Awash: Modeling Wave Movement in a Ripple Tank

New Mexico
Supercomputing Challenge

Final Report
April 4, 2007

Team 62
McCurdy School

Team Members

Sarah Armstrong
Joseph Koby
Juan-Antonio Vigil
Vanessa Trujillo

Teacher

Robert Dryja

Project Mentors

Bob Roby
Nick Bennett

Table of Contents

Executive Summary / Problem Statement

The team is creating a numerical simulation to model the behavior of waves in a liquid. This model is then used to predict the behavior of waves observed under controlled conditions. Our program is written in FORTRAN due to prior experience with the language, and familiarity with Finite Difference modeling of fluid dynamics.

Methods used

1.0 Mathematics

1.1 Objectives

The mathematical portion of this project is concerned with shallow water equations. Shallow water equations are a collection of equations used to explain the flow of a fluid below a horizontal surface pressure. The word shallow refers to the depth of the water in comparison with the horizontal length of the water.

The flow explained by these equations is the horizontal flow caused by changes in the height of the pressure surface of a fluid. Height cannot be taken into account with shallow water equations because they only have one vertical level. Shallow water equations are ideal for describing the flow of a fluid in real life situations such as: rivers, oceans, ripple tanks, atmospheric waves, gravity waves,

1.2 Shallow Water Equations

The two shallow water equations we will be dealing with for our project are: The Conservation of Momentum and the Conservation of Mass. (The Conservation of Energy is irrelevant because our model is incompressible and isothermal.)

The Conservation of Momentum states that the momentum of a set of objects, that are unaffected by external forces, is constant, it can neither be created nor destroyed. Momentum is the mass of an object multiplied by the velocity of the object. This momentum can even be conserved during and after a collision. Since the momentum is conserved, it can be used to determine the unknown velocity following a collision.

The Conservation of Mass states that mass can neither be created nor destroyed. The mass of an item can be calculated by multiplying the volume of the item by the density of the item. In fluid dynamics, if the density of the item is constant, the Conservation of Mass can be used to determine the velocity of the flow of fluid.

1.3 Definitions:

mv = momentum

p = pressure

v = velocity (y axis)

t = time

h = height

d = density

x = distance

□ anḍΔ = the change or variation of a quantity\

Derivation of Shallow Water Equations

Conservation of Mass (Continuity)

1.4 Equations

$$\frac{\delta\, m}{\delta\, t} + \frac{\delta(v\, m)}{\delta\, x} = 0$$

original equation

$$\frac{\Delta\, m}{\Delta\, t} + \frac{\Delta\,(m\, v\,)}{\Delta\, x}$$

change δ to Δ

$$\frac{\Delta\,(\,\rho\ \ h\ \ w\, d\,)}{\Delta\, t} + \frac{\Delta\,(\rho\, h\, w\ \ d\ \ v\,)}{\Delta\, x} = 0$$

expand *m* and *mv* to
all variables

$$\frac{\Delta\ \ h}{\Delta\, t} + \frac{\Delta\,(h\, v\,)}{\Delta\, x}$$

cancel because constant

$$\Delta\ \ h\ + \underline{\Delta\,(h\, v\,)\Delta\, t}\ = 0$$

Δx

$$m_2 - m_1 = h_2 - h_1 = - \Delta (h v) \frac{\Delta t}{\Delta x}$$

$$h_2 = h_1 - \frac{\Delta t}{\Delta x} \Delta (h v)$$

$$h_2 = h_1 - \frac{\Delta t}{\Delta x} \Delta (h_2 v_2 - h_1 v_1)$$

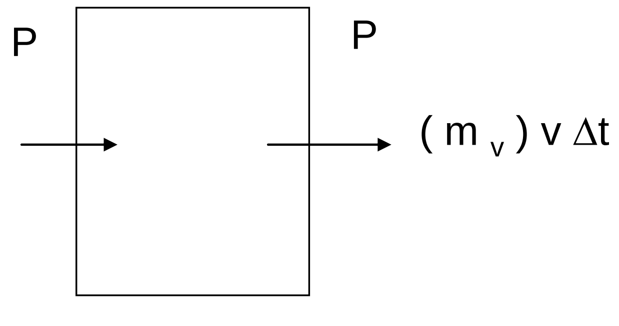
Conservation of Momentum

$$\frac{\delta (m v)}{\delta t} + \frac{\delta (m v^2)}{\delta x} + \frac{\delta \rho}{\delta x} = 0$$

$$\frac{\delta (h v)}{\delta t} + \frac{\delta (h v^2)}{\delta x} + \frac{\delta \rho}{\delta x} = 0$$

STATE **FLUX**

$$\begin{bmatrix} h \\ h v \end{bmatrix}_t + \begin{bmatrix} h v \\ v2 + \tfrac{1}{2} g h2 \end{bmatrix}_x = 0$$



2.0 Computer Modeling

2.1 Objectives

The computer model is a simple one-dimensional numerical simulation written in FORTRAN. The program is designed to approximate the behavior of a single wave in a tank filled with water. This approximation is calculated using a method called the Lax-Wendroff two step, which is one of several systems of Shallow Water Equations. This method uses both a predicting step, and a correcting step to increase the accuracy of the calculations. Specifically, the purpose of the model is to predict the activity of waves based on their properties. For instance the model can simulate waves based on mass distribution (a low amplitude wave with a longer wavelength, or a high amplitude wave with a short wavelength.) The effect of different amounts of momentum applied to the wave can also be simulated (waves can be driven solely by gravity, or forcefully pushed through the water.) Finally, the model can simulate the first few seconds of collisions between two waves of differing masses and velocities.

2.2 Structure of Code

The current version of the code, for the simulation is fairly straightforward and can be broken up into five sections. The first section is where all variables are defined, and is also where an input file is read by a subroutine in order to define the constants (the size of the tank, the length of the time steps, the wave properties, etc.) Additionally, the model's domain (the tank) is broken into a specified number of zones. The second section is where the variables are initialized. First, all of the arrays used to calculate mass and momentum are set to zero. Next the initial conditions (such as amount of water in the tank and the properties of the wave being simulated) are defined. Then the output routine gets set up; this part tells the program what data to put into an Excel spreadsheet for analysis, and how to organize it. Finally a few simple equations are defined to check for any mass or momentum conservation errors the program could potentially make.

The third section is where all the calculations for the simulation take place. Everything in this section is contained in one very large do loop, which is cycled once for each time step. The first part of this loop has two counting variables which keep track of the time elapsed, and thus send data to the output file at regular intervals. Next, the boundary conditions are defined. In this simulation, the boundary conditions place a wall at each end of the domain. This is done by creating a ghost cell at each end of the domain, and setting these ghost cells such that:

$$\text{Mass}_0 = \text{Mass}_1,\ \text{Mass}_{\text{jbar}+1} = \text{Mass}_{\text{jbar}},\ \text{and}$$

$$\text{Momentum}_0 = -\text{Momentum}_1,\ \text{Momentum}_{\text{jbar}+1} = -\text{Momentum}_{\text{jbar}}$$

where jbar is the total number of zones in the virtual tank. This procedure places an equal amount of mass moving in the opposite direction at each end (just like a wall pushes back with equal force in the opposite direction.) After this is finished, the predicting step of the Lax-Wendroff is calculated. The predictor step is offset by a factor of 0.5 relative to the corrector step. The corrector step is calculated immediately afterwards, and uses the answer from the predictor step to calculate the final answer for the current time interval. A representation of how these steps are offset is shown in figure 0.1.

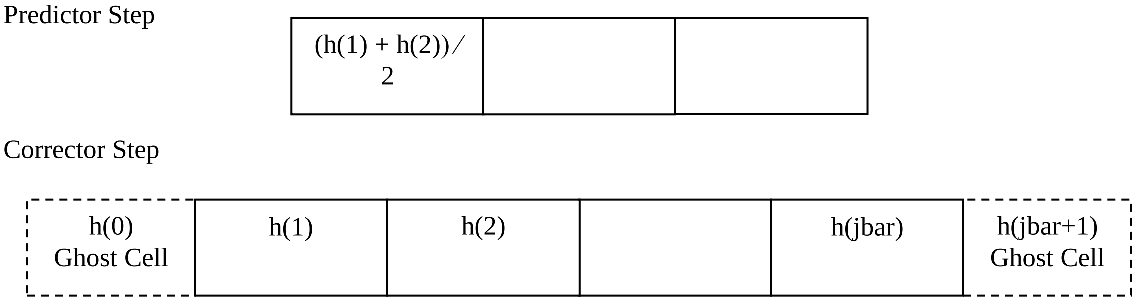


Fig. 0.1

Both sets of equations use their answers from the previous zone to calculate results in the next zone. Therefore, in the fourth section the answers from the Lax-Wendroff equations are used to update the "old" mass and momentum figures for the next series of calculations. After the program does this, it tests for any mass conservation errors by using the equations defined earlier (in the second section.) If the print interval has been reached, the mass figures from the current time step are printed into an array labeled "output." If the stop time has been reached, the program stops calculating and continues to the fifth section. In this section, all of the answers in the output array are written into an output file in Excel. This file can then be used to generate graphs for visualizing the data. After the file is written to, the program closes the input and output files and terminates.

2.3 Development

The first version of the code is fairly primitive to make sure it works correctly. The biggest difference between this model and the current version is the initial conditions. The early code used a dam break to generate a wave. After experiencing the teething problems typical of a newly written code, the model functioned as expected. But the biggest pitfall for this version is the fact that the Lax-Wendroff method has an inherent instability that occurs whenever there is a sharp discontinuity, such as in Figure 0.2.

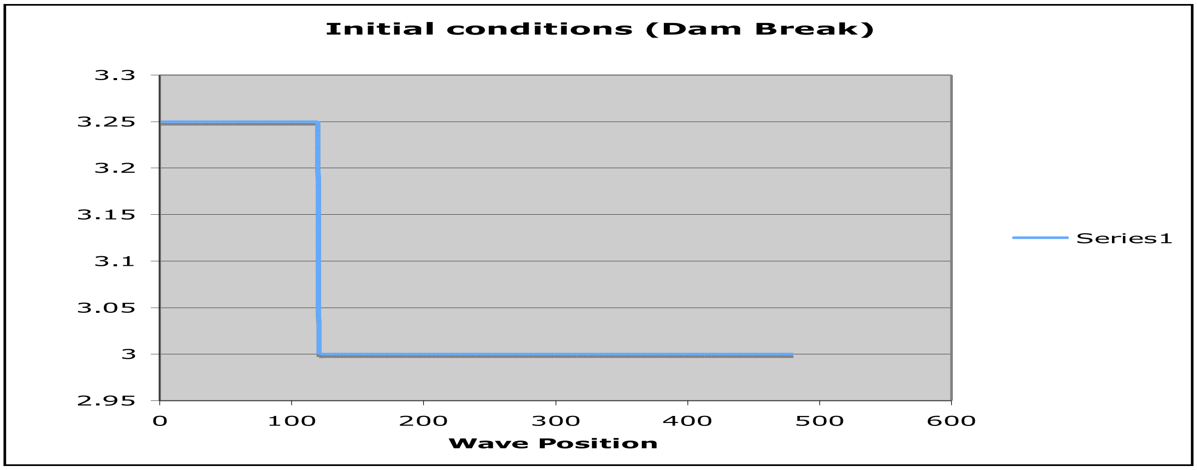


Fig. 0.2

To counter this effect, the raised portion of water in the final version of the code is curved using the following equation:

$$\text{Mass}=d + h \cdot \cos(4\pi \cdot (j/2)/(\text{jbar}))$$

Fig. 0.3

where d is equal to the water depth, h is equal to the wave mass, j equals the number of a specific zone in the domain, and jbar equals the total number of zones. By setting the initial conditions in this manner there are no sudden discontinuities and therefore, no instability in the result.

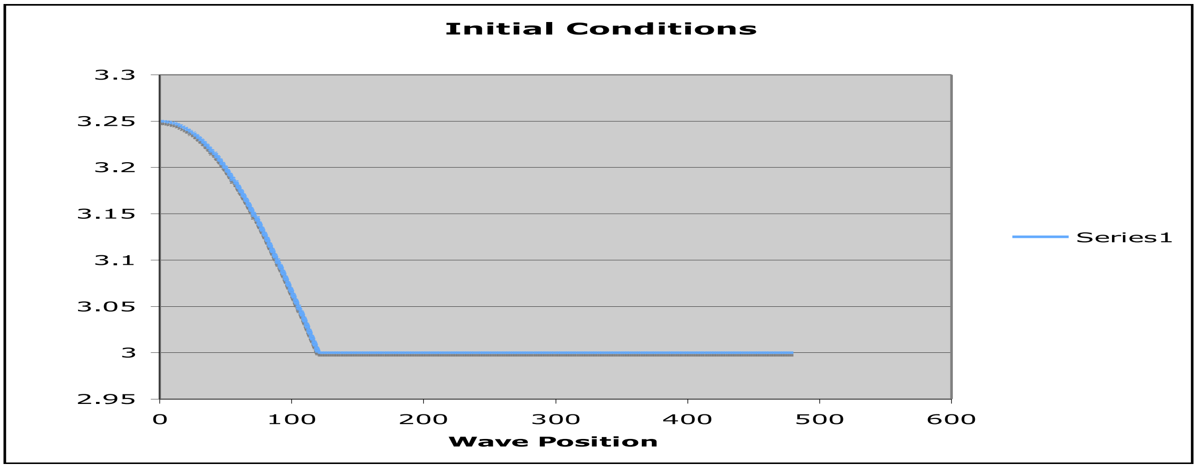


Fig. 0.4

2.4 Results and Limitations

It should be noted however, that there are limitations to how extreme these properties can be defined before the numbers go unstable. The ideal initial conditions are a wave mass between 0.1 and 0.25 units, and a length of twelve inches (1⁄4 of the length of the tank, which measures 4 feet.) If any momentum is applied in these initial conditions, it needs to be kept below 0.5 meters per second. If these conditions are exceeded, a numerical instability will result. This is likely due to the absence of additional equations to calculate viscosity, friction, and other real-world forces that act upon a wave in a fluid. Although this compromises some elements of the simulation, the code was created with simplicity in mind and has worked sufficiently for the research being conducted. Furthermore when enough force is applied to a wave, it breaks. This may be an additional source contributing to numerical instabilities.

The results obtained from the simulations consistently show that a wave generated in a tank four feet long will take approximately three seconds to travel from one end of the tank to the other. Another characteristic that stands out is how the wave builds up considerably when it reflects off the walls of the tank. Another item of interest is how the front portion of the simulated wave tends to be steeper than the trailing portion, because the water in front of the wave is at rest. There is also a small oscillation at the very back of the wave, which is observable as turbulence behind real waves.

2.5 Validation

The results produced by the math based computer model have been compared to behavior observed firsthand in experiments using a real wave tank. Like in the model, the real tank is four feet in length. Surprisingly, the computer simulation has been far more accurate than originally expected. A real wave takes almost exactly three seconds to travel the length of the tank.

Additionally, the water almost splashes out of the tank when a fairly small wave is reflected off one of the walls (another similarity to the model.)

Additional work that can be done with minimal additions to the code includes modeling waves in thicker fluids than water, and modeling waves in tanks considerably longer than four feet. With more extensive modifications, the code could possibly model a breaking wave, and multiple waves chaotically interacting (although a multi-dimensional simulation would be the ideal way to do this.) A very early version of the program was adequate for the planned research, but this version has very high potential for modification.

3.0 The Engineering

3.1 Physical Model

The physical model is very straight forward. The team built an 8'x24'x48' tank out of ¼' Plexiglas. Not knowing how much weight the Plexiglas would hold, the team initially planned to build the tank inside a wooden frame. David Olivas helped us build the tank. The tank is held together with screws and 100% silicon sealant. When the tank was finished, it looked strong enough without a frame, so it was left alone. However, there was a stand built for it. The team used a 48' long 2x12 on one side and three 24' long 2x12s on the ends and the middle. We decided to build a stand so light would go through the tank, so the waves could reflect on the table beneath it. The tank was also built to put a projector underneath it so the waves could be projected on the wall. When the tank was first filled there were some leaks on the bottom corners. The team then fixed it with silicon, but there were still some other leaks because one side didn’t have anything to support the entire edge. So a long piece if 1x2 was placed to fix the problem.

3.2 The Generators

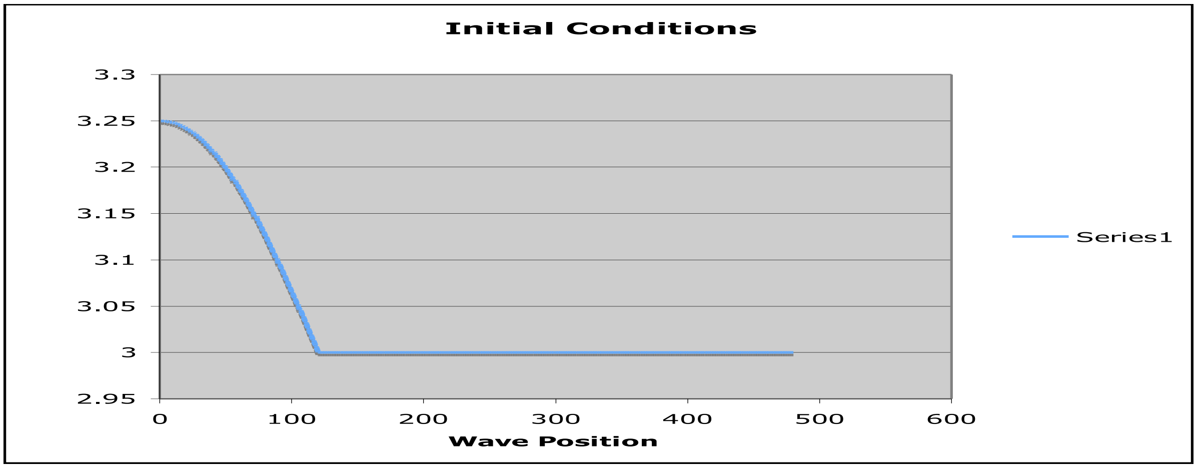
Tow different generators to generate our waves were used.. The first is a motor with an offset weight on it. The motor is connected to a 1x2 with two small spheres that vibrate in the water to create waves. The other is an oscillator that creates larger waves. The motor is connected to a bar that will make another bar with a threaded rod rise up and down. Attached to the threaded rod is a small plastic bar that created the waves.

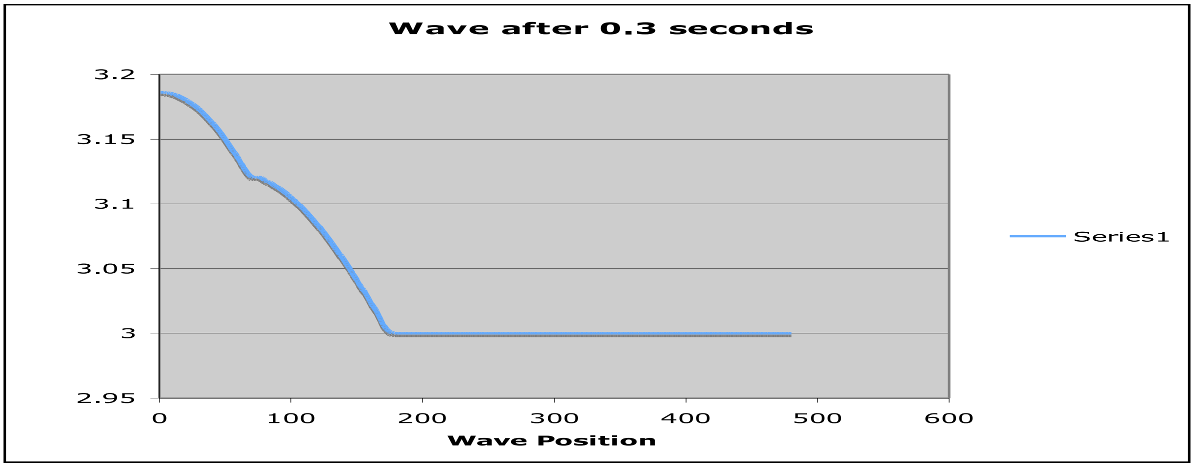
3.3 Experimentation

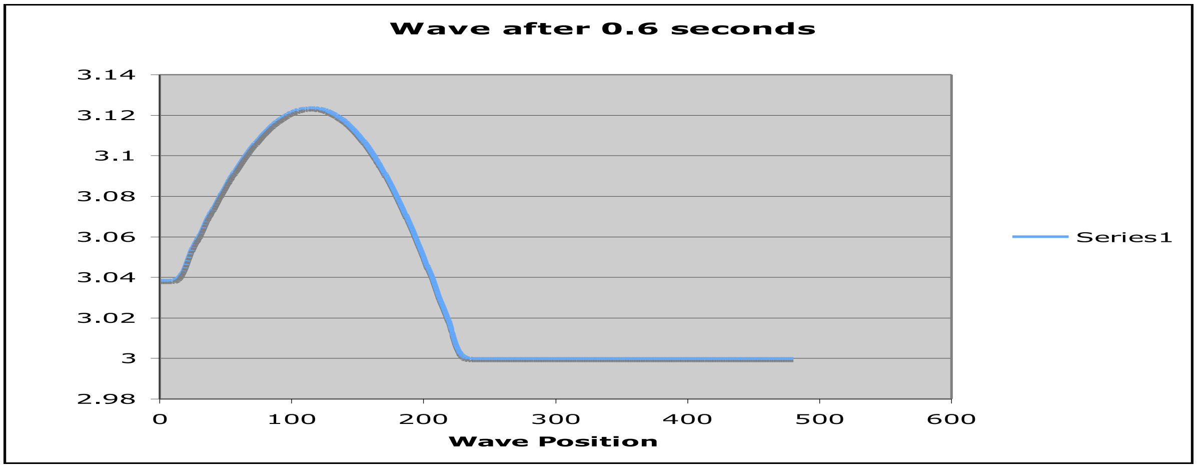
The tank was filled with about four inches of water to generate waves. When filming was attempted with two different cameras, the cameras were focusing on the wall behind the tank, so the waves were not visible. Black paper was taped behind the tank, but the cameras were only showing a reflection of the room behind them. Finally it was decided to put a layer of vegetable oil in the surface of the water, turn off the lights, and turn on a UV light. The UV light made the oil fluoresce light green. Due to the glow, the cameras had a distinct image to focus on.

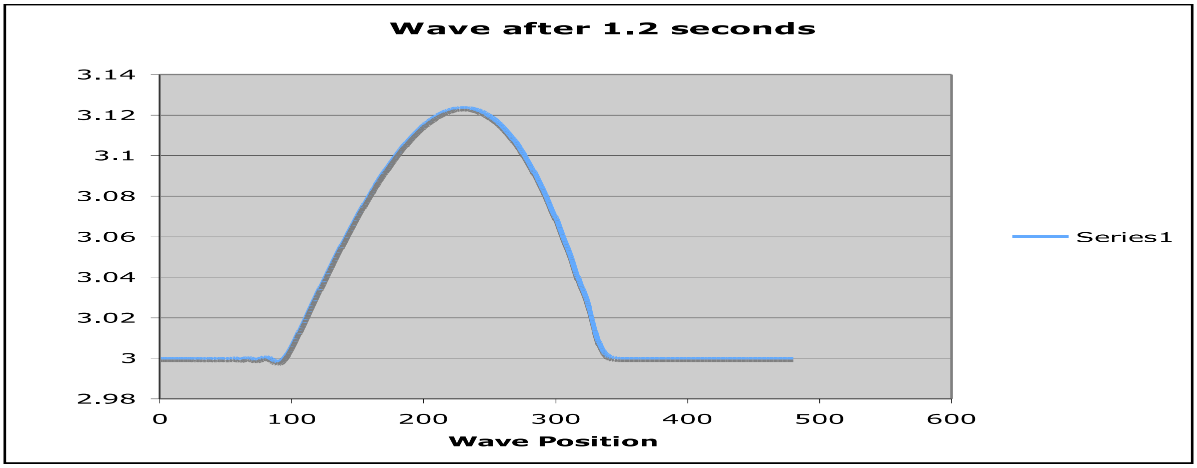
Software, References, and Tables

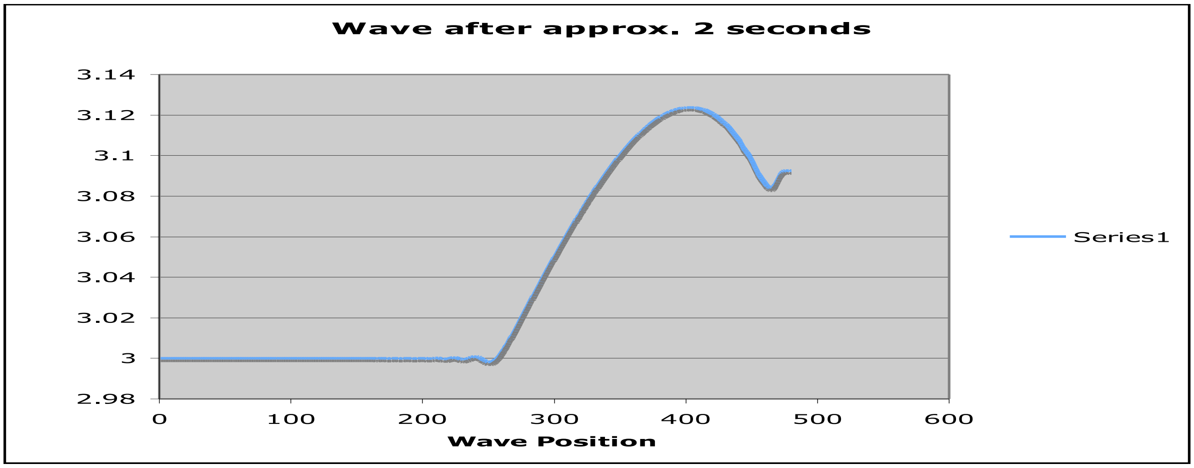
The Following graphs are generated by the team’s one-dimensional Lax-Wendroff Simulation. Note that the X-axis is wave mass (due to technical difficulty, the label didn’t print.)

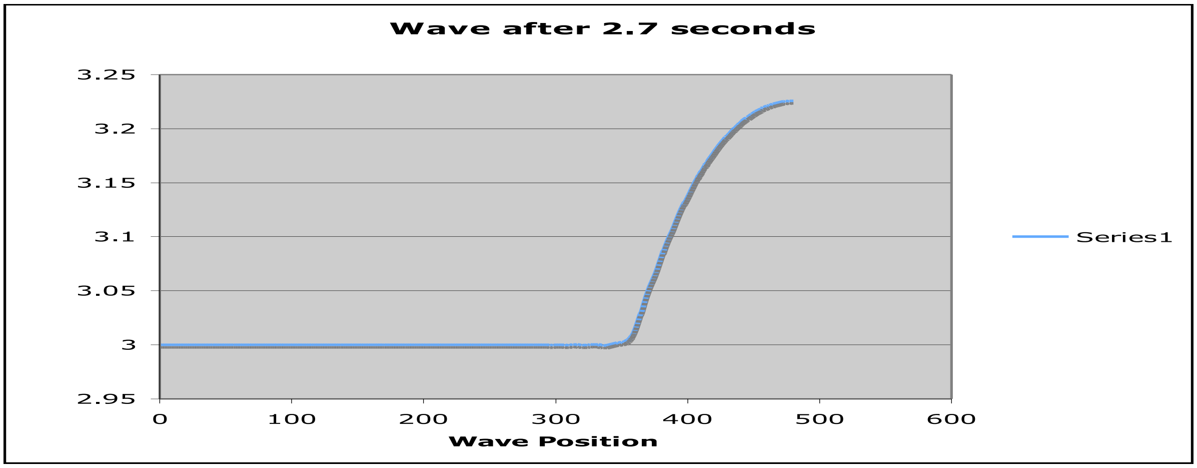

Initial Conditions
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1

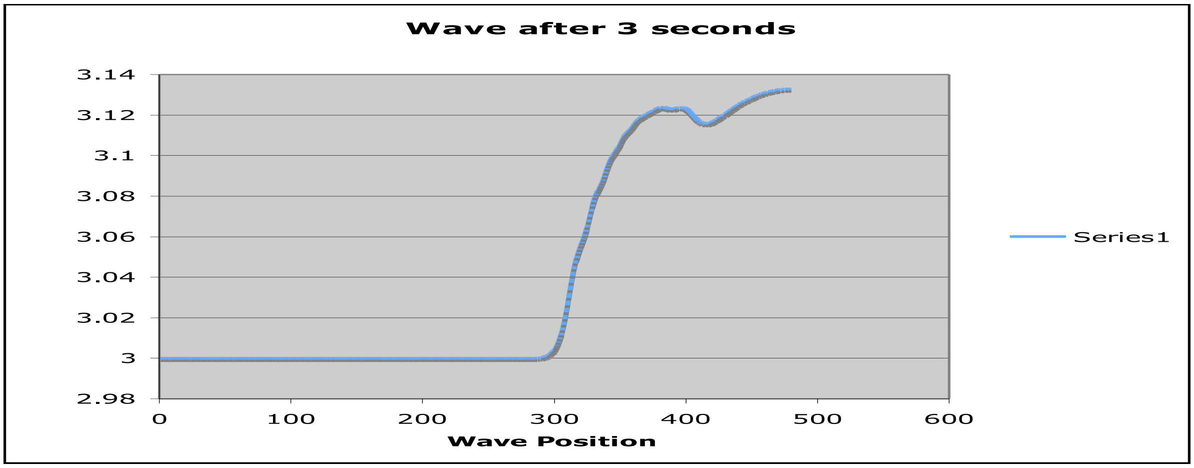

Wave after 0.3 seconds
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1

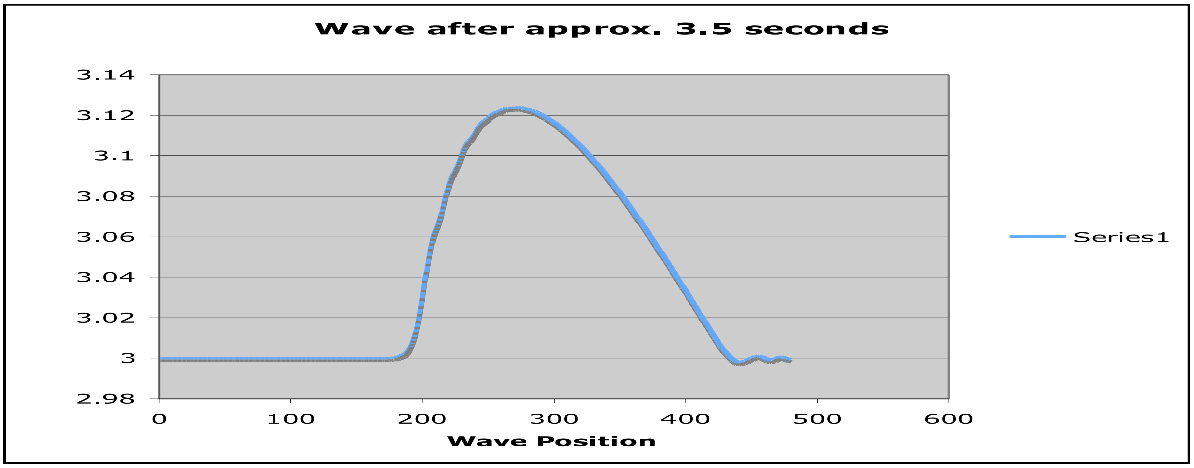

Wave after 0.6 seconds
3.14
3.12
3.1
3.08
3.06
3.04
3.02
3
2.98
0
100
200
300
400
500
600
Wave Position
Series1

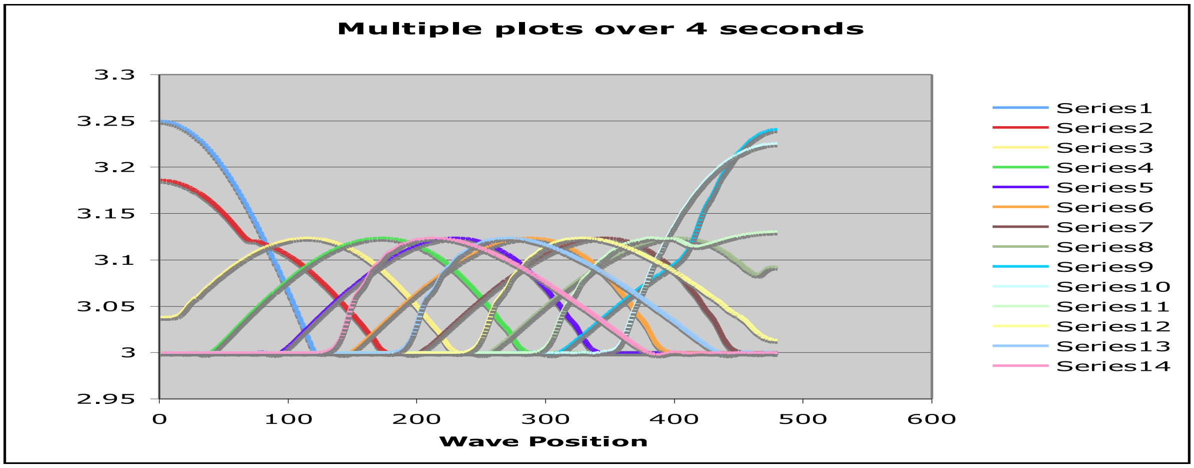

Wave after 1.2 seconds
3.14
3.12
3.1
3.08
3.06
3.04
3.02
3
2.98
0
100
200
300
400
500
600
Wave Position
Series1

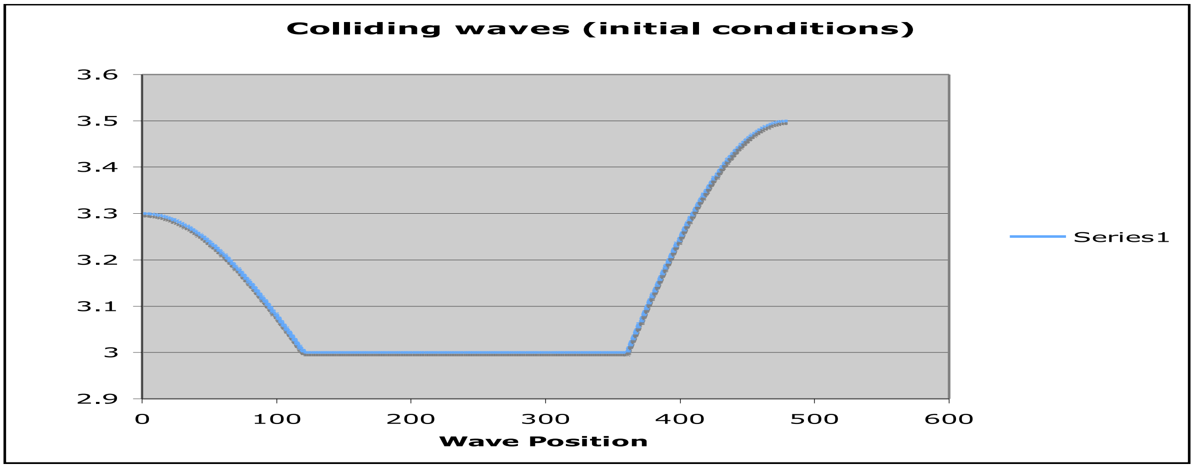

Wave after approx. 2 seconds
3.14
3.12
3.1
3.08
3.06
3.04
3.02
3
2.98
0
100
200
300
400
500
600
Wave Position
Series1

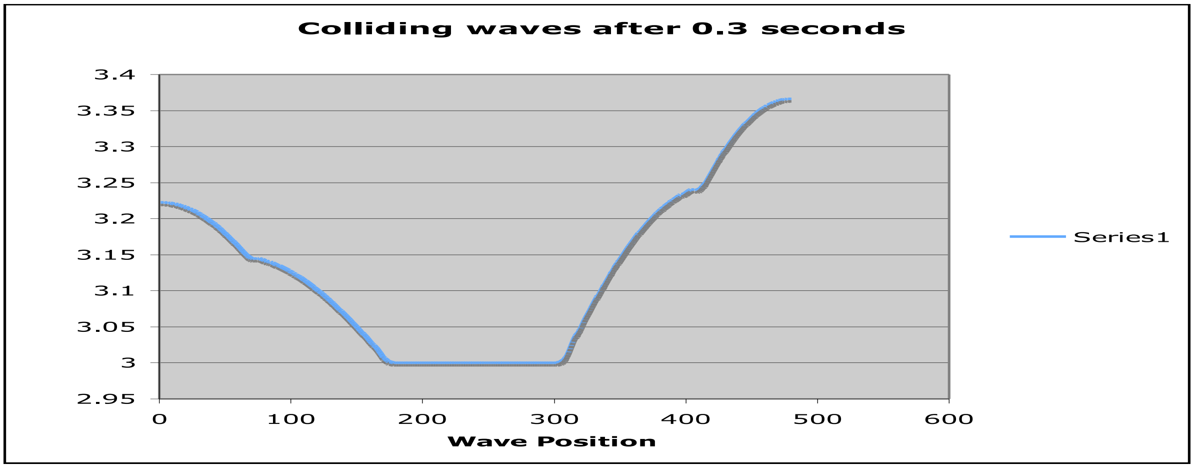

Wave after 2.7 seconds
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1

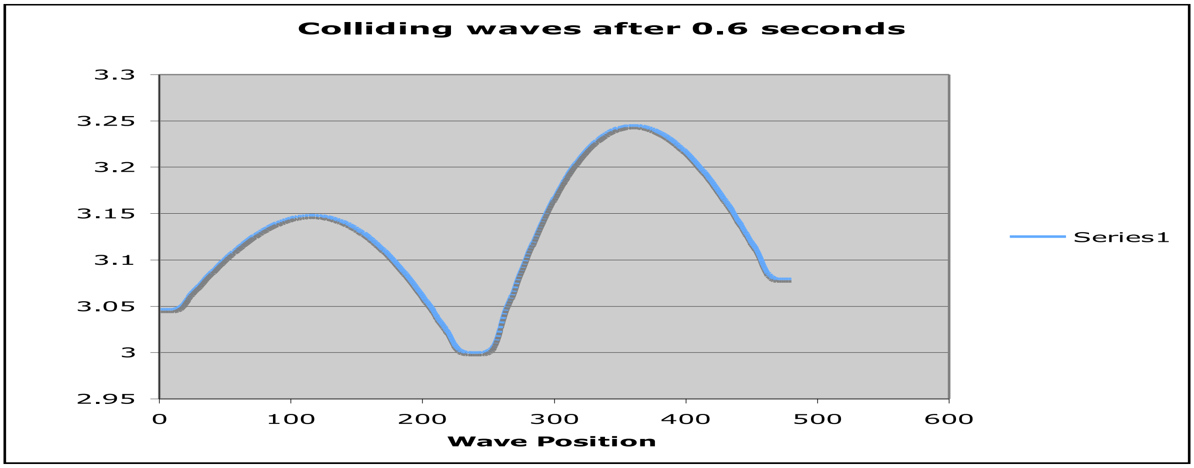

Wave after 3 seconds
3.14
3.12
3.1
3.08
3.06
3.04
3.02
3
2.98
0
100
200
300
400
500
600
Wave Position
Series1

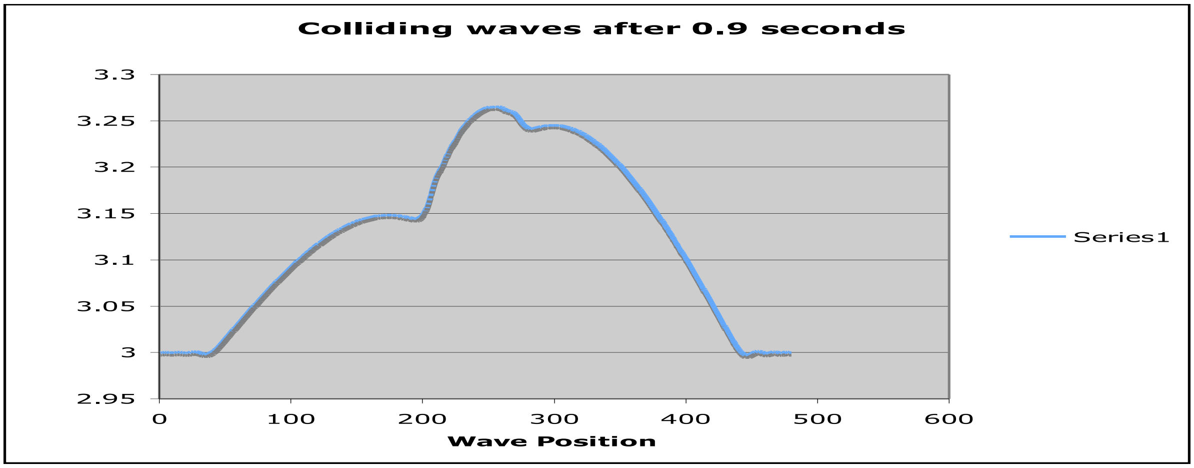

Wave after approx. 3.5 seconds
3.14
3.12
3.1
3.08
3.06
3.04
3.02
3
2.98
0
100
200
300
400
500
600
Wave Position
Series1

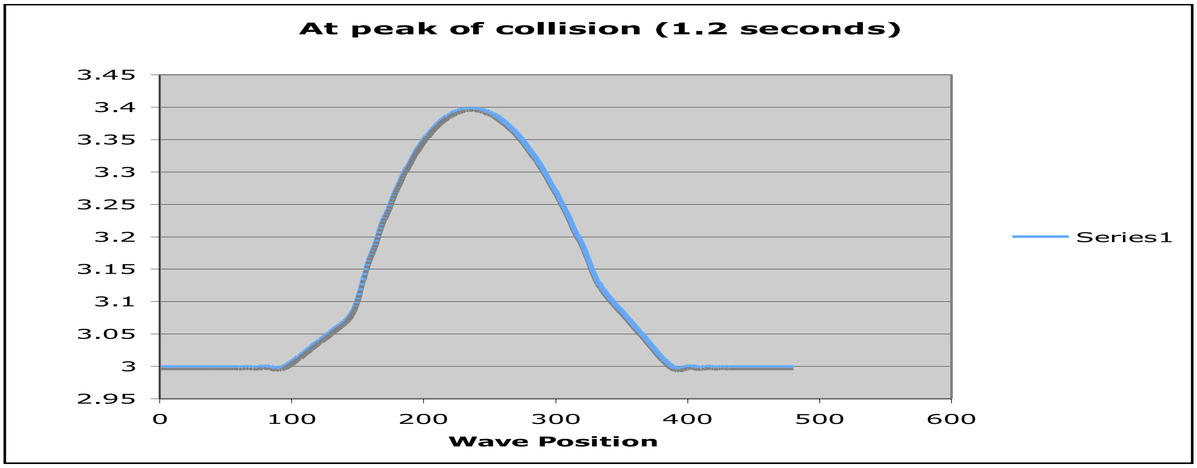

Multiple plots over 4 seconds
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1
Series2
Series3
Series4
Series5
Series6
Series7
Series8
Series9
Series10
Series11
Series12
Series13
Series14

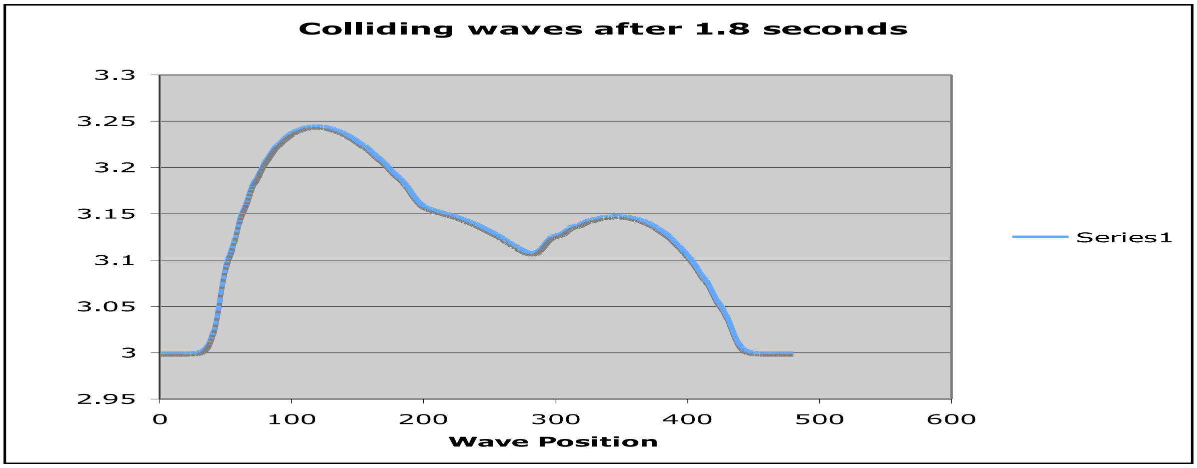

Colliding waves (initial conditions)
3.6
3.5
3.4
3.3
3.2
3.1
3
2.9
0
100
200
300
400
500
600
Wave Position
Series1

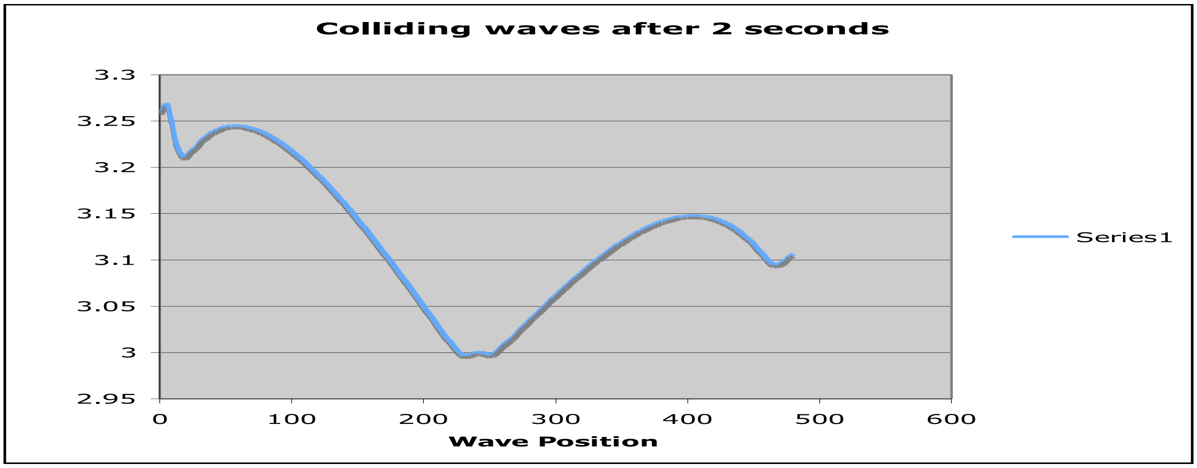

Colliding waves after 0.3 seconds
3.4
3.35
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1

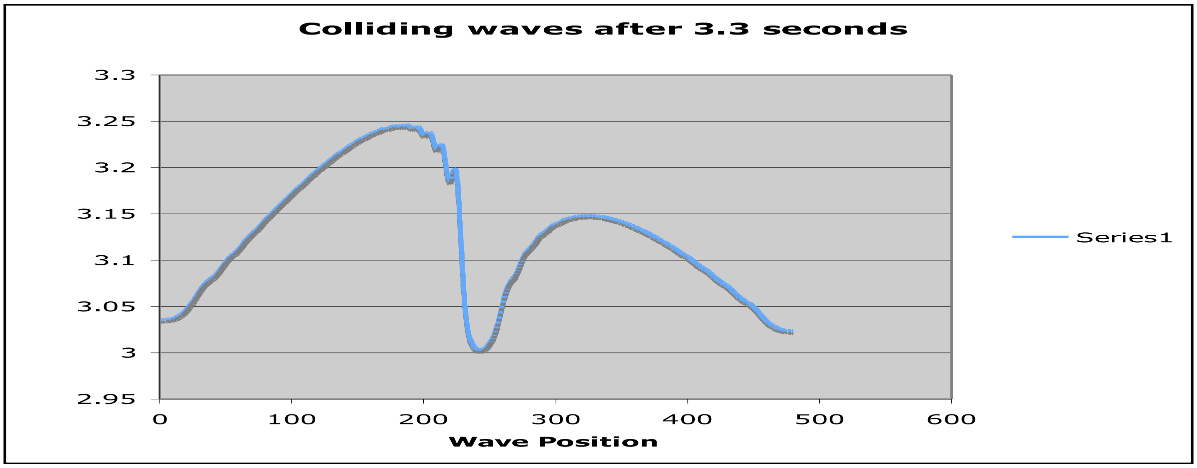

Colliding waves after 0.6 seconds
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1


Colliding waves after 0.9 seconds
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1


At peak of collision (1.2 seconds)
3.45
3.4
3.35
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1


Colliding waves after 1.8 seconds
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1


Colliding waves after 2 seconds
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1


Colliding waves after 3.3 seconds
3.3
3.25
3.2
3.15
3.1
3.05
3
2.95
0
100
200
300
400
500
600
Wave Position
Series1

Conclusion / Achievements

After conducting virtual simulations and then comparing them to observed phenomena, it has been proven that Lax-Wendroff based simulations can accurately predict the behavior of waves in a liquid. The two mimic each other very closely and with additional modifications, the simulation will likely model more complex scenarios just as accurately. The team worked together effectively, sharing ideas and information to help contribute to this research.

Acknowledgements

Special thanks to Mr. Bob Robey and David Olivas for their assistance with the project. Mr. Robey assisted with development, debugging, and testing of the program. He also provided considerable assistance with introducing the numerical methods used in the simulation. Mr. Olivas assisted with the construction of the tank used for conducting the physical experimentation. Additional thanks to Nick Bennett for his suggestions for additions to our program.

-Works Consulted

Sod, Gary: *A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws*- Journal of Computational Physics vol.27, pp1-31 (1978)