.

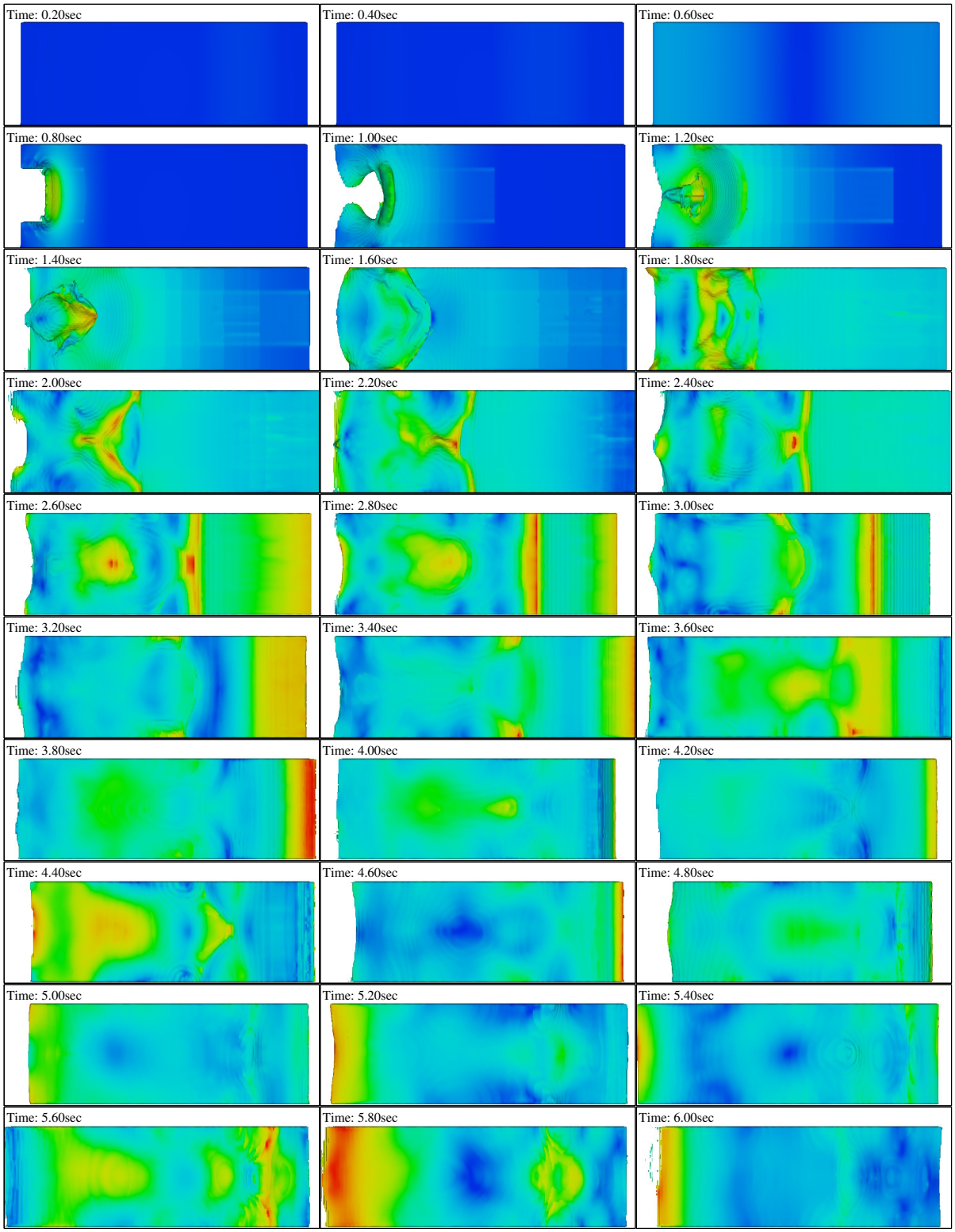

Figure 4.3: Baseline plan view of F3 Flow-3D^{TM} frames (Test- #F1)

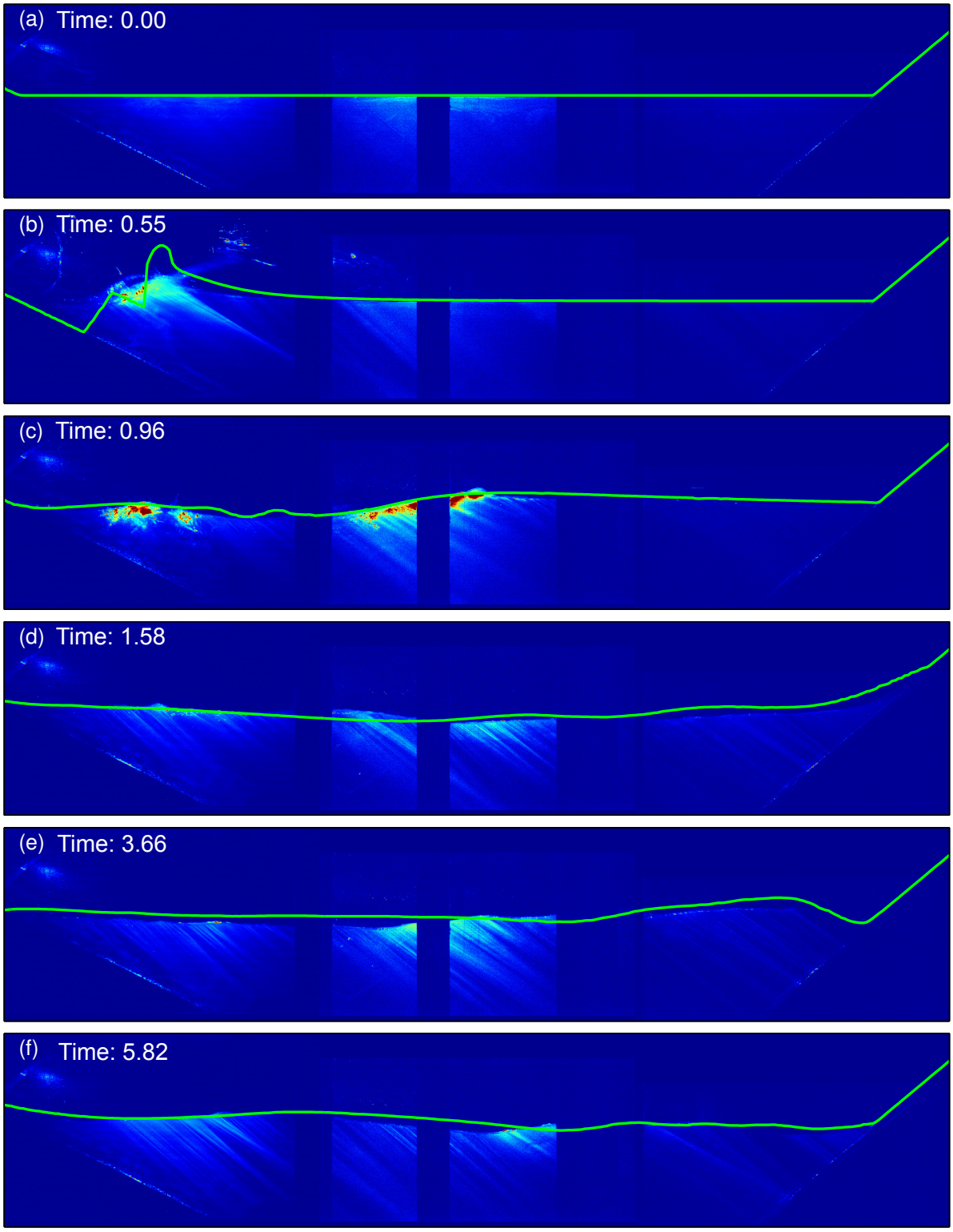

Figure 4.4: Baseline Flow-3D^{TM} centreline free surface overlaid onto physical experiment images. Panels a, b, c, d, e, and f represent time at 0.00, 0.55, 0.96, 1.58, 3.66, and 5.82-sec, respectively.

Table 4.1: Baseline Flow-3D^{TM} error

Wave probe	Relative max. wave amplitude error	Root mean square error
WS1	-36%	1.70-cm
WS2	-19%	1.40-cm
WS3	-18%	1.07-cm
WS4a†	96%	3.62-cm
WS4b‡	100%	2.82-cm

† Maximum second derivative; ‡ Threshold operator

4.1.2 Smoothed particle hydrodynamics

A copy of the STL file used in the Flow-3D^{TM} model was imported into DualSPHysics and the domain was discretized into 700,000 boundary particles and 1.3 million fluid particles, for a total of 2.0 million 0.010-m sized particles.

The slide was prescribed a simple velocity profile based off data collected from the physical model (as discussed above, Section 3.2.1), and default settings presented in Table 4.2 were used to initiate a simulation:

Table 4.2: Baseline DualSPHysics default settings

Parameter	Value
CFL	0.2
Coefficient of sound	10
Smoothing length coefficient	0.866025
Reference density	1000kg/m^3
Time stepping	verlet
Kernel	Cubic spline
Viscosity coefficient	0.24

As with the physical experiment and Flow-3D^{TM}, DualSPHysics simulated a period of six seconds, during which it computed the free surface height at the wave probe 1-4[1] locations at a frequency of 50Hz. The data from this test can be seen in Figure 4.5. Frames of the DualSPHysics simulation can be seen in Figure 4.6 and Figure 4.7. Here, like the Flow-3D^{TM} simulation, the slide can be seen impacting (Figure 4.8b) the fluid at 0.8-sec and generating an initial impact wave that is 1% higher then the physical experiment. At 1.0-sec, the fluid starts to wrap around behind the slide and then hits its self at 1.2sec (Figure 4.7). Then, the initial impact wave is observed to travel from west to east (Figure 4.8c), but its motion is 'sluggish' after running up 0.12-m (or 50% more than the physical experiment) up the east wall (Figure 4.8d), where the wave dissipates quickly (Figure 4.8e and f).

Comparing Figure 4.6 and Figure 4.7 against Figure 4.2 and Figure 4.3 shows DualSPHysics is more dissipative than Flow-3D^{TM} . Table 4.3 quantifiably presents the associated error with the baseline testing. Furthermore, looking at the width of the impact crater 1.2-sec for both the DualSPHysics and Flow-3D^{TM} simulations suggests that the DualSPHysics slide is much wider. In fact, when measured it is 13.25% wider than the Flow-3D^{TM} impact crater. Also, in Flow-3D^{TM} the initial impact wave appeared radial, whereas DualSPHysics it clearly is not radial.

[1] Unlike Flow-3D^{TM} , DualSPHysics is well-poised to capture maximum wave run-up heights. This is primarily due to its ability to explicitly distinguish between the fluid and solids. As such, post-processing of the output data was trivial.

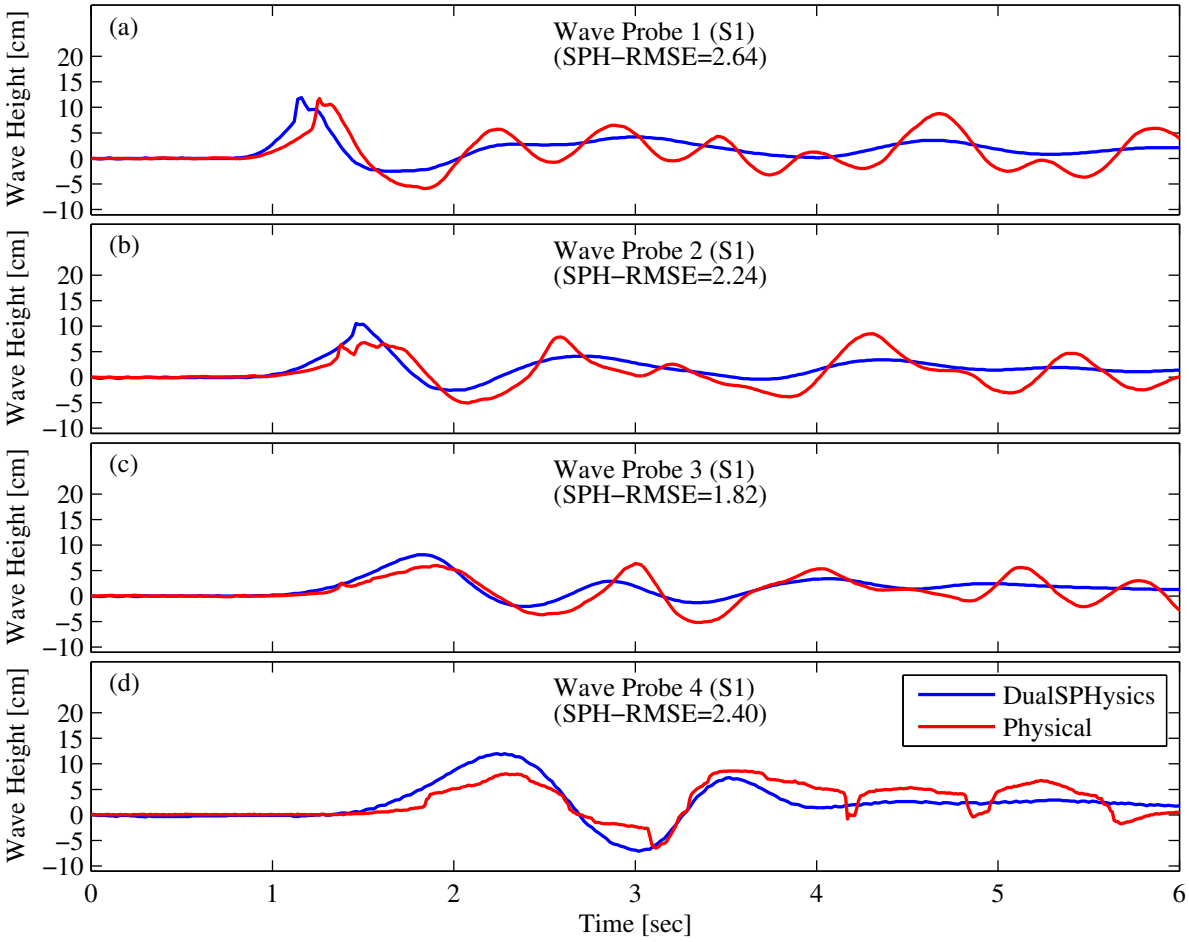

Figure 4.5: Baseline DualSPHysics and physical model data of the free surface measured at wave probe 1-4 locations.

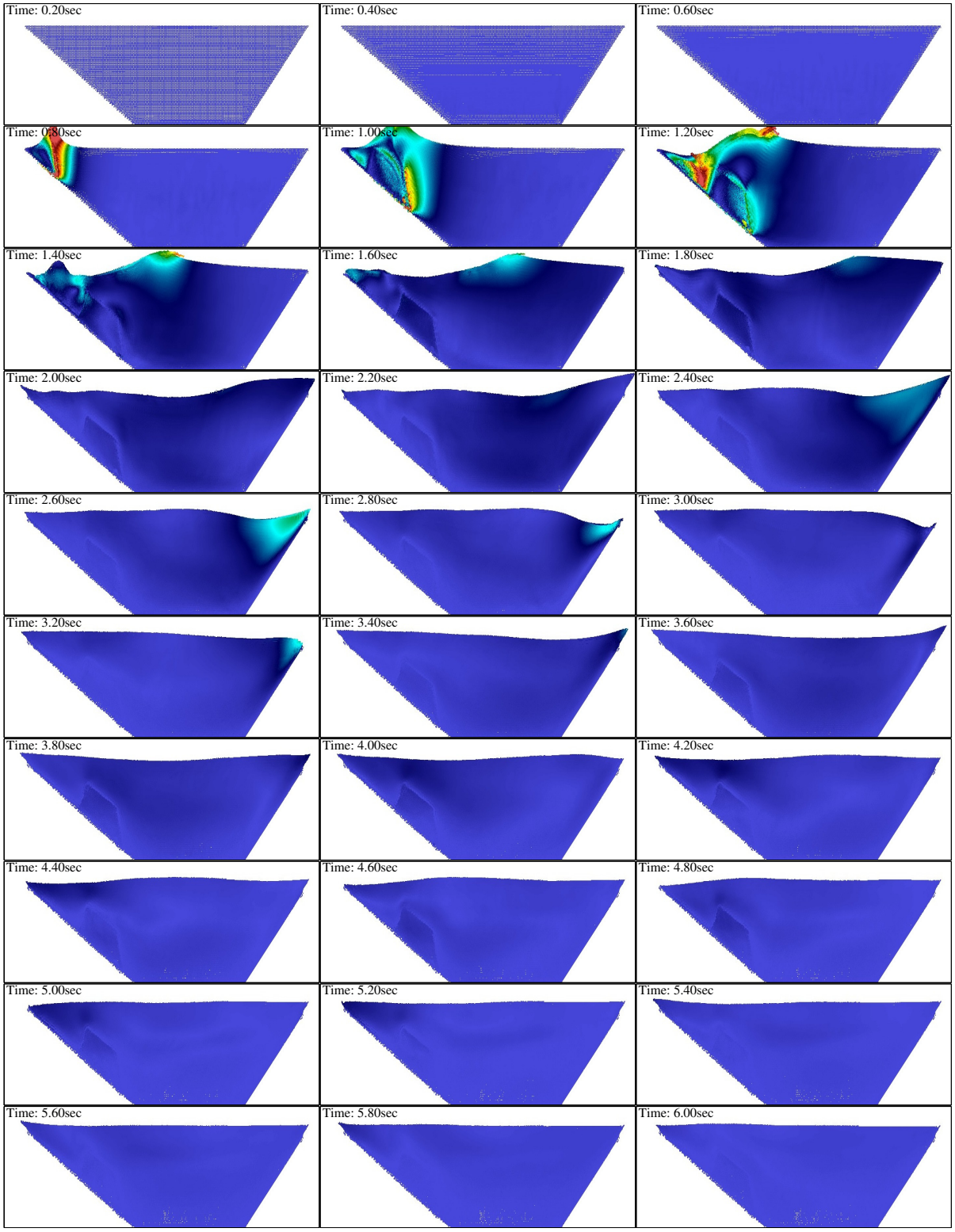

Figure 4.6: Baseline elevation view of DualSPHysics frames (Test- #S1). Note, colour gradient represent velocity magnitude, where blue is 'slow' and red is 'fast'.

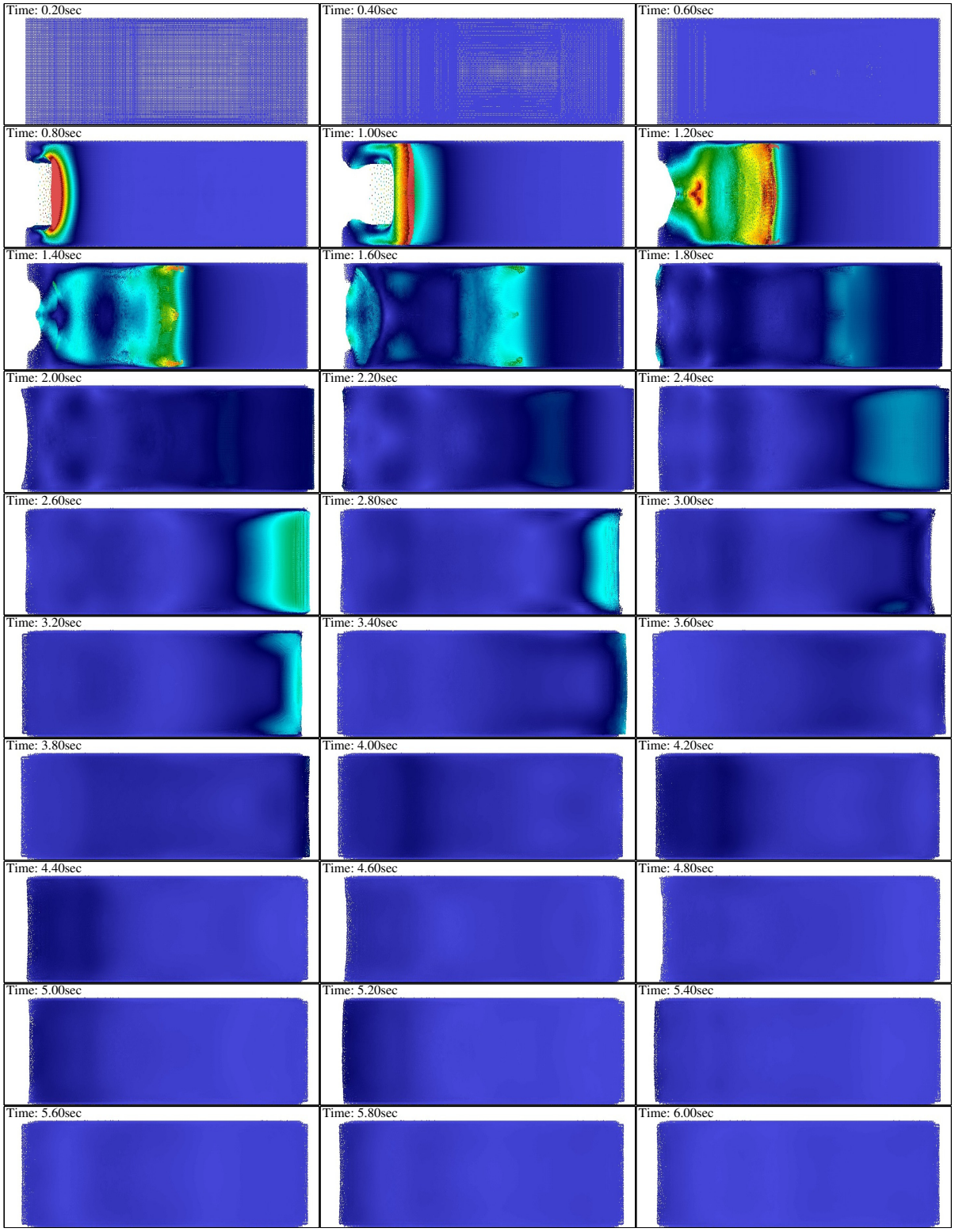

Figure 4.7: Baseline plan view of DualSPHysics frames (Test- #S1). Note, colour gradient represent velocity magnitude, where blue is 'slow' and red is 'fast'.

Therefore, with such poor agreement with the physical experiment, the following questions must be asked:

1. Is DualSPHysics accurately generating the same waves as the physical experiment?
2. Is DualSPHysics overly dampend due to inaccurate numerical dissipation?
3. Are waves accurately reflected at boundaries?

To answer these questions, a series of developmental tests were carried out to parse out the issue(s). A description of these tests and their results can be found in Section 4.2.2.

Table 4.3: Baseline DualSPHysics error

Wave probe	Relative max. wave amplitude error	Root mean square error
DualSPHysics		
1	1%	2.64-cm
2	23%	2.24-cm
3	27%	1.82-cm
4	39%	2.40-cm
Flow-3D™		
1	-36%	1.70-cm
2	-19%	1.40-cm
3	-18%	1.07-cm
4a†	96%	3.62-cm
4b‡	100%	2.82-cm

† Maximum second derivative; ‡ Threshold operator

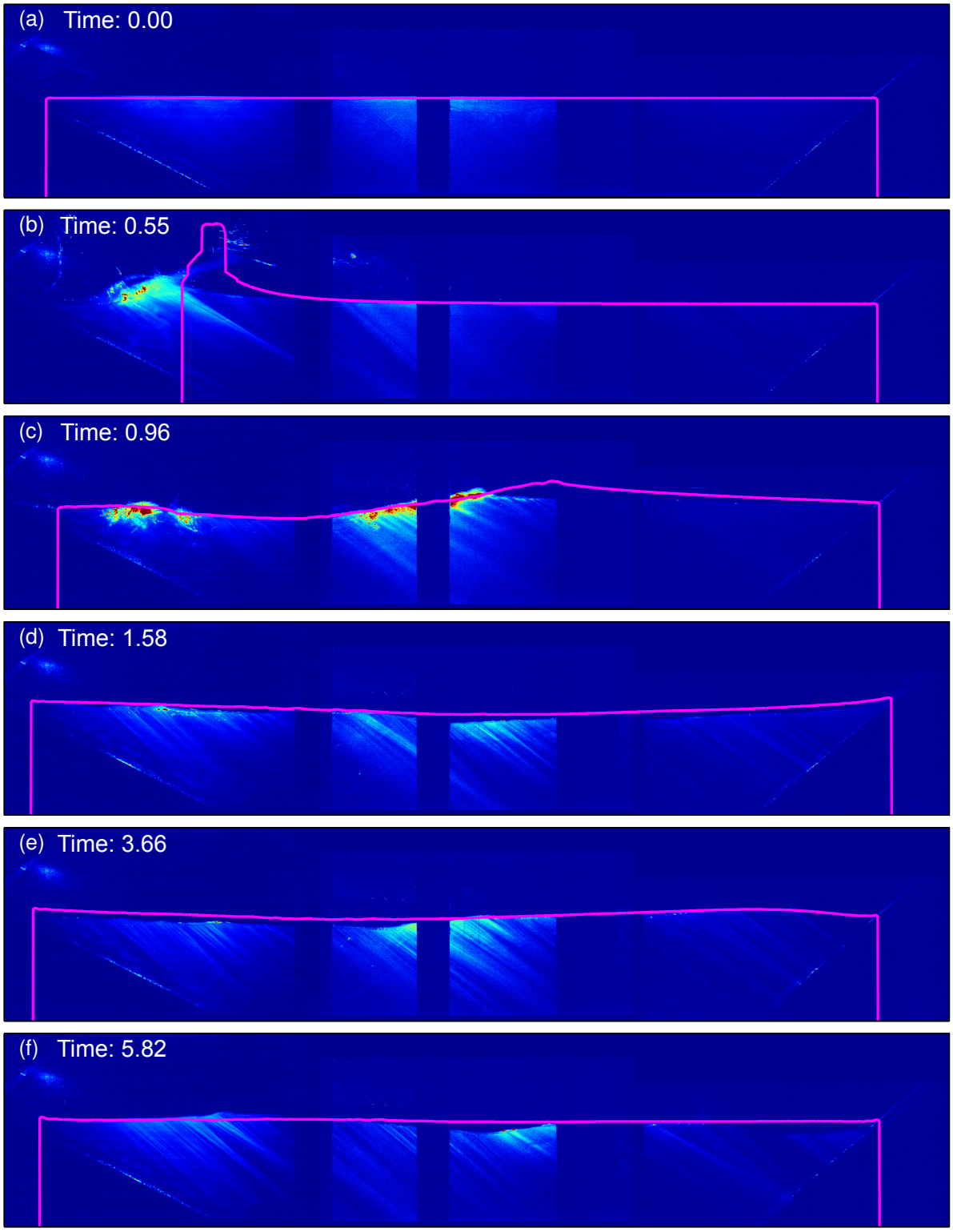

Figure 4.8: Baseline DualSPHysics centreline free surface overlaid onto physical experiment images. Panels a, b, c, d, e, and f represent time at 0.00, 0.55, 0.96, 1.58, 3.66, and 5.82-sec, respectively.

4.2 Developmental modelling

The purpose of this section is to evaluate appropriate user-controlled variables, and to discover how they influence Flow-3D^{TM} and DualSPHysics solutions, an objective of which is to find an optimal series of variable settings that can be used for similar simulations.

4.2.1 Eulerian computational fluid dynamics

Flow-3D^{TM} allows a user to select which physics need to be solved for a specific problem, the type of solver (i.e. explicit or implicit), and in some cases, what degree of accuracy (i.e. first or second-order) the solver may have.

The hydrotechnical physics packages that Flow-3D^{TM} includes are as follows:

- Air entrainment
- Bubble and phase change
- Cavitation
- Density evaluation
- Dissolving objects
- Drift-flux
- Elasto-visco-plasticity
- Fluid sources
- Granular flow
- Heat transfer
- Moving and simple deforming objects
- Porous media
- Sediment scour
- Shallow water
- Surface tension
- Viscosity and turbulence

For this work, the "moving and simple deforming objects" as well as the "viscosity and turbulence" physics packages were primarily used. However, "air entrainment", "density evaluation", "drift-flux", and "surface tension" physics packages were also used on selected simulations. Flow-3D^{TM} suggests using an explicit scheme when possible, as it usually produces a more accurate result[18]. The only exception to this is solving the pressure field. In this case, an implicit scheme is used to maintain simulation stability[18].

As for degree of accuracy, the only possible user selection is the accuracy of the momentum advection solver. The default is set to a first-order approximation; however, there is an option of a second-order solver.

To evaluate all these options, a series of seventeen development tests were conducted. Parameter changes included altering cell size (see Section 2.2), surface roughness, slide description (see Section 3.2.1), momentum advection accuracy, and which physics to include. A complete list of all the simulations (including the baseline and final test) are presented in Table 4.5. Images of these simulations are not shown because changes in the results are not easily observed through visual inspection, instead a description of observations is provided.

Changing the surface roughness of the east and west sloped walls (north and south stayed at 0-m) from 0-m (F1) to 0.00025-m (F3) did very little to change the results. Further increasing the surface roughness to 0.001-m (F6) did reduce the maximum wave run-up error from 96% to 88% when using the maximum second-derivative method (MSD), but did little for the threshold operator (TO) method. Qualitatively, 0.001-m seems too high when looking at the physical experiment surfaces, and 0-m seems inappropriate. As such, all other tests were carried out using a surface roughness of 0.00025-m.

Including Flow-3D's^{TM} default surface tension package with the baseline settings (F4) reduced the maximum wave run-up error from 96% to 94% while using the MSD method, but did little for the TO. At first, it was thought that surface tension effects may be important near the leading run-up edge, and may be responsible for energy loss in the physical experiment. However, these results suggest otherwise.

Using a second-order momentum advection scheme instead of a first-order scheme (F5) not only reduces the root-mean-square-error of all the probes, but also reduces the maximum wave run-up error from 96% to 64%. This seems appropriate, as the driving mechanism for this problem is the transfer and advection of momentum. It is also noted that "this method performs well for free surface waves"[18]. In fact, on average, using a second-order scheme versus a first-order one reduced the error by about 50%.

When reducing the baseline mesh from 0.01-m to 0.0075-m (F14), the maximum wave run-up error actually became much worse. It went from 96% to 122% while using the MSD method, and from 99% to 124% for the TO method. This is most peculiar, as a reduction in cell size tends to be more accurate. However, when using a mesh cell size of 0.0075-m and a second-order momentum advection scheme (F9), the error is reduced to 59% and 54% for the MSD and TO methods, respectively.

If the complex slide description is used (see Section 3.2.1) with the baseline settings (F12), there is no notable changes in to the results. But, if the mesh cell size is reduced to 0.0075-m (F11), then the maximum wave run-up error balloons to 130% and 137% for the MSD and TO methods, respectively.

If the second-order momentum advection scheme is used with the complex slide description and a mesh cell size of 0.0075-m (F10), the error drops to 43% and 44% for both the MSD and TO methods, respectively. However, if the mesh cell size is further reduced to 0.005-m (F15), then the maximum wave run-up error increases to 80% and 85% for the MSD and TO methods, respectively. **Note**, *besides the change in cell size, the 0.005-m simulation (F15) was executed using Flow-3D^{TM} MP v5.02.15. Theoretically, this version of Flow-3D^{TM} is identical to Flow-3D^{TM} v10.1, except for the added benefit of mesh decomposition and multi-node computations; however, there are some minor discrepancies. To determine whether or not the increase in error in F15 (when compared to F10) was due to a version issue, F10 was re-run on Flow-3D^{TM} MP v5.02.15. This test did yield different results, and the RMSE and REMA error's did change, but only by a small percentage. As such, a version issue is not considered a problem for this large increase in error.*

Finally, knowing that the baseline test neglected air entrainment, the Flow-3D's^{TM} "air entrainment" and "drift-flux" package[2] was activated while using a complex slide description and a second-order momentum advection scheme. The maximum wave run-up error for this simulation is 54% and 53% for the MSD and TO methods, respectively. This simulation was also attempted with a mesh cell size of

[2]Both the "air entrainment" and "drift-flux" packages are required for both entrainment and release of air

0.0075-m, but failed due to an excessively small time step requirement.

In summary, surface tension and surface roughness have little impact on Flow-3D^{TM} maximum wave run-up predictions, while the accuracy of the momentum advection scheme is the single most important parameter. Combinations of surface roughness, surface tension, mesh cell sizes, slide descriptions, and momentum advection accuracy do not always yield better results. In some cases, the error is magnified. That said, using a second-order momentum advection scheme, with a complex slide description, and a mesh cell size of 0.0075-m provided the most accurate Flow-3D^{TM} results for prediction of the maximum wave run-up during SLGW simulations.

Table 4.4: Developmental Flow-3D^{TM} Error

Test Number	RMSE[1]				REMA[2]				
	WS1	WS2	WS3	WS4	WS1	WS2	WS3	WS4a[3]	WS4b[4]
F1[5]	1.7	1.4	1.1	3.6	-36	-18	-17	96	99
F3	1.7	1.4	1.0	3.6	-36	-19	-17	96	100
F4	1.7	1.4	1.1	3.5	-36	-17	-17	94	99
F5	1.4	1.2	1.0	3.1	-44	-25	-30	64	65
F6	1.7	1.4	1.0	3.5	-37	-22	-17	88	98
F7	1.4	1.3	1.1	2.9	-48	-30	-23	59	63
F8	1.4	1.2	1.0	3.0	-45	-25	-30	59	63
F9	1.5	1.2	1.0	2.8	-47	-24	-27	59	54
F10[6]	1.5	1.1	1.0	2.7	-47	-20	-30	43	44
F11	1.6	1.4	1.3	3.5	-15	-10	5	130	137
F12	1.6	1.4	1.2	3.4	-30	-16	-15	96	99
F13	1.5	1.2	1.2	2.9	-48	-22	-26	55	55
F14	1.6	1.3	1.2	3.6	-26	-21	-6	122	124
F15	1.2	0.9	0.7	3.1	-38	-27	-13	80	85
F16	1.5	1.1	1.1	2.5	-47	-26	-30	48	43
F17	1.6	1.1	1.0	3.1	-47	-19	-32	54	53

[1] Root-mean-square-error [cm]
[2] Relative error of maximum amplitude [%]
[3] Maximum second-derivative
[4] Threshold operator
[5] Baseline test (Section 4.1.1)
[6] Final test

Table 4.5: Developmental Flow-3D^{TM} parameter settings

Test Number[1]	Flow-3D Version[1]	Slide Velocity	Cell Size[2]	Surface Tension[3]	Surface Roughness[2]	Momentum Advection[4]	Run Time[5]
F1[6]	10.1	Simple	0.010	No	0.00000	1st Order	6
F3	10.1	Simple	0.010	No	0.00025	1st Order	6
F4	10.1	Simple	0.010	Yes	0.00025	1st Order	-
F5	10.1	Simple	0.010	No	0.00025	2nd Order	-
F6	10.1	Simple	0.010	No	0.00100	1st Order	-
F7	10.1	Simple	0.010	No	0.00100	2nd Order	-
F8	10.1	Simple	0.010	Yes	0.00100	2nd Order	-
F9	10.1	Simple	0.008	No	0.00025	2nd Order	-
F10[7]	10.1	Complex	0.008	No	0.00025	2nd Order	26
F11	10.1	Complex	0.008	No	0.00025	1st Order	27
F12	10.1	Complex	0.010	No	0.00025	1st Order	6
F13	10.1	Complex	0.010	No	0.00025	2nd Order	11
F14	10.1	Simple	0.008	No	0.00025	1st Order	22
F15	MP 5.02.15	Complex	0.005	No	0.00025	2nd Order	33
F16	MP 5.02.15	Complex	0.008	No	0.00025	2nd Order	5
F17	10.1	Complex	0.010	No	0.00025	2nd Order	9

[1] All simulations used an implicit pressure solver, explicit free surface pressure & advection solvers
[2] Units are meters
[3] Activation of Flow-3D's default surface tension module
[4] Order of momentum advection scheme (i.e. accuracy)
[5] Run time is measured in hours of CPU time on an 8 core AMD FX-8320 3.5Hz processor
[6] Baseline test
[7] Final test

4.2.2 Smoothed particle hydrodynamics

Unlike Flow-3D^{TM}, which has an intuitive physics packages and well researched user variables, DualSPHysics is a bit less mature and requires both a strong knowledge of its underlying numerics and brute force to find the appropriate combination of variables for a given problem. Consequently, twenty-two development tests were required to find a suitable solution. Table 4.6 presents a complete list of all the tests and their settings. Table 4.7 presents the root-mean-square-error and relative error of the maximum amplitude. A description of observations follows.

Reducing the viscosity coefficient from 0.25 (S1) to 0.05 (S2-S5) produces less dissipative results, and the DualSPHysics results approach the physical experiment results (Figure 4.9 and Figure 4.10). However, as viscosity was reduced, the maximum wave run-up increased. This physically makes sense, because if less energy is dissipated, then there is more energy driving wave run-up.

DualSPHysics results are even more dissipative when the coefficient of sound is increased (S1 to S6 & S7-Figure 4.11; and S13 to S14). This agrees with Robinson's observation that "increasing the sound speed results in a significant increase in numerical dissipation"[63]). Additionally, increasing the coefficient of sound from 10 (S1/S13) to 20 (S6/S11) doubled the computational time. While going from 10 (S1) to 30 (S7) tripled the computational time.

Switching from a Cubic kernel with a smoothing length coefficient of 0.87 to a Wendland kernel with a smoothing length coefficient of 1.5 (from S2 to S8) increases the impact wave error to 113% (panel a Figure 4.12) and the maximum wave run-up error to 59% (Figure 4.12).

Reducing the particle size from 0.01-m to 0.005-m while using a smoothing length coefficient of 1.23 (see Section 2.3.1) appears to do little to enhance the solution. Aside from the spike in the initial impact wave[3] (Figure 4.12), this simulation qualitatively appears very similar to the previous simulation (S8), and does not notably change with a smaller particle size (S9 to S13). Therefore, combining this with a required 49-hours of GPU time and 450-GB of hard dive space, it was decide to keep the DualSPHysics particle sizes at 0.01-m. However, the smoothing length coefficient was kept at 1.23.

For all developmental simulations, the Delta-SPH coefficient (see Section 2.3.6) had been set to zero. That said, for test S10 (Figure 4.13) it was set to 0.1 and the simulation 'leaked' particles and failed. Consequently, the Delta-SPH coefficient was kept at zero for the reminder of the study.

In an attempt to evaluate the implications of different time integration schemes, test S12 used the same parameters as S11, but used a Symplectic scheme rather than a Verlet scheme. The difference between the two RMSEs and REMAs is marginal, however the main difference is that the Symplectic scheme took twice as long to complete and there appears to be a bug in the DualSPHysics code, as the output for wave probe 4 was all zeros. This occurred when running S21 with a Symplectic scheme (S24 - results not presented).

For test S1-S13 it is noted that the first wave generated upon impact notably differs from the physical experiment. It appears to occur before the physical wave. Also, looking at the width of the impact crater at 1.0sec (in Figure 4.7) it is apparent that its width is much larger than the Flow-3D^{TM} impact crater. In fact, when measured it is 13.25% wider than the Flow-3D^{TM} impact crafter - which is an artifact of the

[3]Likely caused by a rogue particle

Dynamic boundary particles (see Section 2.3.7) used in DualSPHysics. Consequently, scaling the slide by 86.75% about the centreline at the slide's leading edge, yields an almost perfect computed impulse wave. As such, all other simulations used a slide that was 13.25% smaller.

Above, it was asked if DualSPHysics accurately reflects waves at boundaries? To evaluate this, the North and South walls were converted from Dynamic boundary particles to open periodic boundaries. The justification for this is if two identical waves hit each other, then this would be the equivalent of a single wave hitting a vertical wall. If the dynamic boundary particles were preventing waves from being reflected, then this approach would parse that out. However, it appears from test S15 (Figure 4.15) that this is not the case. The RMSE of all the probes did decrease, but only marginally (average of RMSE from S14 was 1.95-cm, while for S15 it was 1.80). Furthermore, this setup is not physical, so simulations using this type of boundaries were no longer carried out.

In Section 2.3.5, it is explained how the particles are moved using an XSPH variant. For all simulations this was set to a default value of 0.5. However, for this test, it was normally set to zero. Meaning the particles were moved according to their individual velocity and not some fraction of their group velocity. The results of this indicate that it marginally influences the RMSE, but did reduce the relative maximum error in each probe. That said, the results are still highly dissipative.

Oger[58] states "errors in the gradient approximations are non-negligible" unless a high value of $\frac{h}{\Delta x}$ is used. Therefore, this value was increased from 2.13 (S14) to 3.5 (S17). By doing this, the number of particles in each kernel support went from 77 to 343 and dissipation marginally decreased, while the computational time tripled. As such, $\frac{h}{\Delta x}$ was reverted back to 2.13.

With the kernel type, particle size, $\frac{h}{\Delta x}$, Delta-SPH coefficient, XSPH variant, and time stepping method optimally selected as Wendland, 0.01-m, 2.13, 0, 0.5, and Verlet, respectively, viscosity was again looked at. For tests S18-S21 (Figure 4.17 and Figure 4.18) the viscosity coefficient was adjusted between 0.05, 0.01, 0.02, and 0.03, respectfully. Here it can be seen that DualSPHysics is now approaching the physical experimental results. The data is slightly out of phase, but the wave amplitudes are resolved much better. Test S21 specifically has the lowest RMSE and REMA.

For tests S1-S22 a simple velocity prescription (see, Section 3.2.1) was used. Test S23, on the other hand, used S21's settings and a complex description. The average RMSE went from 2.15 (S21) to 2.35 (S23) while the REMA only marginally changed.

In summary, DualSPHysics has many user definable settings that all play a significant role in the final solution. For this SLGW case, one of the main issues was finding the right combination that limited excess numerical dissipation and allowed higher-order waves to be generated. The optimal combination (test run S21) was found to be using a simple slide velocity, a particle size of 0.01, a Wendland kernel, a smoothing length of 1.23, a viscosity coefficient of 0.03, a coefficient of sound of 10, an XSPH coefficient of 0.5, and Verlet time integration scheme with a CFL of 0.5, which can be used for future ALGW testing.

The more pressing issue is the artificial increase in slide volume caused by the dynamic boundary particles. For this work, this issue was fixed by simply reducing the slide volume by 13.25%. However, this value is somewhat arbitrary and it is not known if all subsequent SLGW simulations should use the

same reduction or if they will have their own unique reduction value.

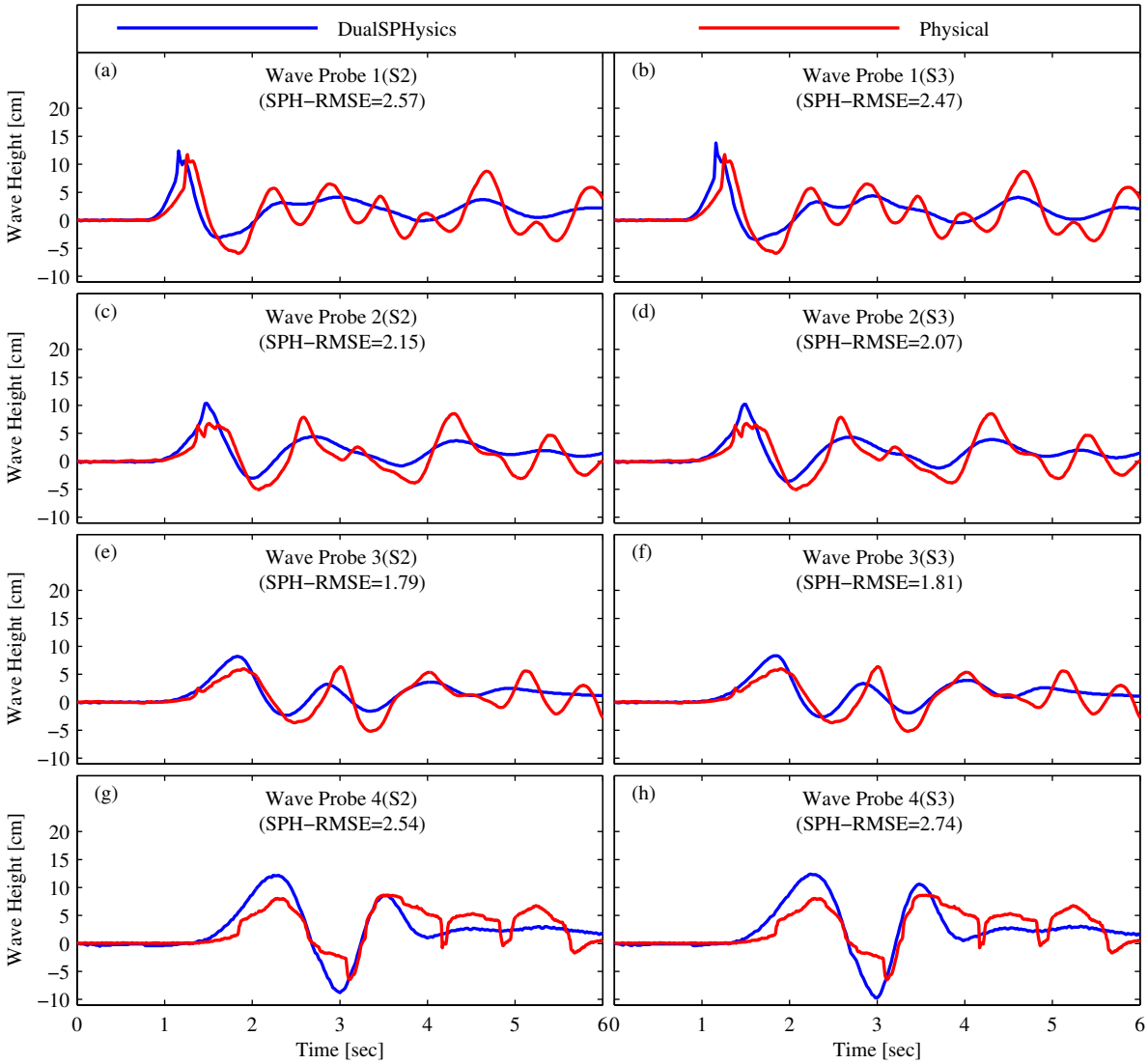

Figure 4.9: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation with a new viscosity coefficient of 0.20 (test number S2). Panels b, e, f, and h represent the baseline simulation with a new viscosity coefficient of 0.15 (test number S3)

Table 4.6: Developmental DualSPHysics parameter settings

Test Number[1]	Slide Velocity	Particle size	Kernel Type	S. Length Coeff.[2]	h/Δx[3]	Viscosity Coeff.	Coeff. of Sound	ε[4]	CFL[5]	Time Integration	Run Time[6]
S1[8]	Simple	0.01	Cubic	0.87	1.50	0.25	10.00	0.50	0.20	Verlet	3
S2	Simple	0.01	Cubic	0.87	1.50	0.20	10.00	0.50	0.20	Verlet	3
S3	Simple	0.01	Cubic	0.87	1.50	0.15	10.00	0.50	0.20	Verlet	3
S4	Simple	0.01	Cubic	0.87	1.50	0.10	10.00	0.50	0.20	Verlet	3
S5	Simple	0.01	Cubic	0.87	1.50	0.05	10.00	0.50	0.20	Verlet	4
S6	Simple	0.01	Cubic	0.87	1.50	0.25	20.00	0.50	0.20	Verlet	6
S7	Simple	0.01	Cubic	0.87	1.50	0.25	30.00	0.50	0.20	Verlet	9
S8	Simple	0.01	Wendland	1.50	2.60	0.15	10.00	0.50	0.20	Verlet	7
S9	Simple	0.005	Wendland	1.23	2.13	0.15	10.00	0.50	0.50	Verlet	49
S10[7]	Simple	0.01	Wendland	1.23	2.13	0.15	10.00	0.50	0.50	Verlet	1
S11	Simple	0.01	Wendland	1.23	2.13	0.15	40.00	0.50	0.50	Verlet	7
S12	Simple	0.01	Wendland	1.23	2.13	0.15	40.00	0.50	0.50	Symplectic	13
S13	Simple	0.01	Wendland	1.23	2.13	0.15	10.00	0.50	0.50	Verlet	3
S14	Simple	0.01	Wendland	1.23	2.13	0.15	10.00	0.50	0.50	Verlet	2
S15	Simple	0.01	Wendland	1.23	2.13	0.15	10.00	0.50	0.50	Verlet	2
S16	Simple	0.01	Wendland	1.23	2.13	0.15	10.00	0.00	0.50	Verlet	2
S17	Simple	0.01	Wendland	2.02	3.50	0.15	10.00	0.50	0.50	Verlet	6
S18	Simple	0.01	Wendland	1.23	2.13	0.05	10.00	0.50	0.50	Verlet	3
S19	Simple	0.01	Wendland	1.23	2.13	0.01	10.00	0.50	0.50	Verlet	4
S20	Simple	0.01	Wendland	1.23	2.13	0.02	10.00	0.50	0.50	Verlet	3
S21[9]	Simple	0.01	Wendland	1.23	2.13	0.03	10.00	0.50	0.50	Verlet	3
S22	Simple	0.01	Wendland	1.23	2.13	0.05	20.00	0.50	0.50	Verlet	5
S23	Complex	0.01	Wendland	1.23	2.13	0.03	10.00	0.50	0.50	Verlet	3

[1] All simulations had a particle spacing of 0.010-m (except S10, which used a 0.005-m spacing), and a Delta-SPH formulation coefficient equal to zero;
[2] Smoothing Length Coefficient; [3] Smoothing length (h) over particle spacing (Δx); [4] XSPH variant coefficient; [5] Courant-Friedrichs-Lewy Number;
[6] Run time is measured in hours of GPU time on a 448 core Nvidia Tesla C2075 card
[7] Test used a Delta-SPH coefficient equal to 0.1. Simulation also failed to complete due to particle leakage; [8] Baseline test; [9] Final test

Table 4.7: Developmental DualSPHysics Error

Test Number	RMSE[1]				REMA[2]			
	WS1	WS2	WS3	WS4	WS1	WS2	WS3	WS4
S1[3]	2.6	2.2	1.8	2.4	1	23	27	39
S2	2.6	2.1	1.8	2.5	6	22	29	42
S3	2.5	2.1	1.8	2.7	17	20	30	44
S4	2.3	2.0	1.8	3.2	31	18	37	49
S5	2.5	2.0	1.9	4.1	75	18	52	88
S6	3.0	2.5	2.1	2.4	17	9	30	27
S7	3.1	2.6	2.3	2.7	17	6	30	15
S8	2.7	2.3	2.0	2.9	19	37	43	59
S9	2.6	1.9	1.7	2.4	113	6	26	39
S10	16.2	16.1	16.0	2.4	-17	-72	-15	39
S11	3.2	2.8	2.5	2.8	29	17	41	36
S12	3.3	2.9	2.7	2.9	33	23	54	43
S13	2.6	2.2	2.0	2.8	8	42	40	57
S14	2.3	1.9	1.5	2.1	4	-31	-12	1
S15	2.0	1.5	1.3	2.4	-1	-32	-11	36
S16	2.1	1.6	1.3	2.4	-5	-24	0	21
S17	2.3	1.7	1.4	2.6	-0	-21	0	21
S18	2.1	1.6	1.4	3.2	-10	-28	-12	57
S19	2.4	1.8	1.5	4.3	-13	4	-7	85
S20	2.2	1.7	1.4	3.8	-24	-17	-6	78
S21[3]	2.2	1.6	1.4	3.4	-3	-19	-9	72
S22	2.2	1.8	1.5	2.4	9	-25	-6	27
S23	2.3	1.8	1.6	3.7	-7	-14	0	72

[1] Root-mean-square-error [cm] [2] Relative error of maximum amplitude [%]
[3] Baseline test [4] Final test

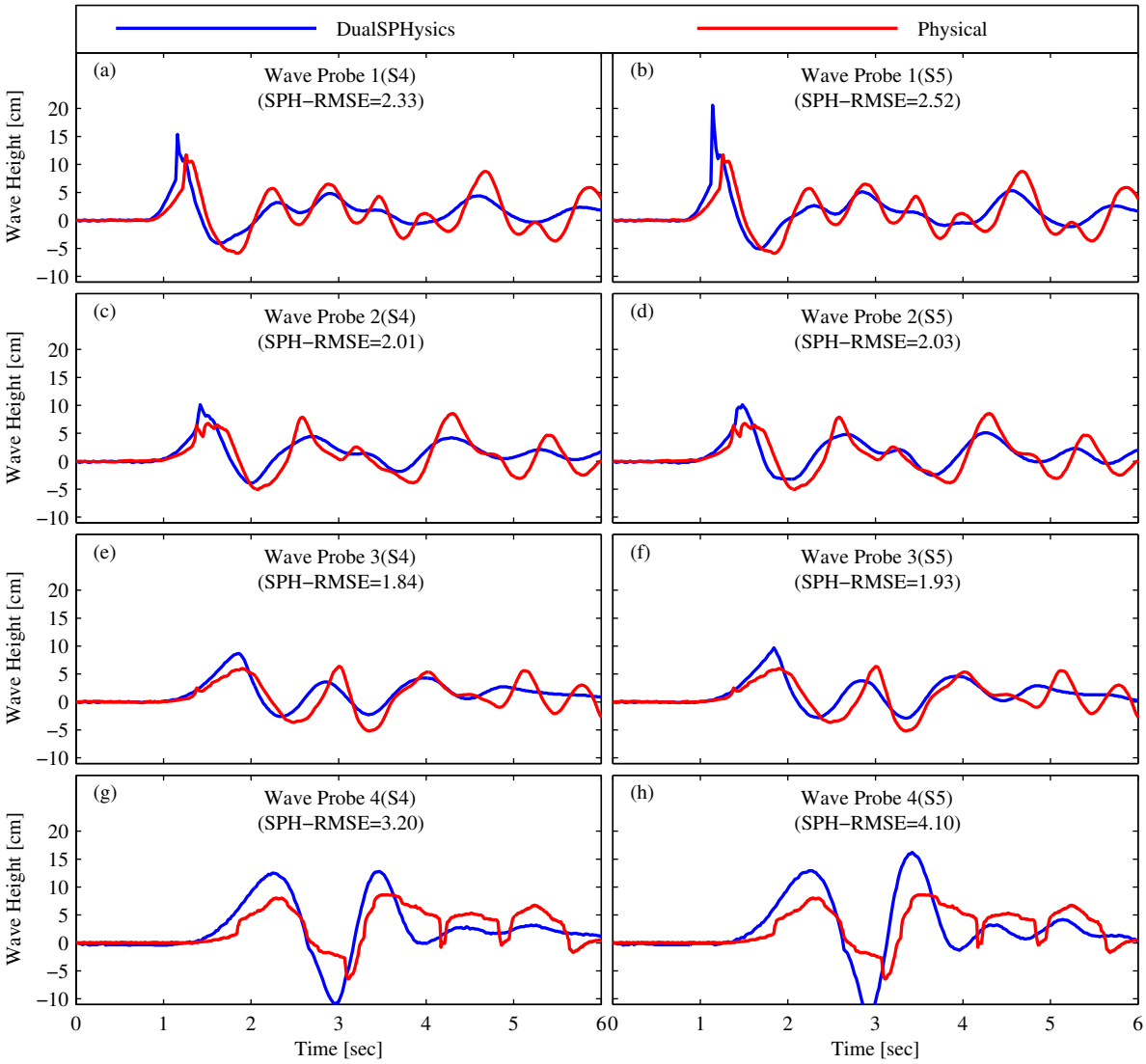

Figure 4.10: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation with a new viscosity coefficient of 0.10 (test number S4). Panels b, e, f, and h represent the baseline simulation with a new viscosity coefficient of 0.05 (test number S5)

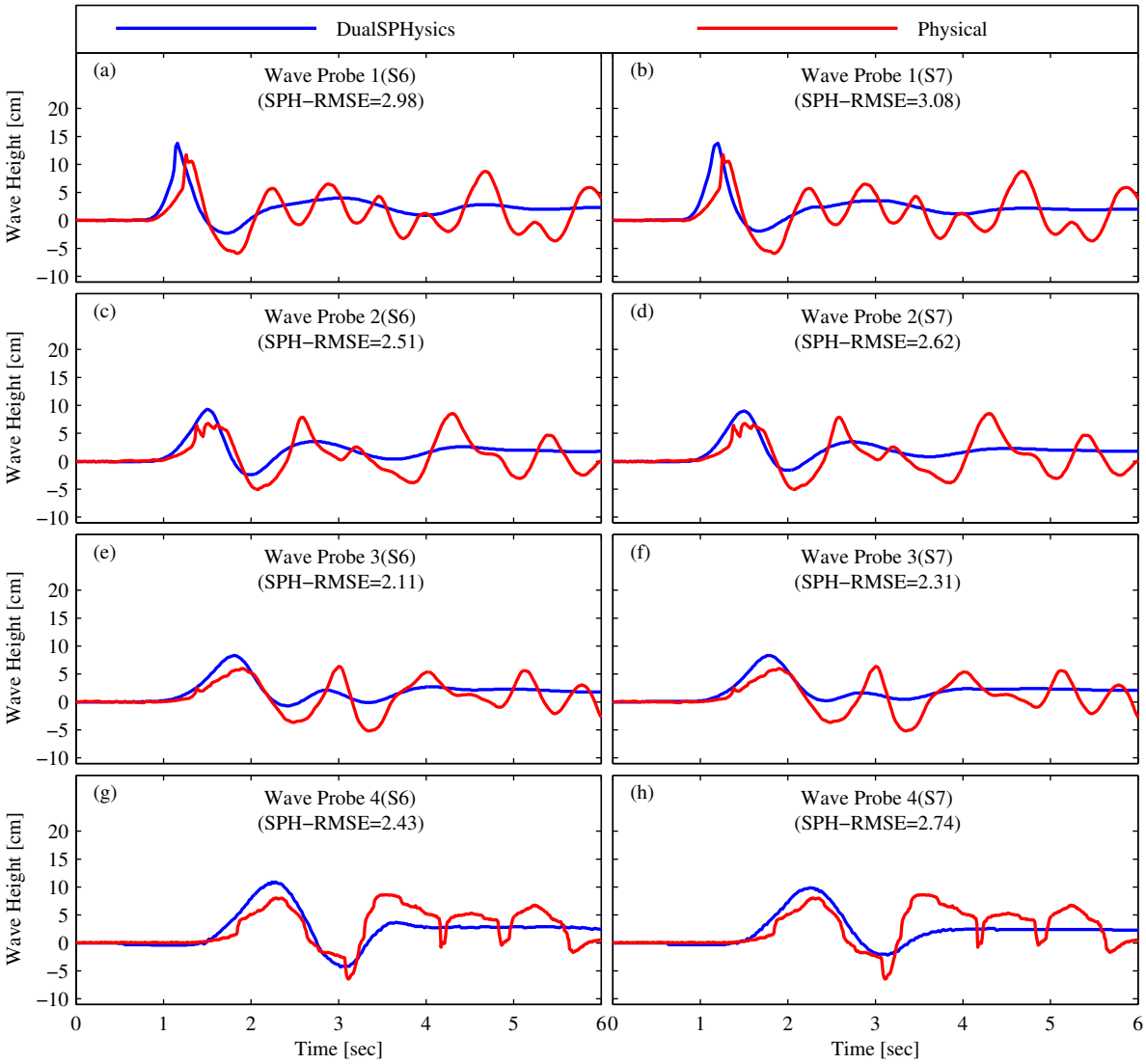

Figure 4.11: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation with a new coefficient of sound of 20 (test number S6). Panels b, e, f, and h represent the baseline simulation with a new coefficient of sound of 30 (test number S7)

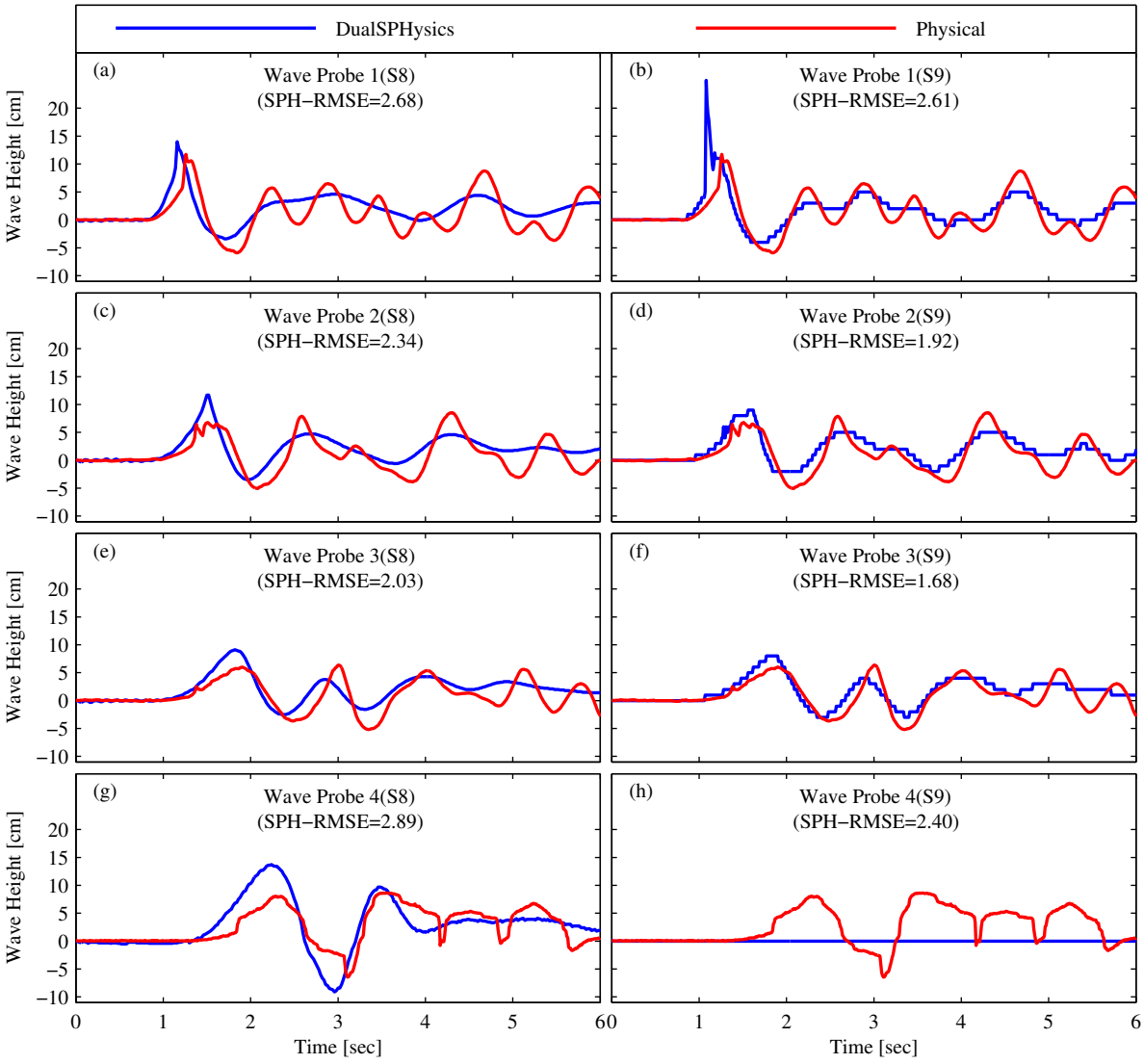

Figure 4.12: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.5, and a new viscosity coefficient of 0.15 (test number S8). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a smaller particle size of 0.005-m, a new smoothing length coefficient of 1.23, and a new viscosity of 0.15 (test number S9 - note, no wave run-up data is available).

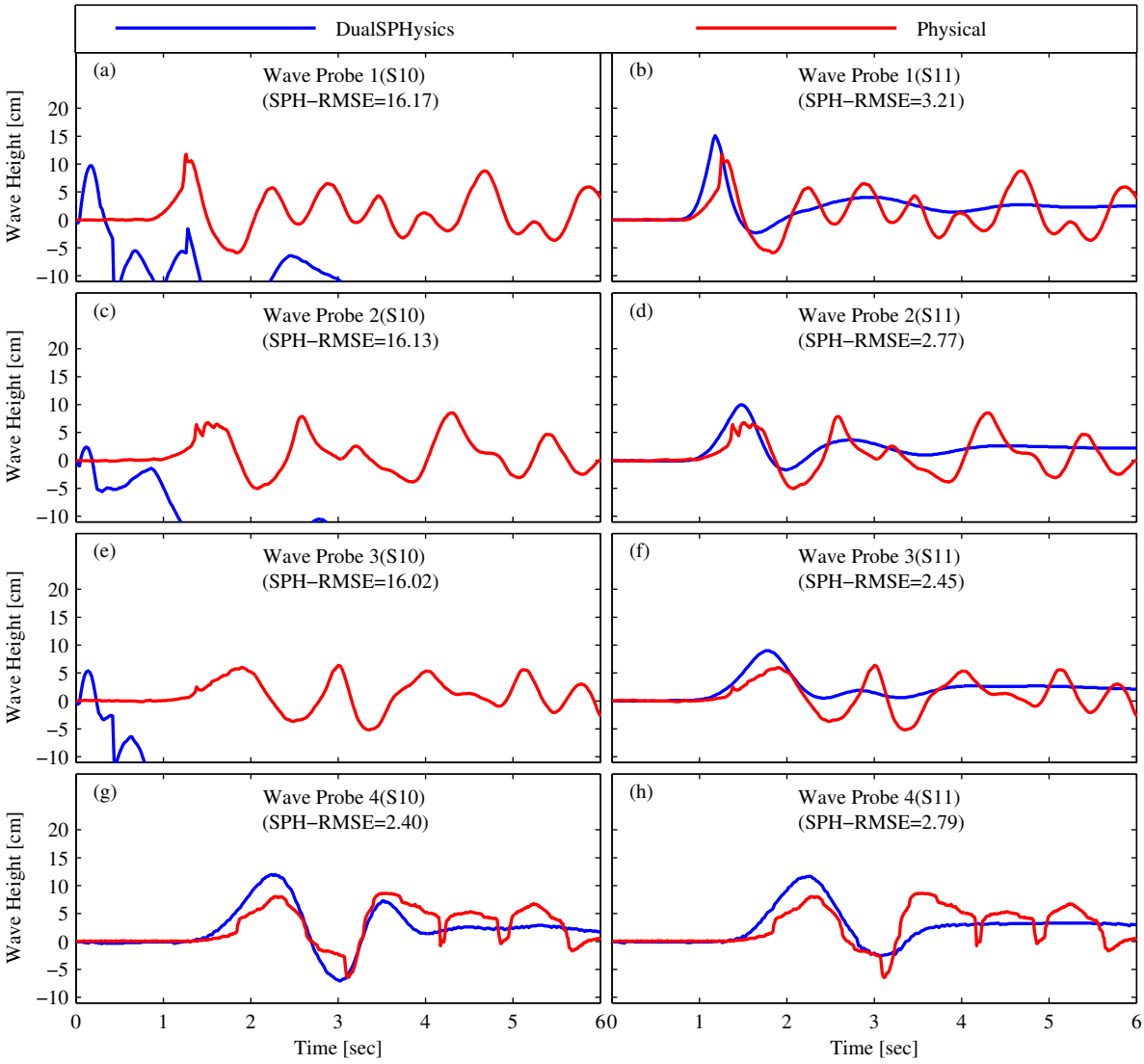

Figure 4.13: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a delta coefficient of 0.1, and a new viscosity coefficient of 0.15 (test number S10 - note due to particle leakage, simulation failed). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new coefficient of sound of 40, and a new viscosity of 0.15 (test number S11).

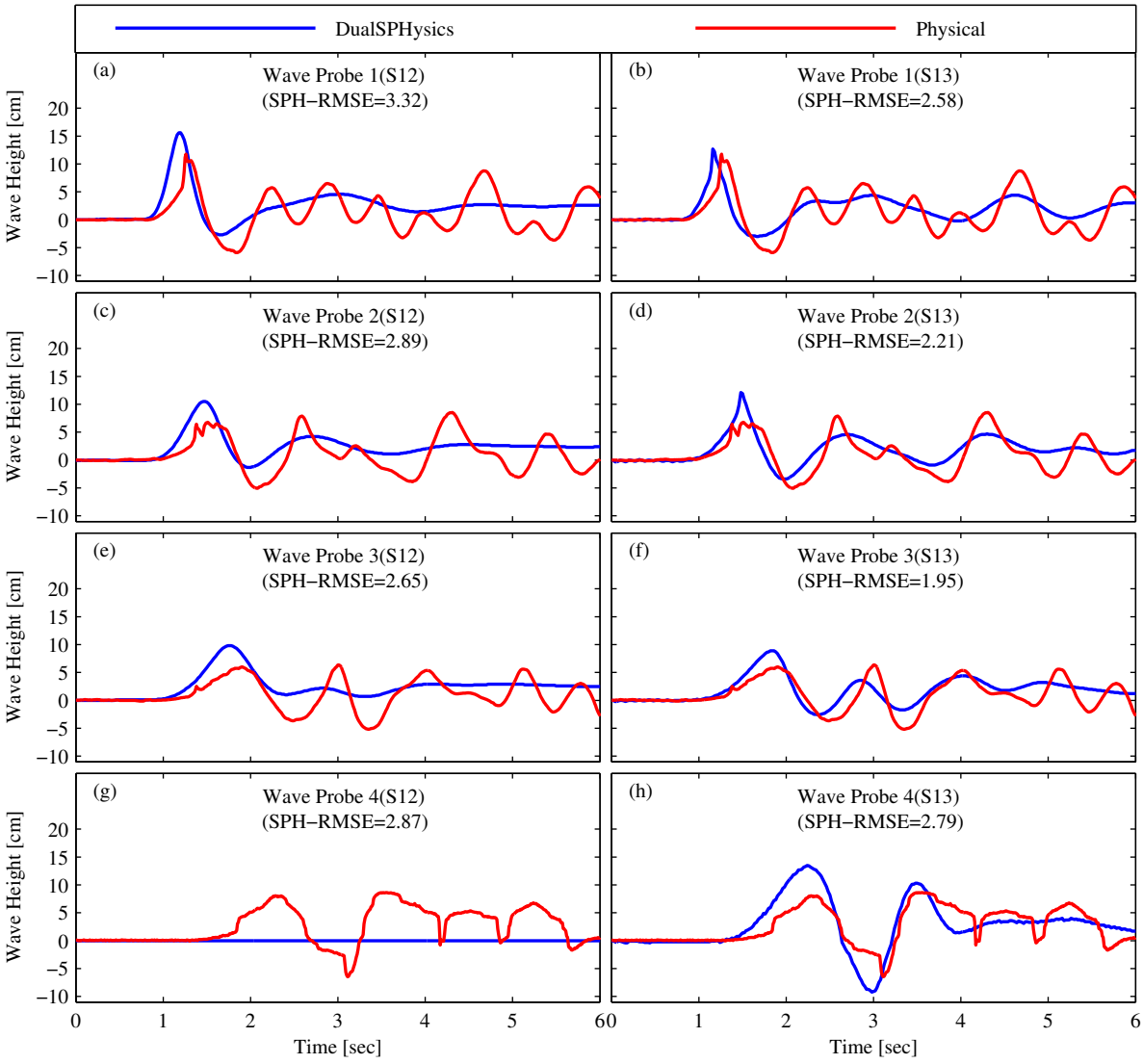

Figure 4.14: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new coefficient of sound of 40, a new viscosity of 0.15, and a symplectic time integration scheme (test number S12). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, and a new viscosity coefficient of 0.15 (test number S13).

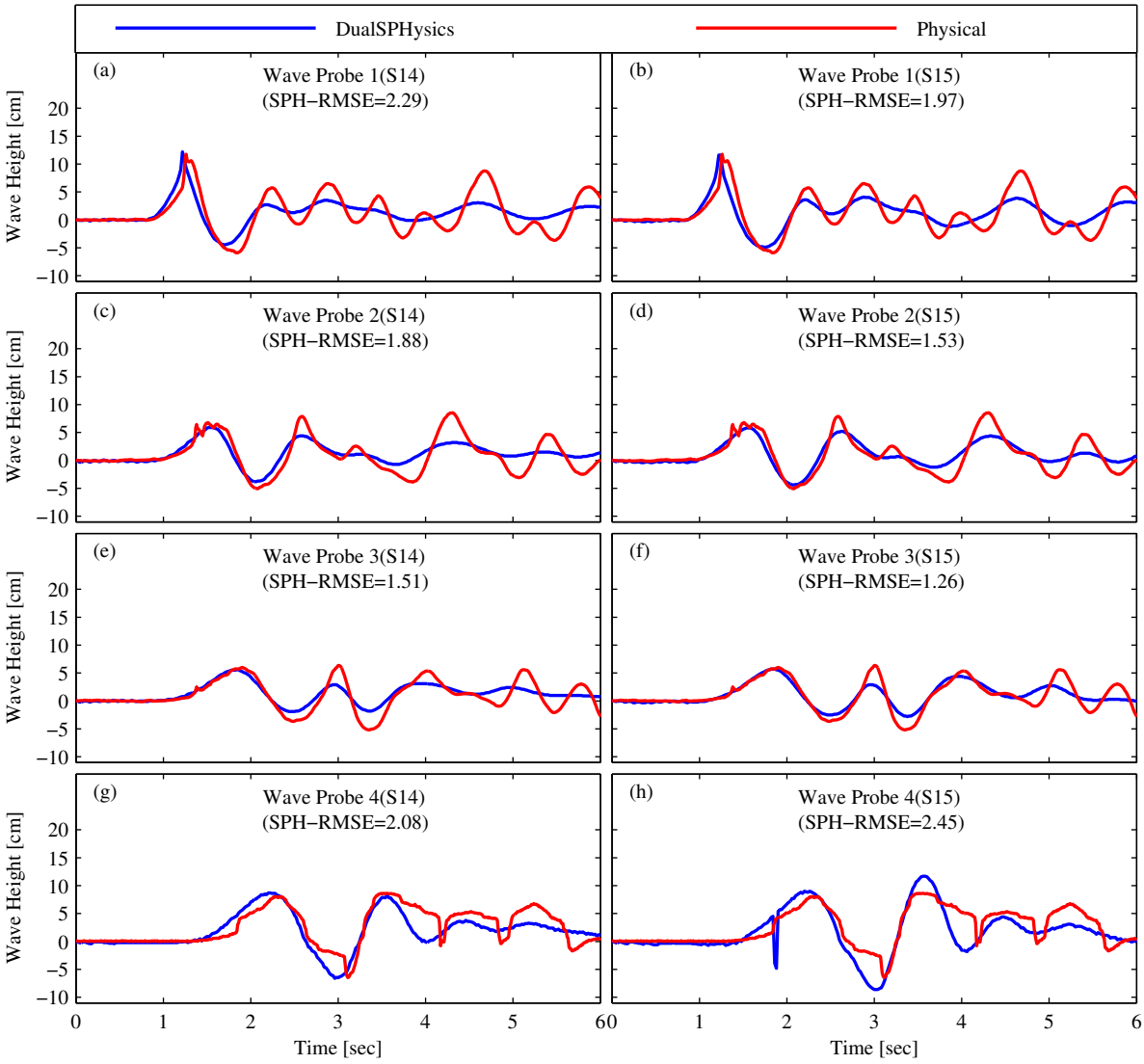

Figure 4.15: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.15, and a slide that is 13.25% smaller (test number S16). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.15, a slide that is 13.25%, and periodic boundary conditions imposed on the North and South wall (test number S15).

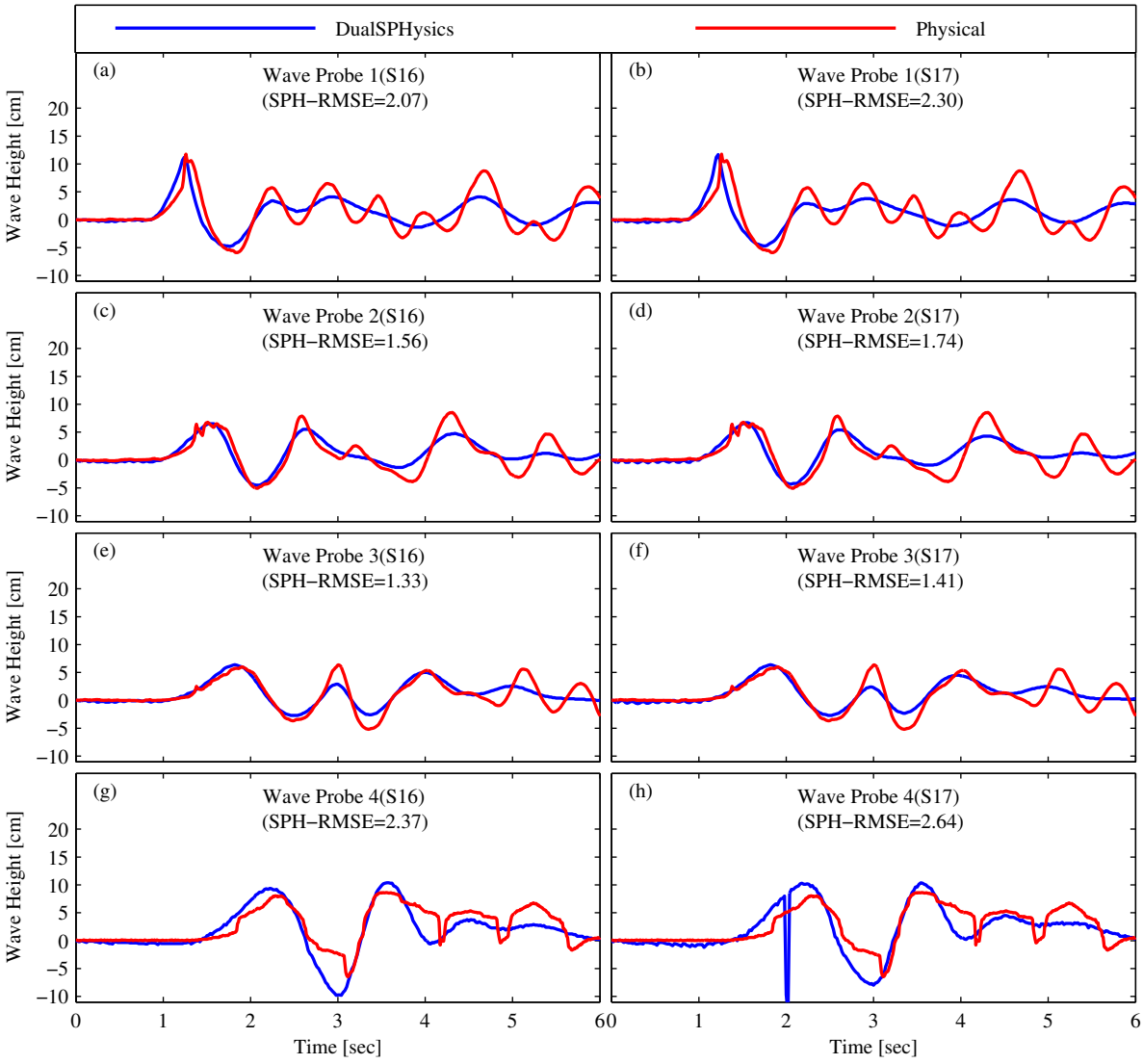

Figure 4.16: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.15, a slide that is 13.25% smaller, and an epsilon of zero (test number S16). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 2.02, a new viscosity of 0.15, and a slide that is 13.25% smaller (test number S17).

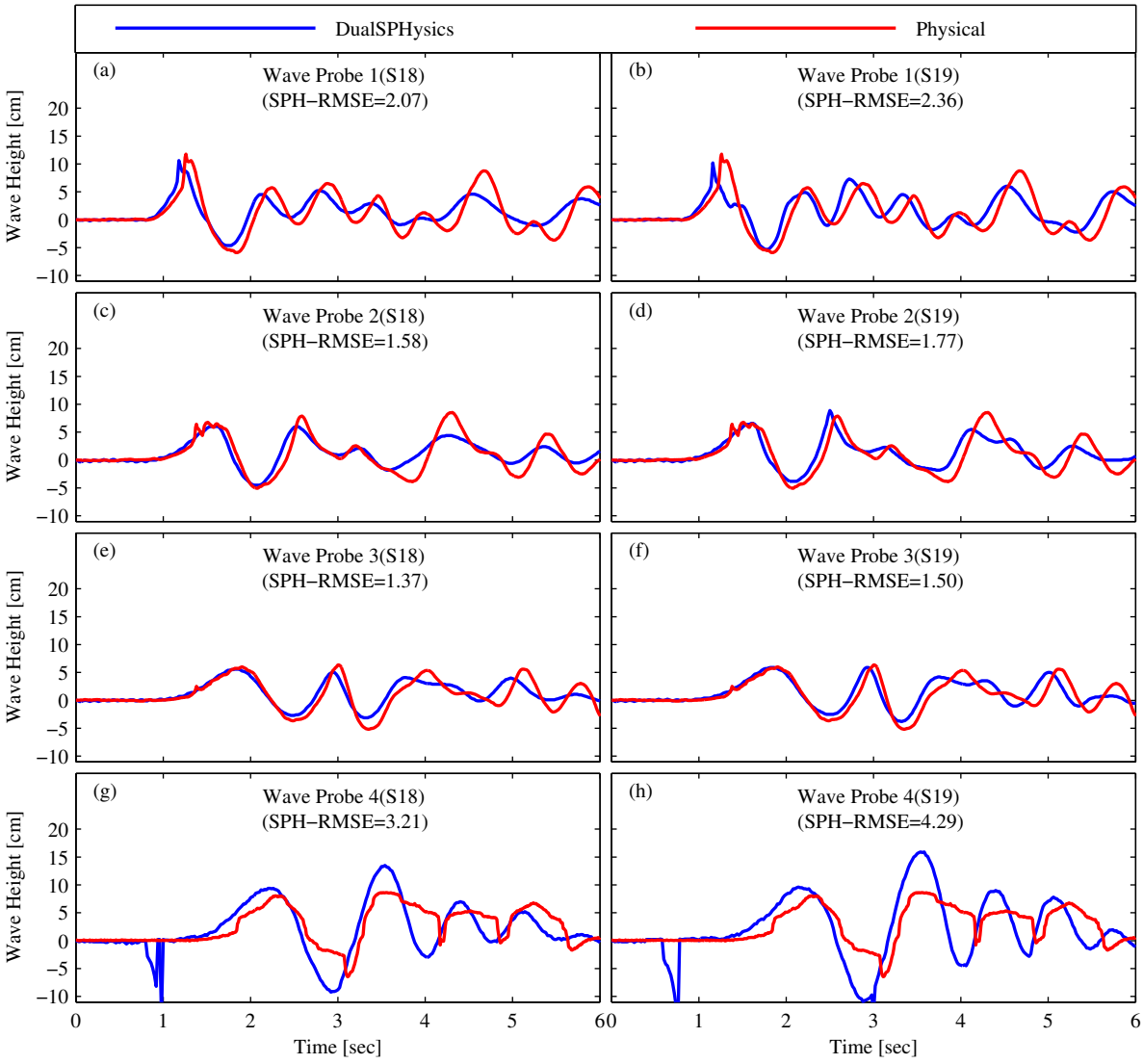

Figure 4.17: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.05, and a slide that is 13.25% smaller (test number S18). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.01, and a slide that is 13.25% smaller (test number S19).

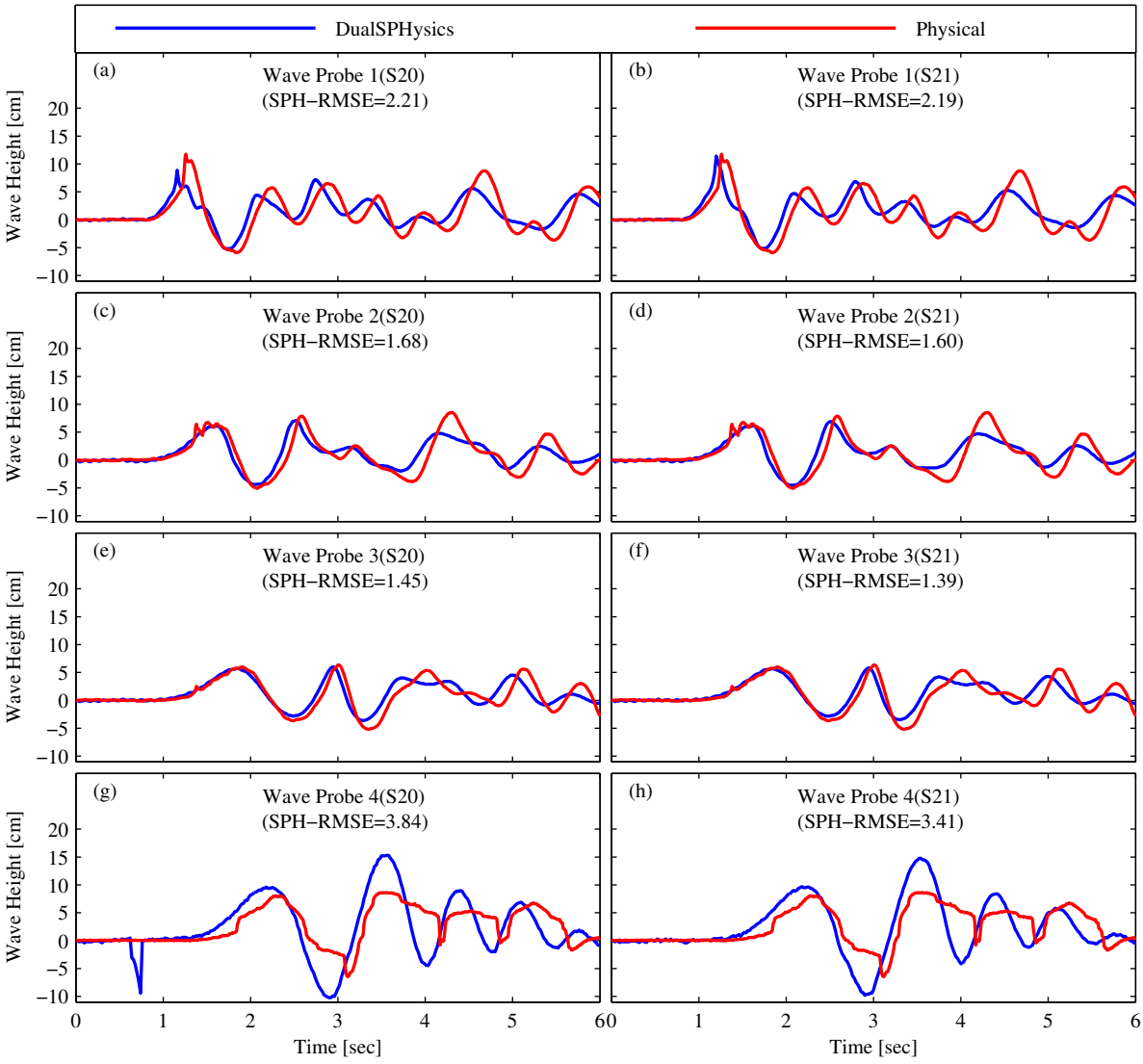

Figure 4.18: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.02, and a slide that is 13.25% smaller (test number S20). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.03, and a slide that is 13.25% smaller (test number S21).

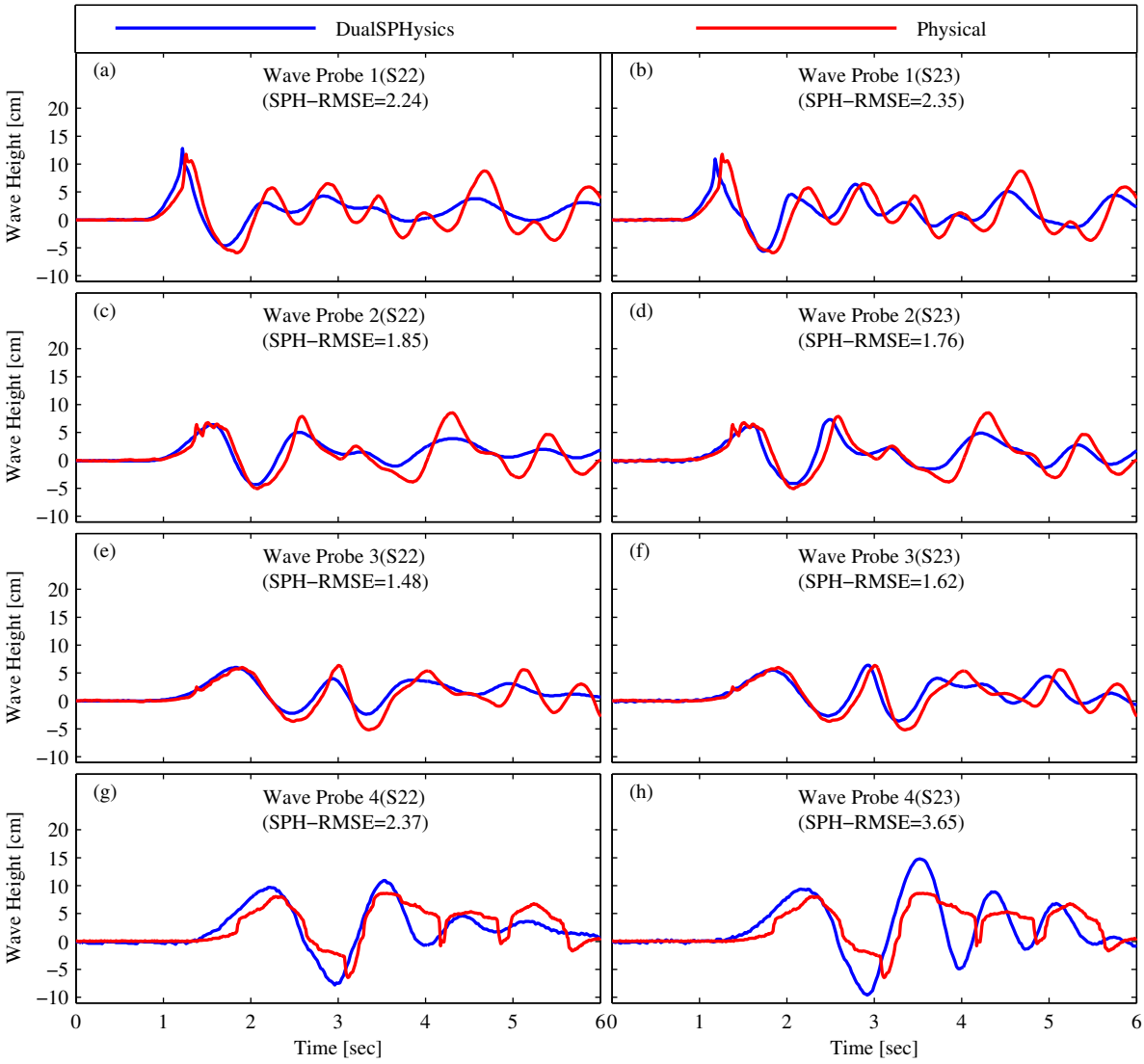

Figure 4.19: Developmental wave height comparison between DualSPHyscis & Physical Data. Panels a, c, e, and g represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.05, a slide that is 13.25% smaller, and a new coefficient of sound of 20 (test number S22). Panels b, e, f, and h represent the baseline simulation using a Wendland kernel, a new smoothing length coefficient of 1.23, a new viscosity of 0.05, a slide that is 13.25% smaller, and a complex slide description (test number S23).

4.3 Final results

Investigations of the developmental tests carried out above indicate the Flow-3D^{TM} test F10 and DualSPHysics S21 yield results that most closely represent the physical experiment results. This section provides a detailed discussion of these results.

4.3.1 Eulerian computational fluid dynamics

Of the seventeen developmental tests, test F10[4] yielded the most accurate results. Here, the initial impulse wave (Figure 4.20a at $\sim$1.2-sec) is in phase but under predicted by -46.5%, after which the wave signal improves in amplitude but becomes out of phase.

For wave probes 2 and 3, Flow-3D^{TM} reproduces the first two waves well; however, once the reflected waves start to interact with the main waves, the results become slightly less accurate. The first wave in wave probe 2 occurs in the physical experiment at 1.51-sec while it occurred in Flow-3D^{TM} at 1.56-sec and was under predicted by -34.4%. The second wave occurred in the physical model at 2.59-sec while it occurred in Flow-3D^{TM} at 2.64-sec and was under predicted by -25.9%. The maximum wave in wave probe 2 occurred at 4.31-sec with a height of 8.52-cm for the physical experiment, and 4.26-sec with a height of 6.81-cm for Flow-3D^{TM} .

The first wave in wave probe 3 occurs in the physical experiment at 1.90-sec while it occurred in Flow-3D^{TM} at 1.80-sec and was under predicted by -25.62%. The second wave occurred in the physical model at time equal to 3.00-sec while it occurred in Flow-3D^{TM} at 3.00-sec and was under predicted by -45.5%. The maximum wave in wave probe 3 occurred at time equal to 3.00-sec with a height of 6.37-cm for the physical experiment, and 1.80-sec with a height of 4.46-cm for Flow-3D^{TM} .

The first run-up on wave probe 4 occurs in the physical experiment at 2.35-sec while it occurred in Flow-3D^{TM} at either 2.26 or 2.24-sec and was under predicted by either -4.75% or -5.27%, depending on which method was used to capture the run-up (see Section 4.1). The second run-up occurred in the physical model at 3.49-sec while it occurred in Flow-3D^{TM} at 3.56 or 3.62-sec and was over predicted by either 43.0% or 44.3%. The maximum run-up in wave probe 4 occurred at time equal to 3.49-sec with a height of 8.6-cm for the physical experiment, and 3.56 or 3.62-sec with a height of 12.32 or 12.43-cm for Flow-3D^{TM} .

[4]Test F16 also generated a suitable result; however, it was executed using Flow-3D^{TM} MP v5.02.15 and is not overly used in the hydrotechnical community

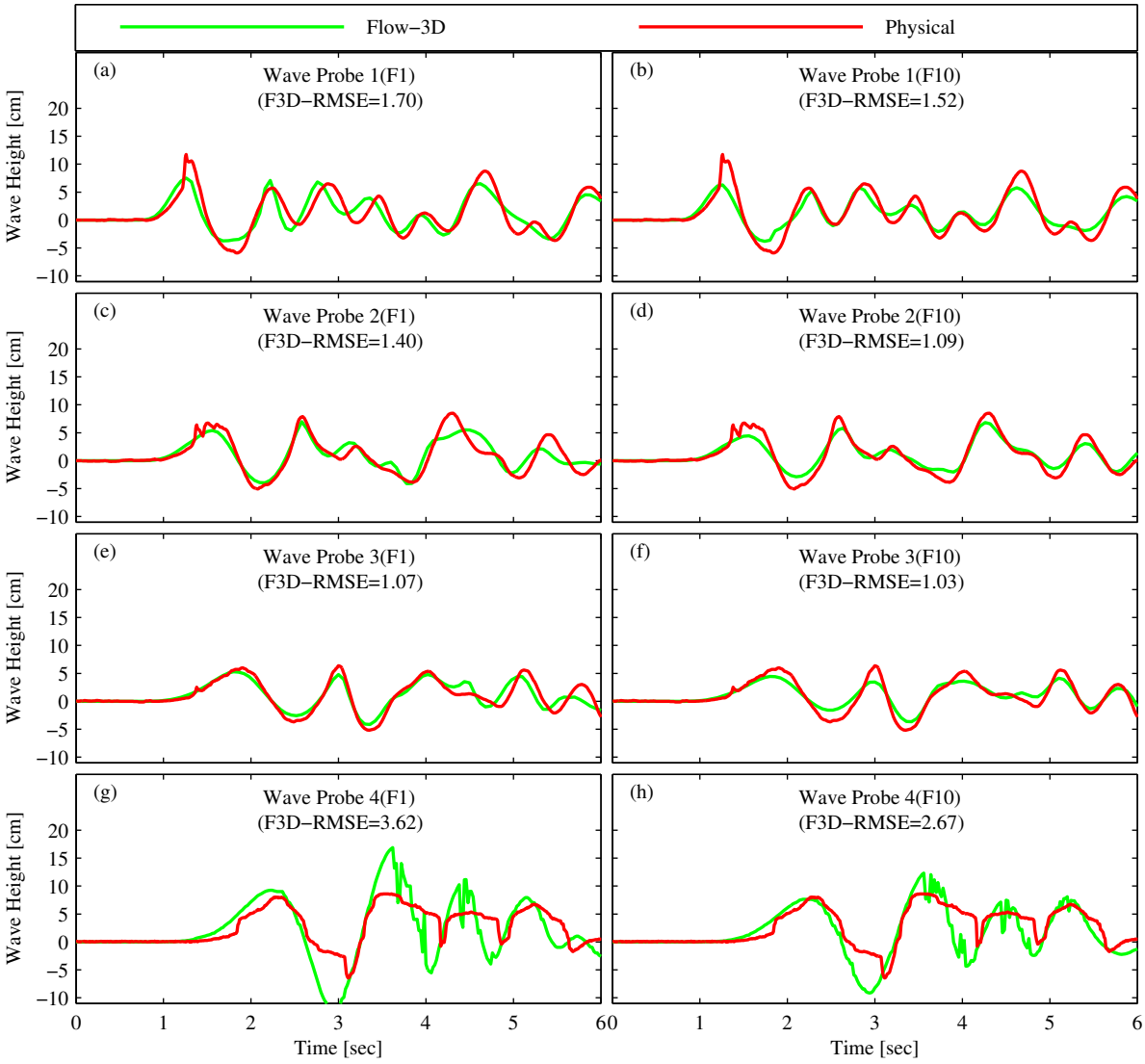

Figure 4.20: Finalized (F10) and baseline (F1) Flow-3D^{TM} and physical model data of the free surface measured at wave probe 1-4 locations. Panels g and h represent the evaluated wave run-up using the maximum second derivative method.

4.3.2 Smoothed particle hydrodynamics

Of the twenty-three DualSPHysics simulations, the twenty-first (S21) yielded the most acceptable results.

Comparing Figure 4.5 and Figure 4.21 (see also Figure 5.1) it is evident that the final test (S21) is much less dissipative than the baseline test (S1). Improvement in fluidity can be seen in Figure 4.22 and Figure 4.23.

The slide impact (Figure 4.21a at $\sim$1.2-sec and Figure 4.24b) generates an impulse wave that is out of phase by 0.06-sec and under predicted by -3%, after which the wave signal gets more out of phase and under predicts amplitudes. This seems appropriate as the reflected waves (Figure 4.21d are incorrectly reproduced.

For wave probes 2 and 3, DualSPHysics reproduces the first two waves well; however, once the reflected waves start to interact with the main waves, they become less accurate. The first wave in wave probe 2 occurs in the physical experiment at 1.51-sec while it occurred in DualSPHysics at 1.62-sec and was under predicted by -8.4%. The second wave occurred in the physical model at 2.59-sec while it occurred in DualSPHysics at 2.50-sec and was under predicted by -12.5%. The maximum wave in wave probe 2 occurred at time equal to 4.31-sec with a height of 8.52-cm for the physical experiment, and 2.50-sec with a height of 6.74-cm for DualSPHysics .

The first wave in wave probe 3 occurs in the physical experiment at 1.90-sec (Figure 4.21) while it occurred in DualSPHysics at 1.84-sec and was under predicted by -4.9%. The second wave occurred in the physical model at 3.00-sec while it occurred in DualSPHysics at 2.94-sec and was under predicted by -9.0%. The maximum wave in wave probe 3 occurred at 3.00-sec with a height of 6.37-cm for the physical experiment, and 2.94-sec with a height of 5.80-cm for DualSPHysics.

While for wave probes 1-3, DualSPHysics tended to under predict wave heights, wave probe 4 was over predicted. The first run-up occurs in the physical experiment at 2.35-sec while it occurred in DualSPHysics at 2.18-sec and was over predicted by 21%. The second run-up occurred in the physical model at 3.49-sec while it occurred in DualSPHysics at 3.54-sec and was over predicted by 71.8%. The maximum run-up in wave probe 4 occurred at 3.49-sec with a height of 8.6-cm for the physical experiment, and 3.54-sec with a height of 14.8-cm for DualSPHysics. That said, it is interesting to note that although the maximum wave run-up is over predicted, DualSPHysics actually locates the horizontal extent of the run-up almost perfectly (Figure 4.24d).

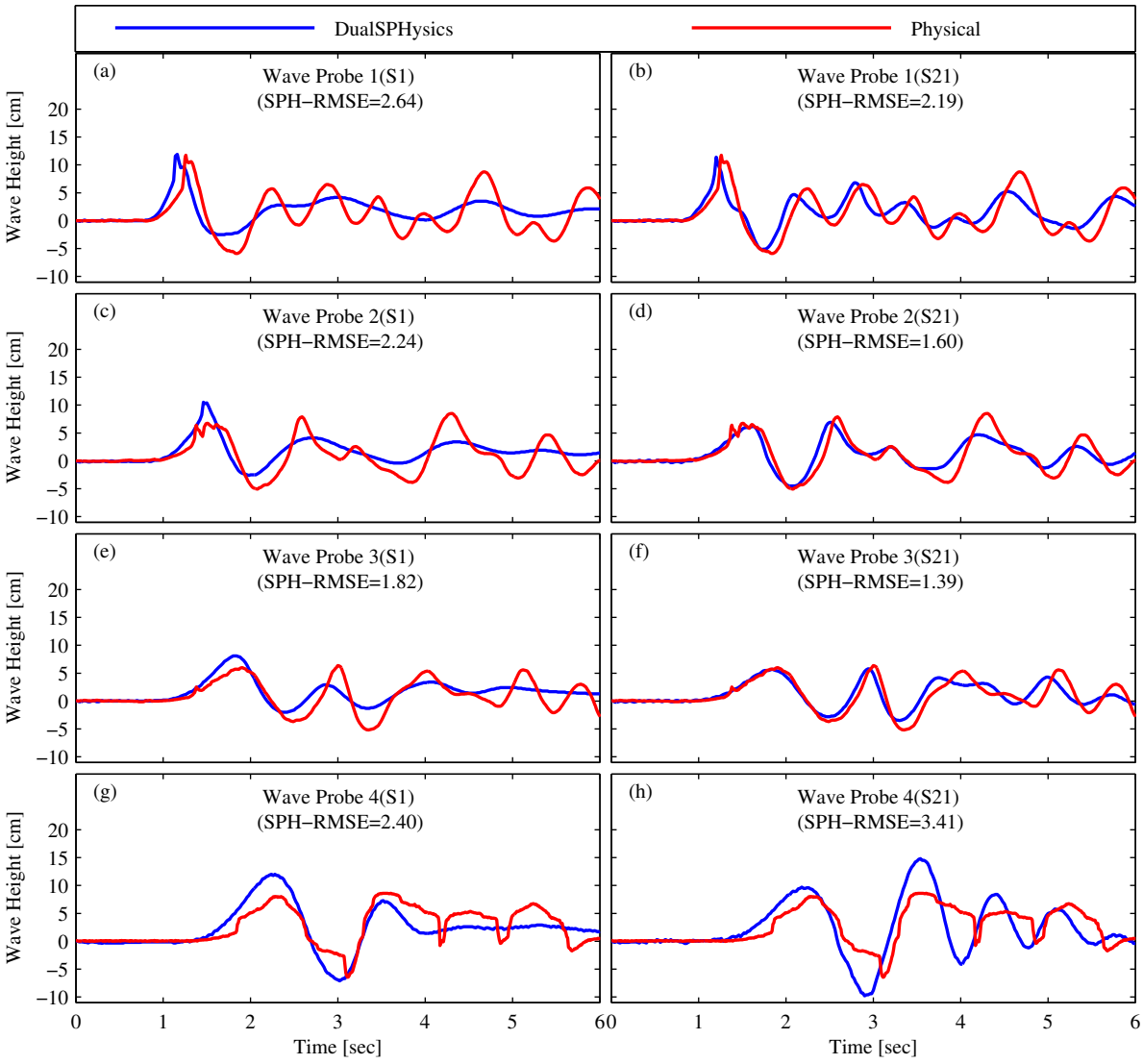

Figure 4.21: Finalized (S21) and baseline (S1) DualSPHysics and physical model data of the free surface measured at wave probe 1-4 locations.

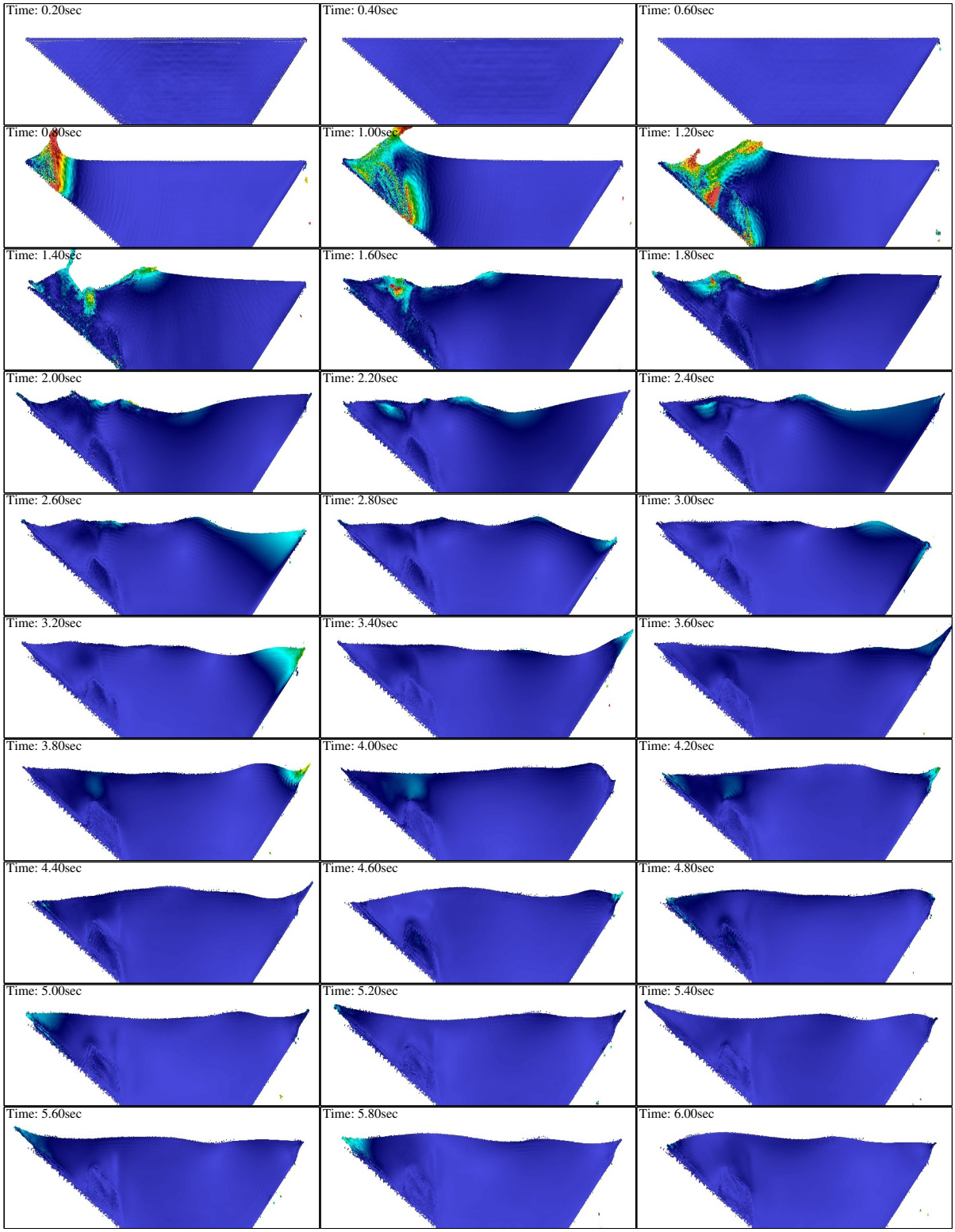

Figure 4.22: Finalized elevation view of DualSPHysics frames (test S21). Note, colour gradient represent velocity magnitude, where blue is 'slow' and red is 'fast'.

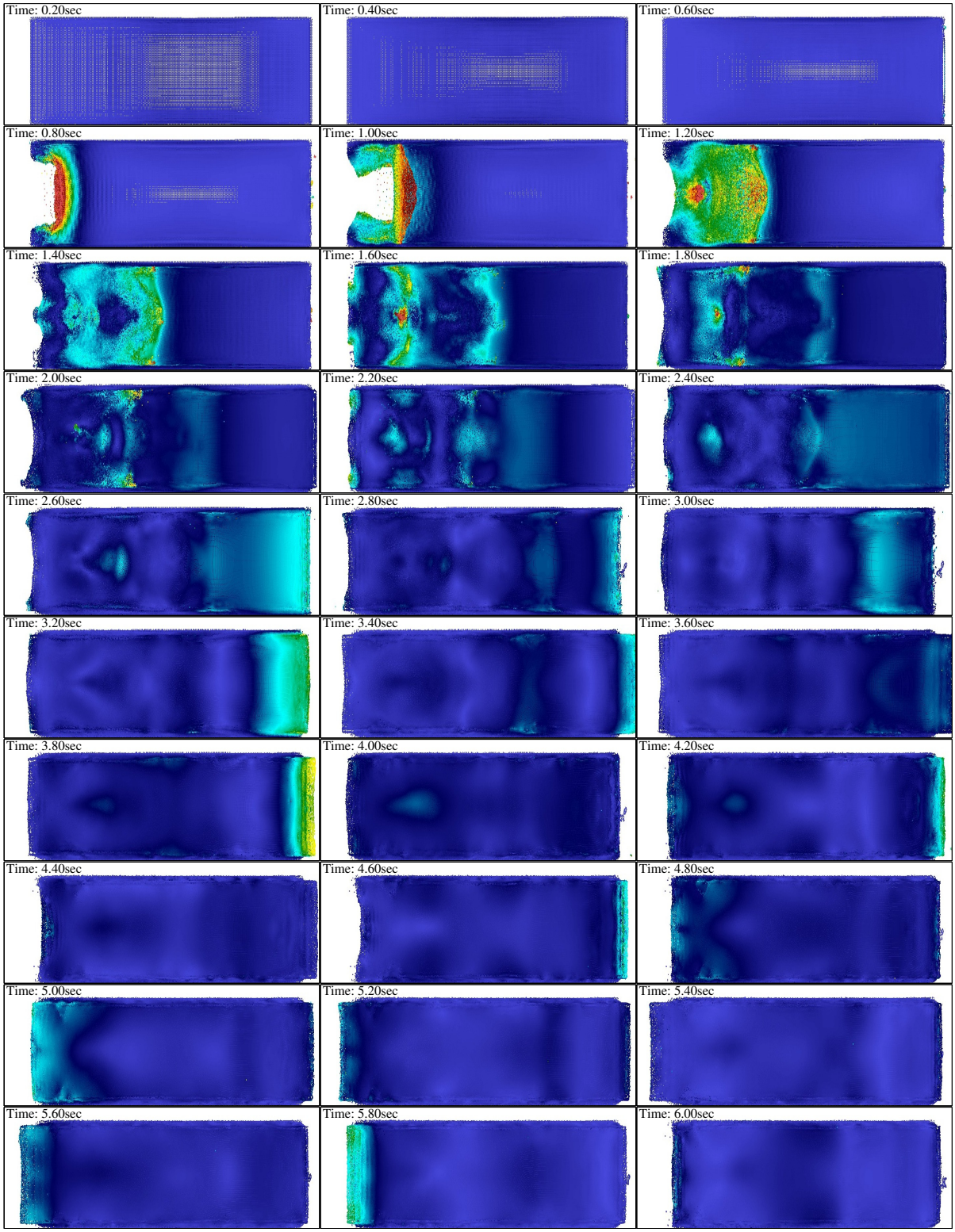

Figure 4.23: Finalized plan view of DualSPHysics frames (test S21). Note, colour gradient represent velocity magnitude, where blue is 'slow' and red is 'fast'.

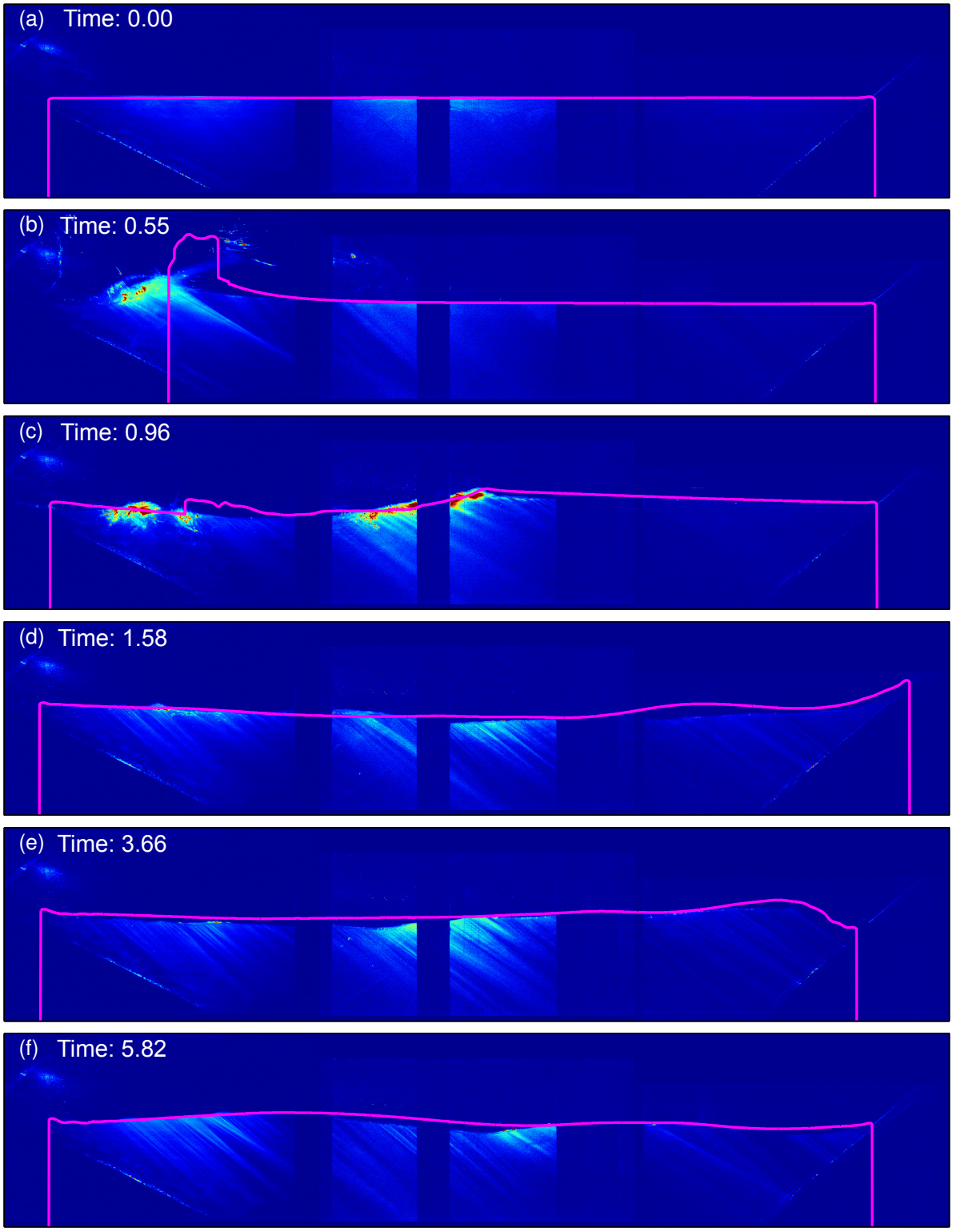

Figure 4.24: Final (test S21) DualSPHysics centreline free surface overlaid onto physical experiment images. Panel's a, b, c, d, e, and f represent time at 0,00, 0.55, 0.96, 1.58, 3.66, and 5.82-sec, respectively.

Chapter 5

Discussion and Conclusions

The aim of this research was to conduct the very first DualSPHysics simulations for three-dimensional inland subaerial landslide generated wave (SLGW) events, and to validate these simulations using high fidelity physical model data. In addition, this research aids in the continuing effort of validating the general purpose computational fluid dynamics model Flow-3D^{TM} for SLGW events.

From the physical model experiment, the maximum wave amplitude was 0.117-m above the still water level and occurred at 1.2-sec. This wave then traveled down the flume where it runs-up the opposing slope to a height of 0.080-m above the still water level at 2.35-sec. The second run-up occurred 1.14-sec later at 3.49-sec, and went up to a height of 0.086-m above the still water level, which was the maximum observed run-up. The reason for the second run-up height being higher then the first is speculated as either the result of constructive interference between reflected waves, and/or the increase in volume associated with slide now being submerged.

Using Flow-3D^{TM}, it was found that with the default settings, Flow-3D^{TM} achieved a good comparison with physical experiment results. However, changing the default first-order momentum advection scheme to a second-order scheme decreased the maximum wave run-up error by 50%. Here, it was able capture the phase of the first generated wave at 1.2-sec, but under predicted its height by -34.4%. As such, it under predicted the first run-up wave by about 5%, but surprisingly over predicted the second run-up wave (i.e. the maximum wave run-up) by about 43% and was out of phase by 0.07-sec to 0.13-sec.

As discussed in Section 1.1.4, Basu[4] and Kim[37] also indicate that Flow-3D^{TM} over predicts the maximum wave run-up of SLGW by 28-56% and 0-78%, respectively — which is consistent with this research. However, unlike Basu and Kim, this research explored, in detail, the effect of both numerical parametrization (i.e. numerical representation of specific parameters) and numerical scheme prescription. Specifically, this research investigated the influence of including Flow-3D's^{TM} air entrainment and surface tension modules, as well as looking at the influence of first- or second-order momentum advection schemes. The results of which suggest that while neither air entrainment or surface tension significantly influence the maximum wave run-up, selection of a second-order momentum advection does play a significant role. As such, it turns out by simply applying a second-order momentum advection scheme to the baseline Flow-3D^{TM} simulation (F1), the maximum wave run-up error is reduced

from 99% to 44%, as shown in Figure 5.1.

DualSPHysics, on the other hand, required significant adjustments to reproduce the physical experiment. Additionally, scaling down the slide to 86.75% of its original size was required to counter the artificial increase in slide volume due to the presence of dynamic boundary particles. After which, DualSPHysics almost exactly reproduced the first generated wave (see Figure 5.1). It was out of phase by 0.06-sec and under predicted the value by -3%. However, the first run-up wave was over predicted by 21% and the second (i.e. the maximum wave run-up) by 72%, which is 29% more error than Flow-3D^{TM}. Despite that, DualSPHysics actually out-performed Flow-3D^{TM} when predicting the maximum wave generated by the slide.

From this work, wave propagation in DualSPHysics has been shown to be significantly influenced by SPH viscosity coefficient and the coefficient of sound. High values of each coefficient drastically increases energy dissipation, and decreased the "fluidness" of the simulated fluid. For this research, the ideal viscosity coefficient and coefficient of sound was 0.03 and 10, respectively. In addition, it has been shown that using a Wendland smoothing kernel is better at limiting artificial energy dissipation in wave propagation then a Cubic kernel is.

Although the treatment of boundaries in DualSPHysics has been shown to be problematic, it does not appear to influence wave propagation nor wave reflection. In DualSPHysics, boundaries are treated as static particles that do not move with the fluid particles. As such, their effect on the SPH integral interplant is the development of a pressure gradient which repels neighbouring fluid particles, but does not induce a no-slip boundary (i.e. zero velocity of fluid particles at the boundary) condition, which therefore does not account for energy losses along a "rough" boundary. For wave reflection, this treatment works very well. By replacing the north and south solid boundaries with periodic boundaries (i.e. the north domain edge is influenced by the south domain edge, and visa versa), an idealized condition of perfect reflection can be evaluated for impinging waves. Here, it is assumed that a wave impacting a solid wall is similar to a wave hitting its mirrored self. Therefore, if a DualSPHysics simulation with dynamic boundaries greatly differed from a simulation with periodic boundaries, it would be known that dynamic boundaries are negatively influencing the fluid particles. However, from this research it seems this is not the case. Waves reflected off solid boundaries comprised of dynamic boundary particles closely match waves reflected due to impact with their mirrored selfs.

The issue with boundary particles in DualSPHysics, is that they artificially increase the size of the boundary. For example, this research shows that the slide used to generated subaerial landslide generated waves was artificially larger in the DualSPHysics domain then what was prescribed. As such, the generated waves were significantly different then that of the physical model. To combat this issue, the slide was scaled down by 13.25%.

Despite this problem being a boundary driven problem, and the issues with the boundary implementation in DualSPHysics, DualSPHysics is much better at generating waves than Flow-3D^{TM}. It has been shown that DualSPHysics can generate the first impulse wave to within -3% of the peak amplitude, while Flow-3D^{TM} generated it to within 46%. However, Flow-3D^{TM} does much better at wave propagation and wave run-up than DualSPHysics does.

The implementation of boundaries in Flow-3D^{TM} is significantly more mature and robust than in DualSPHysics , which prevents issues with dubious energy losses and/or reflections along a boundary. However, Flow-3D's^{TM} implementation of momentum advection[1], does influence artificial energy losses, and subsequently maximum wave run-up. For example, this research has shown that going from a first- to a second-order momentum scheme yields an average reduction in relative maximum wave run-up error by 50%.

In conclusion, although DualSPHysics and Flow-3D^{TM} can sufficiently reproduce a SLGW event for some applications, they may not yet be suitable for high precision applications. Furthermore, neither model is suitable for fully capturing all the wave generation, wave propagation, and wave run-up details. Rather, it has been shown that DualSPHysics is better at wave generation, while Flow-3D^{TM} is better at wave propagation and wave run-up. That said, DualSPHysics and Flow-3D^{TM} both tended to over predict the maximum wave run-up by 43-72%.

[1] In a Eulerian frame of reference, mass, momentum, and velocity need to be advected throughout a computation domain to solve the Navier-Stokes equations. As such, the order of the numerical scheme used to advect momentum dictates how much numerical viscosity is introduced.

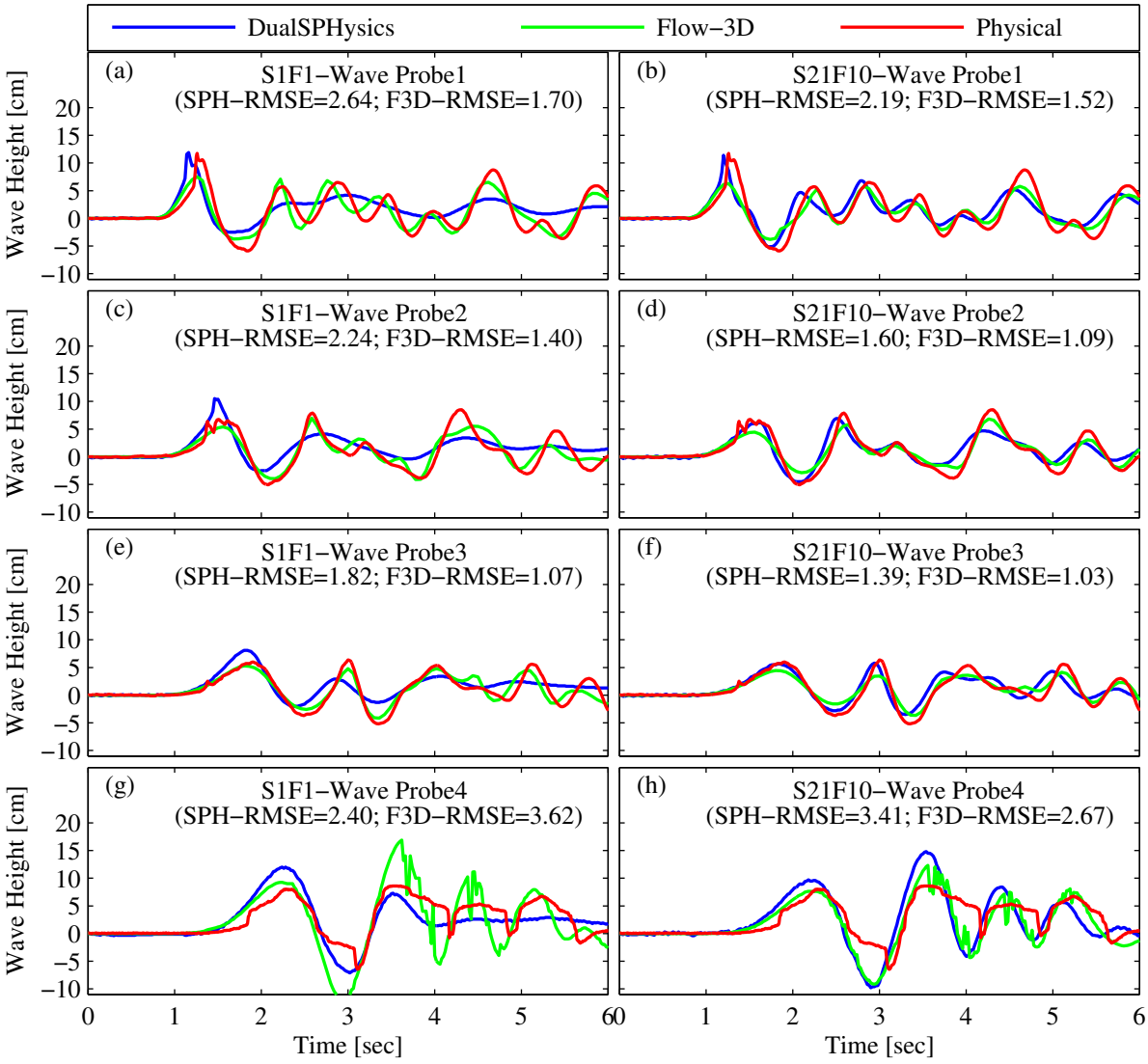

Figure 5.1: Finalized and baseline DualSPHysics, Flow-3D^{TM}, and physical model data of the free surface measured at wave probe 1-4 locations. For the Flow-3D results in panels g and h, the maximum second derivative method is used to present the evaluated wave run-up.

5.1 Future work

Despite much time spent investigating and researching, it is not known why Flow-3D^{TM} or DualSPHysics over predicts the maximum wave run-up. However, two theories are here proposed:

1. **Energy dissipation through air entrainment:** When the slide hits the water it generates a wave that travels the length of the flume and runs-up the east slope, which Flow-3D^{TM} and DualSPHysics capture. As the slide travels completely through the free surface, an impact crater is formed and water starts to collapse behind the slide. The collapse of water generates a spectrum of waves that interacts with the initial impulse wave to generate the second run-up on the east slope, which is the maximum wave run-up. If the collapse of water is not accurately simulated, or energy loss due to air entrainment is not accurately represented, then the driving energy will not be the same as the physical model, and the maximum wave run-up will differ. However, if this be the case, then why are the signals in wave probes 1-3 so much closer to the physical experiment? It is speculated that these apparently minor discrepancies are compounded at the east wall due to the ratio of wave lengths and flume length. That is, net constructive wave interference is likely occurring right at the east slope.

2. **Eigenfrequency:** During impact of the slide with the water, a wide spectrum of waves could be generated and subsequently propagate in all three axis. Over time, these waves interact with one another and generate an Eigen frequency that matches that of the flume. At this point, the wave run-up 'stalls' and the free surface along the east wall no longer moves. After a few tenths of a second the frequency is lost and the wave runs down the east slope. If this is the case, then neither Flow-3D^{TM} or DualSPHysics would be able to recognize this, since they would not be taking into account the eigenfrequency of the flume.

These of course are working theories that require more research to verify.

Bibliography

[1] G. B. Airy. Tides and waves. *Encyclopaedia Metropolitan*, 5:241–392, 1841. → pages 7

[2] B. Ataie-Ashtiani and A. Nik-Khah. Impulsive waves caused by subaerial landslides. *Environmental Fluid Mechanics*, 8(3):263–280, 2008. ISSN 1573-1510. doi:10.1007/s10652-008-9074-7. → pages 4, 5, 6, 10

[3] B. Ataie-Ashtiani and G. Shobeyri. Numerical simulation of landslide impulsive waves by incompressible smoothed particle hydrodynamics. *International Journal for Numerical Methods in Fluids*, 56(2):209–232, 2008. ISSN 1097-0363. doi:10.1002/fld.1526. → pages 6

[4] D. Basu, J. Stamatakos, K. Das, S. Green, and R. Janetzke. Numerical simulation of surface waves generated by a subaerial landslide at lituya bay alaska. *Journal of Offshore Mechanics and Arctic Engineering*, 132(4), 2010. ISSN 0892-7219. doi:10.1115/1.4001442. → pages 10, 12, 69

[5] J. Boussinesq. Thorie de l'intumescence liquide, applele onde solitaire ou de translation, se propageant dans un canal rectangulaire. *Comptes Rendus de l'Academie des Sciences*, 72: 755–759, 1871. → pages 7

[6] S. H. Chue. Wave runup formula of universal applicability. *ICE Proceedings*, 69(4):1035–1041, 1980. ISSN 1753-7789. doi:10.1680/iicep.1980.2185. → pages 9

[7] A. Colagrossi, G. Colicchio, C. Lugni, and M. Brocchini. A study of violent sloshing wave impacts using an improved SPH method. *Journal of Hydraulic Research*, 48(Extra Issue):94–104, 2010. doi:10.3826/jhr.2010.0008. → pages 22

[8] R. Courant, K. Friedrichs, and H. Lewy. ber die partiellen differenzengleichungen der mathematischen physik. 100:32–74, 1928. → pages 23

[9] A. D. D. Craik. The origins of water wave theory. *Annual Review of Fluid Mechanics*, 36(1): 1–28, 2004. doi:10.1146/annurev.fluid.36.050802.122118. → pages 7

[10] A. J. C. Crespo, M. Gomez-Gesteira, and R. A. Dalrymple. Boundary conditinos generated by dynamic particles in SPH methods. *Computers, Materials & Continua*, 5(3):173–184, 2007. → pages 21, 22

[11] A. J. C. Crespo, J. M. Dominguez, A. Barreiro, M. Gomez-Gesteira, and B. D. Rogers. GPUs, a new tool of acceleration in CFD: Efficiency and reliability on smoothed particle hydrodynamics methods. *PLoS ONE*, 6(6):e20685, 2011. doi:10.1371/journal.pone.0020685. → pages 15, 24

[12] I. G. Currie. *Fundamental Mechanics of Fluids*. Marcel Dekker, New York, NY, 3rd edition, 2003. ISBN 0-8247-0886-5. → pages 12

[13] R. A. Dalrymple and B. D. Rogers. Numerical modeling of water waves with the SPH method. *Coastal Engineering*, 53(23):141–147, 2006. ISSN 0378-3839. doi:10.1016/j.coastaleng.2005.10.004. → pages 22

[14] K. Das, R. Janetzke, D. Basu, S. Green, and J. Stamatakos. Numerical simulations of tsunami wave generation by submarine and aerial landslides using RANS and SPH models. In *28th International Conference on Ocean, Offshore and Arctic Engineering*, Honolulu, HI, 2005. American Society of Mechanical Engineers. → pages 6, 12

[15] R. G. Dean. Relative validities of water wave theories. *Journal of the Waterways, Harbors and Coastal Engineering Division*, 96(1):105–119, 1970. ISSN 0044-8028. → pages 7

[16] R. G. Dean and R. A. Dalrymple. *Water wave mechanics for engineers and scientists*, volume 2 of *Advanced series on ocean engineering*. World Scientific, Singapore, 1991. ISBN 978-981-02-0420-4. → pages 8

[17] J. Fang, M. Parriaux, M. Rentschler, and C. Ancey. Improved SPH methods for simulating free surface flows of viscous fluids. *Applied Numerical Mathematics*, 59(2):251–271, 2009. ISSN 0168-9274. doi:10.1016/j.apnum.2008.02.003. → pages 17

[18] FlowScience. Flow-3d user manual v10.1, 2012. → pages 14, 15, 30, 43, 44

[19] H. M. Fritz. *Initial phase of landslide generated impulse waves*. PhD thesis, ETH, Zurich, CH, 2002. → pages 5

[20] H. M. Fritz, W. H. Hager, and H.-E. Minor. Lituya bay case: Rockslide impact and wace run-up. *Science of Tsunami Hazards*, 19(1):3–19, 2001. ISSN 8755-6839. → pages 5, 11

[21] H. M. Fritz, F. Mohammed, and J. Yoo. Lituya bay landslide impact generated mega-tsunami 50th anniversary. *Pure and Applied Geophysics*, 166(1):153–175, 2009. doi:10.1007/s00024-008-0435-4. → pages 5

[22] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389, 1977. ISSN 0035-8711. → pages 6

[23] M. Gomez-Gesteira, D. Cerqueiro, A. J. C. Crespo, and R. A. Dalrymple. Green water overtopping analyzed with a SPH model. *Ocean Engineering*, 32(2):223–238, 2005. ISSN 0029-8018. doi:10.1016/j.oceaneng.2004.08.003. → pages 22

[24] M. Gomez-Gesteira, A. J. C. Crespo, B. D. Rogers, R. A. Dalrymple, J. M. Dominguez, and A. Barreiro. SPHysics development of a free-surface fluid solver part 2: Efficiency and test cases. *Computers & Geosciences*, 48:300–307, 2012. ISSN 0098-3004. doi:10.1016/j.cageo.2012.02.028. → pages 15

[25] M. Gomez-Gesteira, B. D. Rogers, A. J. C. Crespo, R. A. Dalrymple, M. Narayanaswamy, and J. M. Dominguez. SPHysics development of a free-surface fluid solver part 1: Theory and formulations. *Computers & Geosciences*, 48:289–299, 2012. ISSN 0098-3004. doi:10.1016/j.cageo.2012.02.029. → pages 15, 18, 22, 23

[26] C. B. Harbitz, G. Pedersen, and B. Gjevik. Numerical simulations of large water waves due to landslides. *Journal of Hydraulic Engineering*, 119(12):1325–1342, 1993. ISSN 1943-7900. → pages 10

[27] V. Heller. *Landslide generated impulse waves: Prediction of near field characteristics*. PhD thesis, ETH, Zurich, CH, 2007. → pages 5, 7, 10

[28] V. Heller and W. H. Hager. Impulse product parameter in landslide generated impulse waves. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 136(3):145–155, 2010. ISSN 0733-950X. doi:10.1061/(ASCE)WW.1943-5460.0000037. → pages 5, 10

[29] V. Heller and W. H. Hager. Wave types of landslide generated impulse waves. *Ocean Engineering*, 38(4):630–640, 2011. ISSN 0029-8018. doi:10.1016/j.oceaneng.2010.12.010. → pages 5

[30] V. Heller and W. H. Hager. A universal parameter to predict subaerial landslide tsunamis? *Journal of Marine Science and Engineering*, 2(2):400–412, 2014. doi:10.3390/jmse2020400. → pages 5, 6, 10

[31] C. W. Hirt. What are artificial & numerical viscosities? URL http://www.flow3d.com/home/resources/cfd-101/numerical-issues/artificial-numerical-viscosities. → pages 14

[32] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981. ISSN 0021-9991. doi:10.1016/0021-9991(81)90145-5. → pages 13

[33] A. Huber. *Schwallwellen in Seen als Folge von Felssturzen*. PhD thesis, ETH, Zurich, CH, 1980. → pages 3, 5

[34] A. Huber and W. H. Hager. Forecasting impulse waves in reservoirs. In *Dix-neuvime Congrs des Grands Barrages*, pages 993–1005, Florence, 1997. Commission internationale des grands barrages. → pages 3, 4, 10

[35] J. W. Johnson and K. J. Bermel. Impulsive waves in shallow water as generated by falling weights. *Transactions, American Geophysical Union*, 30(2):223, 1949. ISSN 0002-8606. doi:10.1029/TR030i002p00223. URL http://adsabs.harvard.edu/abs/1949TrAGU..30..223J. → pages 3

[36] J. W. Kamphuis and R. J. Bowering. Impulse waves generated by landslides. *Coastal Engineering Proceedings*, 1(12), 1970. ISSN 2156-1028. doi:10.9753/icce.v12. → pages 3, 4, 5, 10

[37] G. Kim. *Numerical Simulation of Three-Dimensional Tsunami Generation by Subaerial Landslides*. PhD thesis, Texas A&M University, Collage Station, TX, 2012. → pages 11, 12, 69

[38] D. J. Korteweg and G. de Varies. On the change of form of long waves advancing in a rectangular canel, and on a new type of long stationary waves. *Philosophical Magazine Ser. 5*, 35:422–443, 1895. → pages 8

[39] P. K. Kundu, I. M. Cohen, P. S. Ayyaswamy, and H. H. Hu. *Fluid Mechanics*. Elsevier, New York, NY, 4th edition, 2008. ISBN 978-0-12-373735-9. → pages 12, 18

[40] P. K. Kundu, I. M. Cohen, and D. R. Dowling. *Fluid Mechanics*. Elsevier, New York, NY, 5th edition, 2012. ISBN 978-0-12-382100-3. → pages 12

[41] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics (Volume 6 of Course of Theorectical Physics)*. Pergamon Press, New York, NY, 2nd edition, 1986. → pages 12

[42] G. R. Liu and M. B. Liu. *Smoothed Particle Hydrodynamics: A Meashfree Particle Method*. World Scientific, New Jersey, NY, 2003. ISBN 981-238-456-1. → pages 17, 18

[43] P. L.-F. Liu, T.-R. Wu, F. Raichlen, C. E. Synolakis, and J. C. Borrero. Runup and rundown generated by three-dimensional sliding masses. *Journal of Fluid Mechanics*, 536:107–144, 2005. doi:10.1017/S0022112005004799. → pages 6, 10

[44] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal*, 82:1013–1024, 1977. ISSN 0004-6256. doi:10.1086/112164. → pages 6

[45] H. Mase. Random wave runup height on gentle slope. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 115(5):649–661, 1989. ISSN 0733-950X. doi:10.1061/(ASCE)0733-950X(1989)115:5(649). → pages 9

[46] L. Mehaute. *An introduction to hydrodynamics and water waves*. Springer Berlin Heidelberg, 1976. ISBN 978-3-642-85567-2. → pages 8

[47] D. J. Miller. Giant waves in lituya bay alaska. Geological Survey Professional Paper 354-C, United States Government Printing Office, Washington, D.C, 1960. → pages 4, 5

[48] D. Molteni and A. Colagrossi. A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH. *Computer Physics Communications*, 180(6):861–872, 2009. ISSN 0010-4655. doi:10.1016/j.cpc.2008.12.004. → pages 21

[49] J. Monaghan. On the problem of penetration in particle methods. *Journal of Computational Physics*, 82(1):1–15, 1989. ISSN 00219991. doi:10.1016/0021-9991(89)90032-6. URL http://adsabs.harvard.edu/abs/1989JCoPh..82....1M. → pages 20, 23

[50] J. J. Monaghan. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30:543–574, 1992. doi:10.1146/annurev.aa.30.090192.002551. → pages 20

[51] J. J. Monaghan. Simulating free surface flows with SPH. *Journal of Computational Physics*, 110 (2):399–406, 1994. ISSN 0021-9991. doi:10.1006/jcph.1994.1034. → pages 6, 21

[52] J. J. Monaghan. Smoothed particle hydrodynamics. *Reports on Progress in Physics*, 68(8): 1703–1759, 2005. ISSN 0034-4885, 1361-6633. doi:10.1088/0034-4885/68/8/R01. → pages 19, 22, 23

[53] J. J. Monaghan and A. Kos. Scott russell's wave generator. *Physics of Fluids*, 12(3), 2000. ISSN 10706631. → pages 6

[54] J. J. Monaghan and J. C. Lattanzio. A refined particle method for astrophysical problems. *Astronomy and Astrophysics*, 149:135–143, 1985. ISSN 0004-6361. → pages 17

[55] J. J. Monaghan, A. Kos, and N. Issa. Fluid motion generated by impact. *Journal of Waterway, Port, Coastal & Ocean Engineering*, 129(6):250–259, 2003. ISSN 0733950X. doi:10.1061/(ASCE)0733-950X(2003)129:6(250). → pages 6

[56] F. Montagna, G. Bellotti, and M. Di Risio. 3d numerical modeling of landslide-generated tsunamis around a conical island. *Natural Hazards*, 58(1):591–608, 2011. ISSN 1573-0840. doi:10.1007/s11069-010-9689-0. → pages 12

[57] E. Noda. Water waves generated by landslides. *Journal of the Waterways, Harbors and Coastal Engineering Division*, 96(4):835–855, 1970. ISSN 0044-8028. URL http://cedb.asce.org/cgi/WWWdisplay.cgi?17336. → pages 3, 4, 10

[58] G. Oger, M. Doring, B. Alessandrini, and P. Ferrant. An improved SPH method: Towards higher order convergence. *Journal of Computational Physics*, 225(2):1472–1492, 2007. doi:10.1016/j.jcp.2007.01.039. → pages 17, 48

[59] A. Panizzo. *Physical and Numerical Modelling of Subaerial Landslide Generaged Waves*. PhD thesis, University of Rome, Rome, IT, 2004. → pages 1, 8

[60] A. Panizzo, P. De Girolamo, and A. Petaccia. Forecasting impulse waves generated by subaerial landslides. *Journal of Geophysical Research*, 110, 2005. ISSN 2156-2202. doi:10.1029/2004JC002778. → pages 4

[61] L. Qiu. Two-dimensional SPH simulations of landslide-generated water waves. *Journal of Hydraulic Engineering*, 134(5):668–671, 2008. ISSN 07339429. doi:10.1061/(ASCE)0733-9429(2008)134:5(668). → pages 6

[62] P. W. Randles and L. D. Libersky. Smoothed particle hydrodynamics: Some recent improvements and applications. *Computer Methods in Applied Mechanics and Engineering*, 139(14):375–408, 1996. ISSN 0045-7825. doi:10.1016/S0045-7825(96)01090-0. → pages 21

[63] M. Robinson. *Turbulence and viscous mixing using smoothed particle hydrodynamics*. PhD., Monash University, 2009. → pages 18, 21, 47

[64] B. D. Rogers, R. A. Dalrymple, and P. K. Stansby. Simulation of caisson breakwater movement using 2-d SPH. *Journal of Hydraulic Research*, 48(sup1):135–141, 2010. ISSN 0022-1686. doi:10.1080/00221686.2010.9641254. → pages 22

[65] J. S. Russell. *Report on waves: made to the meetings of the British Association in 1842-43*. 1845. → pages 2, 7

[66] H. M. Schey. *Div, Grad, Curl, and All That: An Informal Text on Vector Calculus*. W. W. Norton & Company, New York, NY, 4th edition, 2005. ISBN 0-393-92516-1. → pages 18

[67] R. Slingerland and B. Voight. Occurrences, properties and predictive models of landslide generated impulse waves. In *Rockslides and avalanches 2*, pages 317–397. 1979. → pages 2

[68] R. Slingerland and B. Voight. Evaluating hazard of landslide-induced water waves. *Journal of the Waterway Port Coastal and Ocean Division*, 108(4):504–512, 1982. ISSN 0148-9895. → pages 4, 10

[69] R. M. Sorensen. *Basic Coastal Engineering*. Springer, New York, NY, 3rd edition, 2006. ISBN 0-387-23332-6. → pages 2, 8

[70] P. St-Germain. *Numerical Modeling of Tsunami-Induced Hydrodynamic Forces on Free-Standing Structures using the SPH Method*. PhD thesis, University of Ottawa, Ottawa, ON, 2012. → pages 22

[71] G. G. Stokes. On the theory of oscillatory waves. *Transactions of the Cambridge Philosophical Society*, 8:441–455, 1847. → pages 8

[72] C. E. Synolakis. The runup of solitary waves. *Journal of Fluid Mechanics*, 185:523–545, 1987. doi:10.1017/S002211208700329X. → pages 9

[73] D. J. Tritton. *Physical Fluid Dynamics*. Oxford University Press, Oxford, UK, 2nd edition, 1988. ISBN 0-19-854493-6. → pages 12

[74] F. Ursell. The long-wave paradox in the theory of gravity waves. *Mathematical Proceedings of the Cambridge Philosophical Society*, 49(04):685–694, 1953. doi:10.1017/S0305004100028887. → pages 8

[75] USACE. *Shore protection manual: Volume 1*, volume 1. United States Government Printing Office, Washington, D.C, 4th edition, 1984. → pages 9

[76] USACE. *Shore protection manual: Volume 2*, volume 2. United States Government Printing Office, Washington, D.C, 4th edition, 1984. → pages 9

[77] L. Verlet. Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Physical Review*, 159(1):98–103, 1967. doi:10.1103/PhysRev.159.98. → pages 22

[78] H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. 2nd edition, 2007. ISBN 978-0-13-127498-3. → pages 13

[79] S. Viroulet, D. Cebron, O. Kimmoun, and C. Kharif. Evolution of water waves generated by subaerial solid landslide. In *Proceedings of the 27th International Workshop on Water Waves and Floating Bodies*, page 4, Copenhagen, Denmark, 2012. International Workshop on Water Waves and Floating Bodies. → pages 6

[80] S. J. Walder, P. Watts, O. E. Sorensen, and K. Janssen. Tsunamis generated by subaerial mass flows. *Journal of Geophysical Research*, 108, 2003. → pages 4, 5

[81] WCHL. Hydraulic model studies - wave action generated by slides into mica reservoir - british columbia, 1970. → pages 4

[82] H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4(1):389–396, 1995. ISSN 1019-7168, 1572-9044. doi:10.1007/BF02123482. → pages 17

[83] R. L. Wiegel. A presentation of cnoidal wave theory for practical application. *Journal of Fluid Mechanics*, 7(02):273–286, 1960. doi:10.1017/S0022112060001481. → pages 8

[84] R. L. Wiegel, E. K. Noda, E. M. Kuba, D. M. Gee, and G. F. Tornberg. Water waves generated by landslides in reservoirs. *Journal of the Waterways, Harbors and Coastal Engineering Division*, 96(2):307–333, 1970. ISSN 0044-8028. URL http://cedb.asce.org/cgi/WWWdisplay.cgi?17311. → pages 3, 4, 10

[85] A. Zweifel, W. H. Hager, and H.-E. Minor. Plane impulse waves in reservoirs. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 132(5):358–368, 2006. ISSN 0733-950X. doi:10.1061/(ASCE)0733-950X(2006)132:5(358). → pages 5