A User Study: Is the Advection Step in Shallow Water Equations Really Necessary?

T. Kellomäki and T. Saari

Department of Pervasive Computing, Tampere University of Technology, Finland

Abstract

Heightfield methods, such as the pipe method and shallow water equations (SWE), have often been used to simulate large areas of water. Of these, the SWE are often preferred due to being more realistic, but they are also more complex and demand more computational resources than the pipe method. These two methods were presented to over 40 subjects in both a gaming and a video context to see whether they report noticing the advantages of SWE compared to the pipe method. No significant differences were observed in any of the categories measured (hedonic valence, flow, spatial presence, realism). Therefore, at least considering using the pipe method instead of the SWE is recommended. Also, varying the time step between 5 and 20 ms did not affect the user experience.

Categories and Subject Descriptors (according to ACM CCS):
I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically Based Modeling K.8.0 [Personal Computing]: General—Games

1. Introduction

Water is an important part of nature. Virtual worlds and games are becoming more interactive and could benefit from dynamic water behaviour, e.g., a river changing its course due to player actions. However, the needed water simulation is complex to implement and computationally taxing in the real-time environment. The 3D solutions used in engineering are still out of reach for large-scale real-time environments. A solution is to use a 2D heightfield simulation.

This paper concentrates on two heightfield methods, *the pipe method* and *the shallow water equations* (SWE). To our knowledge, these two related methods have never been compared in an empirical study. To measure the quality of experience achieved by each method, a game utilising the two methods was implemented (Fig. 1). The *hedonic valence*, *flow*, *spatial presence*, and *sense of realism* induced by the game experience were measured using self-report scales. The subjects additionally watched videos, which allowed them to fully concentrate on the water simulation.

Section 2 gives background on water simulation. In Section 3, the methodology is presented. Section 4 describes the implementation of the water simulation methods. The results can be found in section 5. Section 6 concludes the paper.

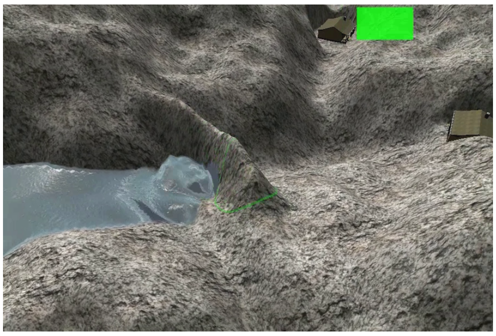

Figure 1: *A screenshot from the water game shows flow being blocked by raising terrain.*

2. Water simulation methods

3D Eulerian simulations are realistic, but too slow for most games. Faster approaches include Lagrangian methods, such as smoothed-particle hydrodynamics (SPH), and heightfield methods. SPH is suited for small amounts of water, while heightfields simulate rivers and lakes well. [Kel12]

The FFT-based method of Tessendorf [Tes99] and wave particles of Yuksel [YHK07] have already been used in

DOI: 10.2312/egsh.20141010

games, such as *Just Cause 2* and *Uncharted 3*. These methods simulate open waters well, but the interaction between water and terrain is limited, and water cannot flow into new areas. Tall cells [CM11] are an interesting hybrid 2D/3D method, but not yet fast enough for most games [Kel12].

Various versions of the shallow water equations (SWE) have been used by researchers (*e.g.*, [LvdP02, CM10, TMFSG07, SBC*11]). The SWE are derived from the *Navier-Stokes equations* by utilising several simplifying assumptions [LvdP02]. The result is simple and fast. The most complex part is the advection, for which a common, stable solution is the semi-Lagrangian method [Sta99].

If the self-advection term is dropped from the SWE, a linearised set of equations corresponding to the wave equation is reached [KM90, Eqs. 1–4]. Kass and Miller write that the simplification is used often, but the origin of the idea is unclear. They also point out that the result is too simple for engineering, but adequate for animation. O'Brien and Hodgins use hydrostatic pressure to derive an equivalent but perhaps more intuitive model [OH95]. The result is an algorithm that is often referred to as the pipe method. It has been reused *e.g.*, by Mei *et al.* to simulate hydraulic erosion [MDH07]. Here the term "pipe method" refers to the whole class of linearised models, regardless of how they are reached.

The advection step of SWE creates whirlpools and other realistic but subtle effects (see Fig 2 for an example), but also causes numerical dissipation that makes the water more viscous and thus less lively [BMF07]. Some researchers have dismissed the pipe method as too simple. Layton and van de Panne solve the full SWE and say, "our solution should be more physically accurate than that of, say, Kass and Miller (1990)" [LvdP02]. Both Chentanez and Müller [CM10], and Solenthaler *et al.* [SBC*11] name lack of vortices as the main drawback of the pipe method. However, we are not aware of any previous work to compare the methods.

In this paper the full SWE are compared to a version without the advection step, which represents the pipe method. This procedure makes sure that any differences are due to the advection step and not caused by implementation details, which maximises the fairness of the comparison.

3. Methodology and procedure

The potential application of water simulation in games directed the selection of the psychological impacts that were measured. Hedonic valence (e.g. pleasure) was selected, because games are often played for psychological gratifications such as fun and enjoyment [BFC*09]. It is also believed that hedonic valence is positively correlated with the feelings of flow and spatial presence [WW11]. Spatial presence is an index of immersion and could also be seen as a psychological gratification goal of virtual worlds in itself. The subjects were also asked directly how realistic they found the different methods. Self-report metrics were used.

Figure 2: *A detail from a video. Without advection (top), a small rapid stream coming from left does not cause any noticeable effects on the surface of the main stream. With advection (bottom), an interesting whirlpool is created.*

58 subjects (55 male) participated in the experiment as a part of their master level game programming course in a university of technology. The population is heavily biased toward experienced gamers, though this was not measured. The average age was 24.8 (SD 4.2). Some subjects were dropped from the data for various reasons, such as disturbing environment noises during the experiment. 41 subjects were finally used for the games and 48 for the videos. The experiment was conducted in a laboratory. A laptop running the simulation was connected to a 24" monitor, keyboard and mouse. Another computer was used to answer the questionnaires repeated after each condition. All stimuli were fullscreen with a resolution of 1920 by 1080 pixels at 60 Hz.

The game was played on a 256 by 256 heightfield with a simulation resolution of 1 meter. Water was continuously created at given areas of the levels. The goal was to protect houses from flooding by lowering or raising the terrain. The game was designed so that the users had to spend most of the game observing instead of altering the terrain because of limited energy given for the changes. The game had three levels with varying geometry, number of houses, and number of water sources. Each game lasted 1 to 3 minutes and ended when at least one of the houses had flooded. The players were given medals for surviving long enough. The accompanying video includes gameplay footage.

Each subject first practised the interface and gameplay by playing each level twice. The measured games then consisted of subjects playing each level once with each simulation method (3×2 games). For each level, the order of simulation methods was random. After each game, a survey with 25 questions about the experience was answered: a single item for hedonic valence (self-assessment manekin scale, SAM [BL94]), 11 items for testing spatial presence (MEC SPQ [VWG*04]), 9 for flow [BFC*09] and 3 for realism). The spatial presence and flow were measured using shortened scales with the most representative items selected. An example question of the realism scale is "I think that the water in the game behaved realistically."

After the games, the subjects watched 12 videos in a completely random order, each 30–60 seconds in length. These consisted of 6 different scenes, each simulated using both methods. A similar survey with 21 items was answered (1 item for hedonic valence, 11 spatial presence, 6 flow, 3 realism). Finally, a background survey was answered.

As a secondary experiment, the time step of the simulation was randomly varied between 5 and 20 ms using an uniform distribution. The range was limited by computer resources on the lower end. The higher end was selected to keep the simulation absolutely stable, since instabilities would probably have affected the main experiment too much.

4. Water simulation implementation

The implementation of the simulation is heavily based on [CM10]. The height and velocity integration steps are identical to their description, except that to enhance stability, a friction parameter as in [MY97, Eq. 4] was added. After these steps the pipe method is reached, as explained in Section 2. For the full SWE, a standard advection step using the semi-Lagrangian method [Sta99] is added. The modified McCormack method used by Chentanez and Müller [CM10] was also tried, but no noticeable difference was seen. Reflecting border conditions were used.

The simulation is implemented using OpenGL shaders and is bound by texture bandwidth. Adding the advection step adds about 38% to the average simulation execution time (averaging over several runs with different scenes). Code complexity is difficult to measure sensibly, but to give a very rough idea, adding the advection adds 72% to lines of code in our implementation of the simulation shaders. An important benefit of the simpler method is the ease of extending and customising the method, which seems important for tinkering game developers, who often want to break the laws of physics for the sake of gameplay.

To focus the subjects' attention to the simulation, the visualisation was bare. The water was rendered using a reflection from the skybox and a Fresnel term, while two normal maps with alternating weights were advected according to the flow, similarly to [Vla10]. This was necessary to visualise the vortices added by the advection step. The visualisation was identical for both simulation methods.

Table 1: *The p-values of the effect of the simulation method on the measured responses*

Response	Games	Videos
Hedonic valence	0.886	0.246
Spatial presence	0.826	0.366
Flow	0.538	0.127
Realism	0.110	0.243

5. Results

ANOVA was used to find the effect of the simulation method on the measured responses (hedonic valence, spatial presence, flow, and realism). The factors in the model were the simulation method (SWE or pipe), the ID of the game level played (or the ID of the video shown) and the interaction of the two. A total of 8 separate ANOVA analyses were done. Repeated measures ANOVA was used to gain additional statistical power due to the fact that each subject played six times and watched 12 videos. Since the order of simulation methods was randomised, no significant ordering bias is expected to be present.

The interaction terms are not reported, because none of them were statistically significant. Additionally, while the IDs had a significant effect on many of the responses, they only represent randomly selected game levels and videos and are also not presented here. The interesting effects are those of the simulation method on each of the responses. The p-values of the ANOVA F-tests are given in Table 1 (small values would be evidence for the claim that the simulation methods have different effects on the associated response). To summarise the findings, the simulation method had no statistically significant effect on any of the measured responses even at the 95% significance level. In realism, where the difference was closest to being significant, the pipe method actually performed better than the SWE.

The Greenhouse–Geisser correction was used wherever sphericity was violated, but this was not needed for the reported results concerning the effects of the simulation method.

Additionally, the effect of the time step was analysed independently of the other factors using multiple regression with time step, method, and the interaction of those as the independent variables. According to the t-tests, the time step coefficients were not statistically significantly different (on the 95% level) from zero for any of the measured responses in either the games or the videos, and are not reproduced here. It is concluded that on the given range, the time step had no significant effect on any of the responses.

6. Conclusion

The subjects played a game where the water simulation method was randomly selected among SWE and the pipe method. None of the self-reported experiences of hedonic valence, flow, spatial presence, or realism were significantly affected by the method. There was no significant difference when watching videos of the two methods in action, either. Therefore, the common assumption that users appreciate the vortices and other effects created by advection does not receive any support. It is possible that the loss of liveliness due to the added advection step counterbalanced the more realistic phenomena achieved by SWE.

Both the simulation methods and the visualisation allow the tweaking of many details and parameters, which radically affect how natural the water looks. Therefore it is very difficult to compare the appearance of two different methods reliably, since these implementation details could affect different methods in different ways. The methods were made as similar as possible by using a version of the pipe method that is simply implemented by omitting the advection step from the SWE simulation.

The time step in the range of 5 to 20 ms did not have a significant effect on the responses. It could be interesting to see whether even longer time steps (causing visible instabilities) could be used without hurting the user experience.

One limitation in the experiment was that only a single type of game was employed. However, the gameplay was designed so that the subjects would see a lot of water and needed to concentrate on its flow. In addition, in many of the videos the camera was close to the water and the subjects devoted their full attention to the water. It is probable that in most games and virtual worlds, there is even less need for the water to behave realistically.

Only a single kind of visualisation, a single simulation resolution and two methods were examined. Studies with more advanced simulation (e.g., tall cells) and different visualisation methods (e.g., particles for splashes, foam) included could be useful ideas for future work. The differences between SPH and the grid-based methods are especially interesting, but making such a comparison fair is difficult due to differences in visualisation methods.

Based on the evidence, it is recommended to at least consider using the simpler and faster pipe method instead of the full shallow water equations in games and other virtual worlds, where strict realism is not needed. It can be as easy as dropping the advection step.

References

[BFC*09] Brockmyer J. H., Fox C. M., Curtiss K. A., McBroom E., Burkhart K. M., Pidruzny J. N.: The development of the game engagement questionnaire: A measure of engagement in video game-playing. *Journal of Experimental Social Psychology 45*, 4 (2009), 624–634. 2, 3

[BL94] Bradley M. M., Lang P. J.: Measuring emotion: The self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry 25*, 1 (1994), 49–59. 3

[BMF07] Bridson R., Müller-Fischer M.: Fluid simulation: SIGGRAPH 2007 course notes. In *ACM SIGGRAPH 2007 courses* (2007), SIGGRAPH '07, ACM, pp. 1–81. 2

[CM10] Chentanez N., Müller M.: Real-time simulation of large bodies of water with small scale details. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2010), SCA '10, Eurographics Association, pp. 197–206. 2, 3

[CM11] Chentanez N., Müller M.: Real-time Eulerian water simulation using a restricted tall cell grid. In *ACM SIGGRAPH 2011 papers* (2011), SIGGRAPH '11, ACM, pp. 82:1–82:10. 2

[Kel12] Kellomäki T.: Water simulation methods for games: a comparison. In *Proceedings of the 16th International Academic MindTrek Conference* (2012), ACM, pp. 10–14. 1, 2

[KM90] Kass M., Miller G.: Rapid, stable fluid dynamics for computer graphics. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques* (1990), SIGGRAPH '90, ACM, pp. 49–57. 2

[LvdP02] Layton A., van de Panne M.: A numerically efficient and stable algorithm for animating water waves. *The Visual Computer 18*, 1 (2002), 41–53. 2

[MDH07] Mei X., Decaudin P., Hu B.-G.: Fast hydraulic erosion simulation and visualization on GPU. In *15th Pacific Conference on Computer Graphics and Applications, 2007. PG '07* (Nov 2007), pp. 47–56. 2

[MY97] Mould D., Yang Y.-H.: Modeling water for computer graphics. *Computers and Graphics 21*, 6 (1997), 801 – 814. Graphics in Electronic Printing and Publishing. 3

[OH95] O'Brien J. F., Hodgins J. K.: Dynamic simulation of splashing fluids. In *Proceedings of Computer Animation '95* (Apr 1995), pp. 198–205. 2

[SBC*11] Solenthaler B., Bucher P., Chentanez N., Müler M., Gross M.: SPH based shallow water simulation. In *Workshop in Virtual Reality Interactions and Physical Simulation* (2011), pp. 30–46. 2

[Sta99] Stam J.: Stable fluids. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (1999), ACM Press/Addison-Wesley Publishing Co., pp. 121–128. 2, 3

[Tes99] Tessendorf J.: Simulating ocean water. In *SIGGRAPH course notes* (1999). 1

[TMFSG07] Thürey N., Müller-Fischer M., Schirm S., Gross M.: Real-time breaking waves for shallow water simulations. In *Proceedings of 15th Pacific Conference on Computer Graphics and Applications* (2007), PG '07, pp. 39–46. 2

[Vla10] Vlachos A.: Water flow in Portal 2. In *ACM SIGGRAPH 2010 courses* (2010), ACM, pp. 1–54. 3

[VWG*04] Vorderer P., Wirth W., Gouveia F. R., Biocca F., Saari T., et al.: Mec spatial presence questionnaire (mec-spq): Short documentation and instructions for application. *Report to the European Community, Project Presence: MEC (IST-2001-37661)* (2004). 3

[WW11] Weibel D., Wissmath B.: Immersion in computer games: The role of spatial presence and flow. *International Journal of Computer Games Technology 2011* (2011), 6. 2

[YHK07] Yuksel C., House D. H., Keyser J.: Wave particles. *ACM Trans. Graph. 26*, 3 (Jul 2007). 1