Model of Breaking Water Waves using 2D Euler Gas Simulation

David Ruffner *

Abstract

This paper explores a simple model of breaking water waves in a gravitional field by solving the two dimensional euler equations for two gases of different densities. Here the water is modelled as a dense gas and the air by a much less dense gas, and there is a constant gravitational field. The euler equations are solved numerically using a HLL Riemann solver. It has second order convergence because of PLM interpolation in space and Runge-Kutta3 time integration. The code is tested with and without gravity and found to be consistent. Waves are made in the interface between the gases by introducing a perturbation of high density gas in the low density region. To make the wave break, the two gases are given opposite horizontal velocities similar to the Kelvin-Hemholtz instability. Breaking phenomena qualitatively similar to breaking water waves is observed, however there are some quantitative differences.

1 Introduction

The impressive breaking phenomena of water waves, commonly observed at beaches, is a significant and interesting physical problem. Practically it is very important in the design of structures near the ocean and also ships that travel through rough seas. Breaking waves are known to create large forces that can cause much destruction. In addition it is a problem that still has more to be understood theoretically, although much progress has been made[1].

The problem of breaking water waves is fully nonlinear, as can be seen in the complex behavior before and after breaking. In addition, it is difficult to solve numerically, because there is an interface between the water and the air that needs to be taken into account. It is important to do this because the equations of state of a liquid and a gas are different, which affects their dynamics given by the navier stokes equation.

*Thanks goes to Professor MacFayden for his discussions, to Craig Lage for his many helpful ideas, to Sven Kreiss for his help with Debugging and for Jonathan Zrake for helping to explain the details of making this code.

An example of the progress made in this area is the work by Lubin et. al. in which they modelled breaking water waves in three dimensions with the navier stokes equation and captured the interface with a Lax-Wendroff TVD scheme[6]. Consequently, they could study the breaking process in detail, especially the effects of air entrapment and the creation of vorticity and turbulence.

However, that is a very extensive calculation, so it could be useful to study a simpler model, that can still produce qualitatively similar results. One simplification is to use the euler equations instead of the navier stokes equations. This greatly reduces the complexity of the problem because the euler equations are hyperbolic and can thus be solved using explicit methods. This is actually not a terrible approximation. The navier stokes equation can reduce to the same form as euler's equations when the Reynold's number is high. The Reynold's number, Re,is given by the equation $Re = \frac{UL}{\nu}$, where U is the characteristic speed, L is the characteristic length, and ν is the kinematic viscosity[8]. In water the $\nu = 10^{-6} m^2 s^{-1}$, and the length scales and velocity scales of water waves are around $1m$ and $1m/s$ respectively. This gives a Reynolds number of 10^6 which is clearly in the limit of invisicid flow, and can be modelled using the euler equations.

The simplification of using two gases with the same equation of state, instead of water and air, has less direct justification. Indeed if there are dynamic flows then the density of the gas can change blurring the distiction between the two gases. However, the continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = 0 \tag{1}$$

where ρ is the density and v is the velocity of the fluid, indicates that if there is no divergence of the velocity then ρ is constant[8]. In our calculations the divergence is non-zero, however it is relatively small on average, making it possible to distinguish between the gases.

1.1 Theory

The gas dynamics in euler's equations can be described by the conservation of mass, momentum, and energy. The conservation of mass is described by the continuity equation (1) which is again,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = 0 \tag{2}$$

where ρ is the density and v is the velocity of the fluid. Momentum conservation is given by,

$$\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla)\vec{v} = -\frac{\nabla P}{\rho} + \vec{f} \tag{3}$$

where P is the pressure and $\vec{f}$ is any body forces such as gravity. Finally the conservation of energy is given by,

$$\frac{\partial}{\partial t}(\frac{1}{2}\rho v^2 + \rho\epsilon) + \nabla \cdot [(\frac{1}{2}\rho v^2 + \rho\epsilon + P)\vec{v}] = \rho\vec{f} \cdot \vec{v} \tag{4}$$

where the ϵ is the internal energy density [8]. The non-linearity of these equations comes from the $(\vec{v} \cdot \nabla)\vec{v}$ term in the momentum conservation. To close these equations there needs to be an equation of state which gives P in terms of the the other quantities. In this paper the equation of state is that of an ideal gas,

$$P = (\gamma - 1)\rho\epsilon \tag{5}$$

where γ is the adiabatic index of an ideal gas, in this case it was taken to be $\frac{3}{2}$ [4].

Two common approximations to the euler equations to understand wave dynamics is the deep water wave and the shallow water wave approximations. The deep water wave approximation holds for when the wavelength λ is much smaller than the height, h, of the water and when the water is incompressible. This leads to a non-linear dispersion relation,

$$\omega = \sqrt{kg} \tag{6}$$

where g is gravity and k is the wavevector and ω is the angular frequency. In addition, the velocity of the fluid decays exponentially with depth.

In the shallow water approximation conversely $\lambda \gg h$ and again the fluid is taken to be incompressible. This leads to a linear dispersion relation,

$$\omega = \sqrt{gh}k \tag{7}$$

.

2 Computational Details

Numerically solving euler's equations involves discretizing the problem into an array of spatial cooridinates and then estimating the derivatives in the conservation equations to get the change in the conserved quantities at each point. For this paper there is a 2D array of values $\mathbf{U} = (\rho, p_x, p_y, E)$, density, momentum in both directions and energy density. To see how these change with time the euler equations are rewritten in flux conserving form following the method shown in class,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0 \tag{8}$$

where $\mathbf{F} = (\rho v_x, \rho v_x^2 + P, \rho v_x v_y, (E+P)v_x)$ is the flux in the x direction and $\mathbf{G} = (\rho v_y, \rho v_x v_y, \rho v_y^2 + P, (E + P)v_y)$ is the flux in the y direction. By writing it in this form we can use a finite volume method to determine the evolution of $\mathbf{U}$ which conserves the flux. This is important for keeping the model physically relevant. Clearly each point in the array $\mathbf{U}$, which is thought of as a volume of fluid, can only change if there is a net flux into that volume. Consequently, we only need to determine the flux at each edge of each element. $\mathbf{U}$ is known at discrete points so it can be approximated as being constant in each cell with discontinuities at the boundaries, which is the basis of the Godunov Method. Consequently determining the flux reduces to solving the Riemann problem. In this paper the HLL approximate Reimann solver was used to approximate the flux at each boundary. The HLL solver takes the value of $\mathbf{U}$ on both sides of the array and returns the flux through the boundary. From this by discretizing the derivatives in equation 8 we can determine the next value of $\mathbf{U}$ in an iterative manner.

To make the code second order, we used an interpolation method to get more accurate values to give to the HLL solver and we used a Runge-Kutta3 time step. The interpolation method we used was piecewise linear method, with a theta of 1.5. This method estimates the slope of $\mathbf{U}$ and estimates the value of $\mathbf{U}$ at the boundary based on the slope. It estimates the slope using two values of $\mathbf{U}$ on one side of the boundary. This method is robust because it ensures that the method becomes first order in the case of a shock, which reduces spurious oscillations.

For each time step we calculated the speed of sound at each point in the array $\mathbf{U}$ and found the max wave speed which can be $c_s \pm v_x$ or $c_s \pm v_y$. Since the euler equations are hyperbolic no information can travel

faster than the max wave speed. Therefore if your time step is small enough that the fastest wave can not travel more than dx or dy then the preceding discussion of calculating fluxes just based on nearest neighbors makes sense. This is made rigourous in the Courant-Friedrich-Levy(CFL) condition,

$$\Delta t < \frac{\Delta x}{MAX(\alpha^{\pm})} \tag{9}$$

where $MAX(\alpha^{\pm})$ is the maximum wave speed.

Our array **U** was finite, so we had to define boundary conditions(BCs) at the edges. In this paper we used three types of BCs, outflow, periodic, and reflecting. Since we used the PLM interpolation scheme we had to define the values of two ghost cells adjacent to each edge cell. Therefore the edge cells could use the values of the ghost cells to calculate its next value. The ghost cells would then be defined based on this new value. In outflow boundary conditions the two ghost cells are set to the same value as the edge cell. This causes any waves the reach the edge to flow out and not return. Periodic BCs on the other hand replaces the ghost cells on the right with the edge values on the left and viceversa for the ghost cells on the left. It goes similarly for the top and bottom. These boundary conditions caused waves that reach the edge to return on the opposite edge. For reflecting boundary conditions we used a similar condition as in outflow, except that the normal component of the momentum was reversed. This had the effect of reflecting any flow that reached the edge.

3 Tests

To verify that the code to numerically solve the euler equations was correct a number of tests were performed. This was found to be very important because there were some subtle bugs that needed to be fixed. The first test done was the strong shock described in a web page by James Stone [9]. This is a one dimensional test without gravity, however when done in two dimensions and a slice normal to the shock should reproduce the results in the website. The grid for this setup went from [-1,1] in the x and y directions. The the initial conditions for this shock were two horizontal strips divided by the line y=0 each with uniform values of **U**. The value of $\mathbf{U_{top}}$ is $\rho = 1.0, v_x = 0.0, v_y = 0.0$,and Pressure = 1.0, where these physical quantities need to be converted into the conserved quantities in **U**. Similarly $\mathbf{U_{bottom}}$ is set to $\rho = 10.0, v_x = 0.0, v_y = 0.0$,and Pressure = 100. These initial conditions can be seen in the the top two panels of Figure 1. The BCs were set to outflow BCs. The high pressure on the bottom creates a shock wave which actually consists of two waves that extend into the low density region. A rarefaction wave exteds back into the high pressure side simultaneously. This behavior is clearly evident in our test case, and it corresponds very well with the example given on the Jim Stone website, indicating the code passes this test.

The next test done was the advection of circular region of high density gas in a surrounding region of low density gas. The whole grid was given the identical velocity with periodic boundary conditions. Theoretically the density profile should remain constant because in the rest frame of the fluid there is no velocity anywere, so there should be no flux in density. This is an important test because it provides evidence for the use of two different density gases as models for air and water, because they don't mix, at least with this simple flow. In addition it is important because it has an exact solution, so by calculating the L1norm, the convergence rate can be estimated. Figure 2 shows the results of this test. The initial conditions in the top left panel were set to $v_x = 1.0, v_y = 1.0$, and P = 1.0 everywhere, and ρ was set to a circular gaussian of height 1 and width .25, on a base of $\rho = 1$. The top right panel is after the density peak has returned to its original location after one circulation, this is at a resolution of 256x256. The two lower graphs are log log graphs of the L1norm on the left and the runtime on the right versus the resolution. The code is second order so the slope of the L1norm should be -2, and on the graph it is -1.968, very close to the theoretical. Interestingly there is some deviation at small resolution, and this seems like it must be due to boundary effects since nothing else changes except the proportion of cells that are near the boundary. The runtime depends on the size of the array and the number of iterations. If the resolution is doubled then the size of the array goes up by four and the time step is halved by the CFL condition. Consequently, the runtime goes up by a factor of 2^3. Therefore runtime is proportional to the resolution cubed, so on a log log plot it should have a slope of 3. And the slope of the run-time observed for the code is 2.977. This is strong evidence that the code is working correctly.

The final test done without gravity was a small perturbation in pressure that created a sound wave that went off in all directions. Since the initial pressure perturbation was set to be circularly symmetric, the resulting sound wave should be symmetric, so it is a test that the code treats both directions equally. In

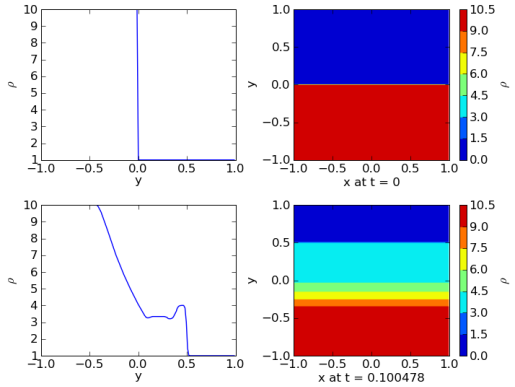

Figure 1: Strong shock test case. The top two graphs are the initial conditions, and the bottom two graphs are after t=0.1 . The left graphs are slices of the density along the y axis at x=0. The right graphs are plots of density on the whole grid with color corresponding to density. Note there is no gravity in this test.

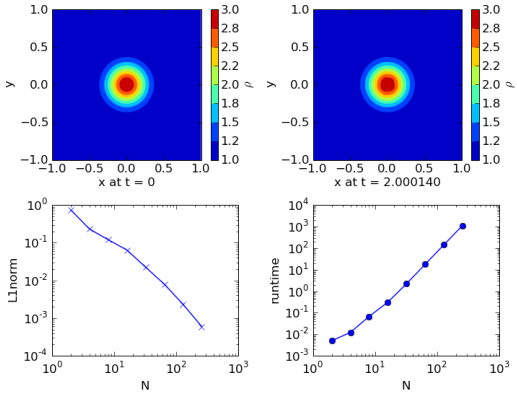

Figure 2: Advection of a gaussian density profile. Upperleft hand graph is the initial conditions, the upper righthand graph is the final condition. The lower left hand plot is L1norm vs. resolution, and the lower righthand plot is runtime vs resolution. The similarities of the initial and final conditions and the slopes of the L1norm and runtime plots indicate the code is working correctly.

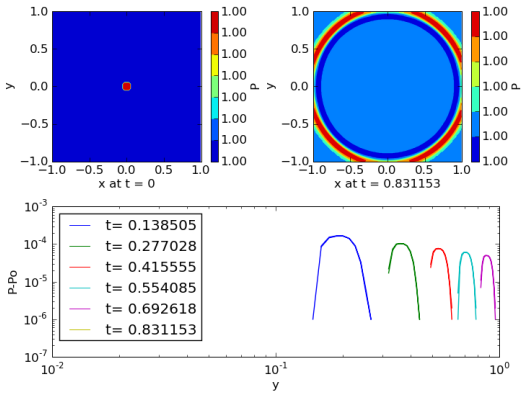

Figure 3: Propogation of a circular sound wave. The top left panel is the initial pressure, the top right panel is the final pressure, and the bottom panel is a loglog plot of pressure vs. radius, indicating that intesity falls of like $\frac{1}{r}$

addition in the limit of small perturbations the wave should travel at the sound speed [4]. In this test a small circular region of radius .1, in the center of the same sized grid, was given $P_{in} = 1.001$, while the rest of the grid was given $P_{out} = 1.0$. The $\rho = 1$ every where and the velocity was zero everywhere. Here in Figure 3 are the results from this test, and the resulting circular sound wave is clearly symmetric as expected. Since $c_s = \sqrt{\frac{\gamma P}{\rho}} \approx 1.225$ the wave should propogate at this speed and it does indeed. In addition, since the wave expands radially, the intensity should go as $\frac{1}{r}$. Making a log log plot of pressure vs. radius, we see that the peaks are linear and have the expected slope of -1.

Gravity was then added to the code, by adding the source terms given in equations 3 and 4. This corresponded to replacing equation 8 with,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{S} \tag{10}$$

where $\mathbf{S} = (0, 0, \rho g, \rho g v_y)$. To test that this was done correctly the Rayleigh Taylor instability was tested and compared qualitatively to the test on James Stone's website. In the Rayleigh Taylor instability a dense gas is placed above a less dense gas, and the pressure is adjusted to support the gas against gravity. In addition, the BCs in the y direction need to be set to reflecting boundary conditions so the gas doesn't fall. This setup is unstable, but if there is no perturbation then it will remain indefinitely. However, if a perturbation is applied then the denser gas begins to flow down in the less dense gas and vice versa. The result is nice swirling plumes [9]. The results for ρ in this test for the code for six different time steps are shown in Figure 4. The initial conditions shown in the upper left hand panel were velocity zero everywhere, except for a small perturbation in v_y at x=0, and y=0. The upper density was 2 and the lower was 1, and the pressure was linear with a slope of -$\rho_{up}g$ for the upper half and -$\rho_{down}g$ for the lower part to support the gas against gravity. As the time steps continue it is clear qualitatively that at first the gas remains stable, then the instability grows exponentially, and finally plumes of high density gas fall, and plumes low density gas rise as is expected theoretically. The graph created by my code is somewhat different than the graph on James Stone's website, but this is due to slightly different initial conditions. Qualitatively they dynamics are the same so this produces evidence that the code works with gravity as well.

4 Main Results and Analysis

The goal of this paper is to model the breaking of waves at the interface between water and air by the waves at the interface between two eulerian gases. So as a first step to study these waves we considered the situation where the wave at the interface of the gas does not break but travels uniformly. To do this we set up the dense gas below the less dense gas with the interface at y=0, velocity was set to zero everywhere and the pressure was set to hold up the gas against gravity. This as opposed to the Rayleigh instability is stable. A perturbation at the surface will not grow exponentially, but instead will send out waves in both

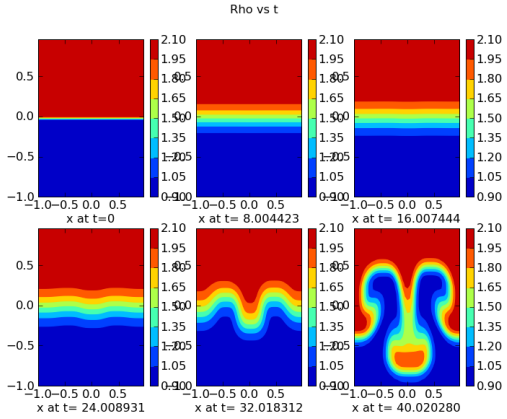

Figure 4: Now with gravity, the Rayleigh Taylor instability is shown here, each graph shows the density at every point for a specific time.

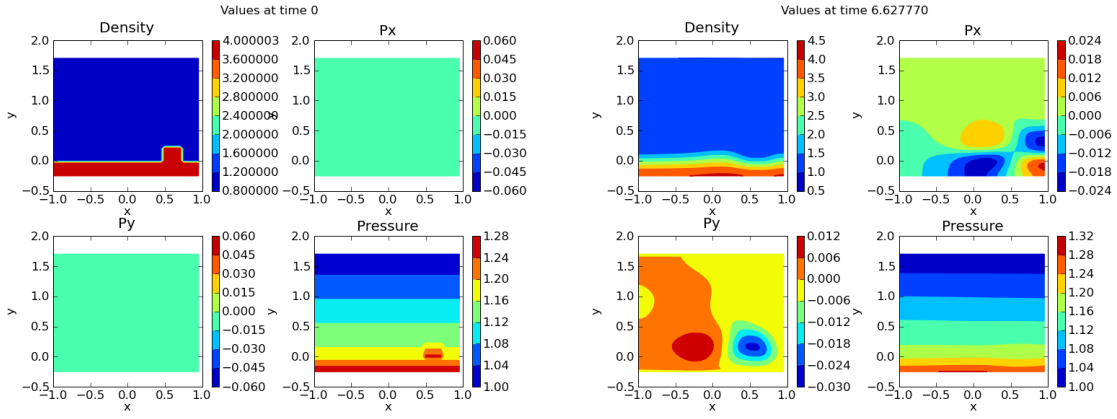

Figure 5: Waves caused by a density perturbation in a shallow setup with gravity. The left four graphs are the initial state, and the right four graphs are the final state.

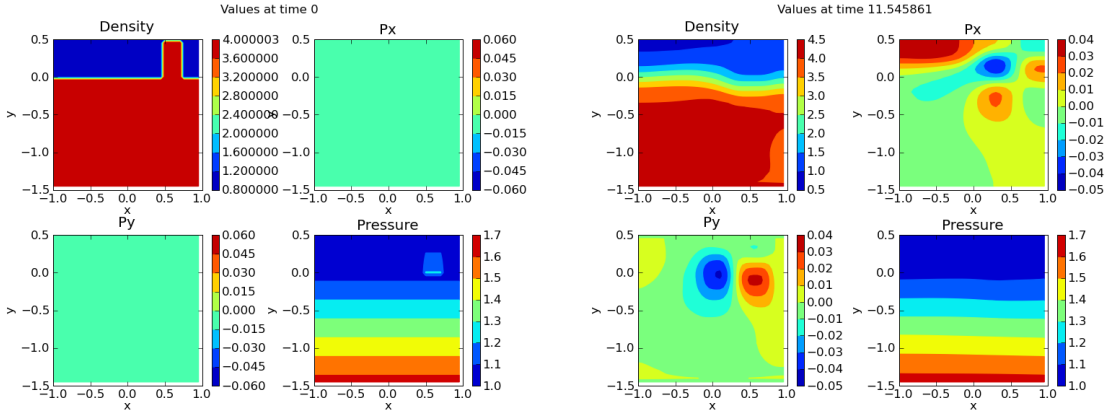

Figure 6: Waves caused by a density perturbation in a deep setup again with gravity. The left four graphs are the initial state, and the right four graphs are the final state.

directions along the surface. This was tested by creating a perturbation in density, a small rectangular region above y=0 was filled with the high density gas instead of the low density. The lower gas had $\rho = 4$ and the upper gas had $\rho = 1$. This initial condition is shown in the left four panels of Figure 5 and again at a larger depth in Figure 6. In both situations, as time evolves the density perturbation caused two waves of the higher density gas to travel in either direction. One is shown in Figure 5 in the left four panels, at a time t = 6.6. The wave is clearly smaller than the initial perturbation and spreads out. In Figure 6 the right panels show the wave at t=11.5, and also the wave has become smaller and more spread out.

This type of wave can be compared to the theory of shallow water waves and deep water waves. In the case of figure 5 it appears that the wavelength of the propogating wave is longer than the height of the dense gas. In this case the shallow water approximation should apply so there should be flow in the x direction even down to the bottom. We can see in right panels of Figure 5, that there is indeed this type of flow. From the dispersion relation we see that the wave speed is expected to be $v = \sqrt{gh} \approx .16$. Roughly from the graph the wave speed was measured to be $\approx .1$, which is the same order of magnitude but could differ because the wavelength initially is not long effectively use the shallow wave approximation.

In the case of figure 6, the deep water waves approximation should apply since the height of the dense gas is 1.55 and the wavelength of the resulting wave is clearly smaller than this. According to this approximation the velocity of the fluid should decay exponentially away from the surface. It can be seen from the right four panels of Figure 6 that qualitatively this is the case. In addition, we see evidence from the overlap of x and y velocities that there is some rotation in the fluid which is expected in the deep water approximation. The velocity in this approximation is expected to be $v = \frac{1}{2}\sqrt{\frac{g\lambda}{2\pi}}$ where λ is the wavelength of the wave. Estimating from the graph that $\lambda \approx 1$ and given that g = .1, this leads to $v \approx .06$, and estimating directly from the graph gives v also around .06, but there is large uncertainty since the wavelength changed as the wave propogated. However, it is evidence that this approximation is valid in this case.

Together the applicability of the shallow water and deep water approximations give evidence that these two gases can be models of the water fluid interface. Therefore, we can look for initial conditions that will create breaking waves in the higher density gas. When analyzing these results we can hope to gain insight into the water air problem. One set of initial conditions that was studied is shown in Figure 7 in the left four panels. Like the previous setups, the higher density gas($\rho = 10$) is below y=0 and the less dense gas($\rho = 1$) is above. In addition, there is also a rectangular perturbation in density, but now it is larger. In this setup the now the higher density gas, including the perturbation has $v_x = 1.0$ which is to the right, and the lower density gas has a $v_x = -1.0$ which is to the left. The pressure holds up all the gas, except under the perturbation it is not adjusted to hold the extra weight of the perterbation. In addition there is a perturbation in $v_y = .1$ in the perturbation of density and under it but offset. These initial conditions were chosen because of there similarity to both the previous tests and to the Kelvin-Hemholtz instability. In the previous tests with the perturbation in density, it was seen that a wave travelled along the interface but did not break, so something more needs to be added. In the Kelvin-Hemholtz instability which is without gravity, two eulerian gases flow past each other like in the horizontal strips in this setup. If there is no perturbation they continue to flow, but if there is a small perturbation, then it grows exponentially, and swirls form between the gases. If there is gravity, and the denser gas is below, then it is conceivable that swirls of high density gas would form but would be pulled back down by gravity, which would resemble breaking.

Running the code with these initial conditions, we see some interesting effects. After a short time, t=.1, shown in the right four panels of Figure 7 we see that a zone of high pressure has developed on the right side of the density perturbation, and a corresponding zone of low pressure on the other side. This clearly caused by the motion of the density perturbation against the flow of the low density gas. There also small zones with non-zero v_y which is due mostly to the flow up and over the perturbation in density. There is positive v_y on the side going into the flow and negative v_y on the trailing edge.This rotating flow should have a shearing effect on the perturbation.

Indeed as time goes on we see that the density perturbation gets sheared, but also rises up due to a wave formed from the pressure differences, and in the process by t=2.35 becomes a breaking wave. This is shown in Figure 8, where we can see in the top left hand graph the breaking wave shaped profile of the

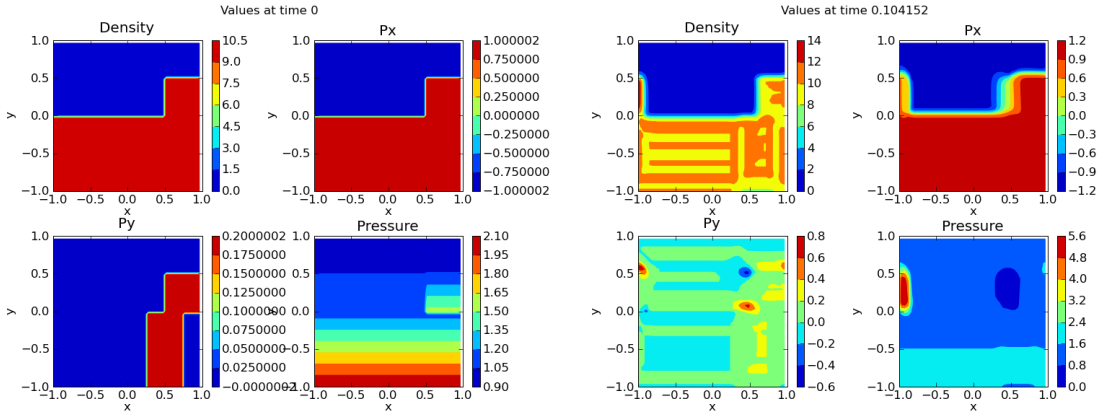

Figure 7: Kelvin-Helmholz setup with gravity and a large density perturbation. The left four graphs are the initial state and the other graphs are at t = .1 .

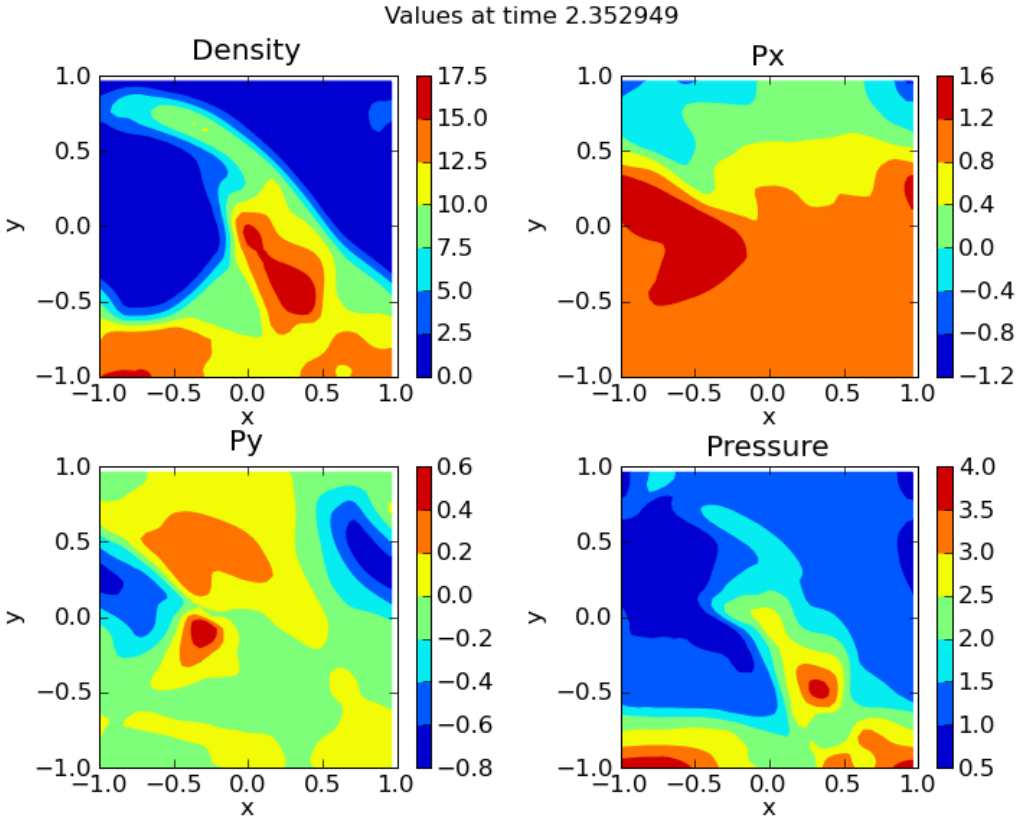

Figure 8: With the same setup as in Figure 7, observation of breaking wave phenomena.

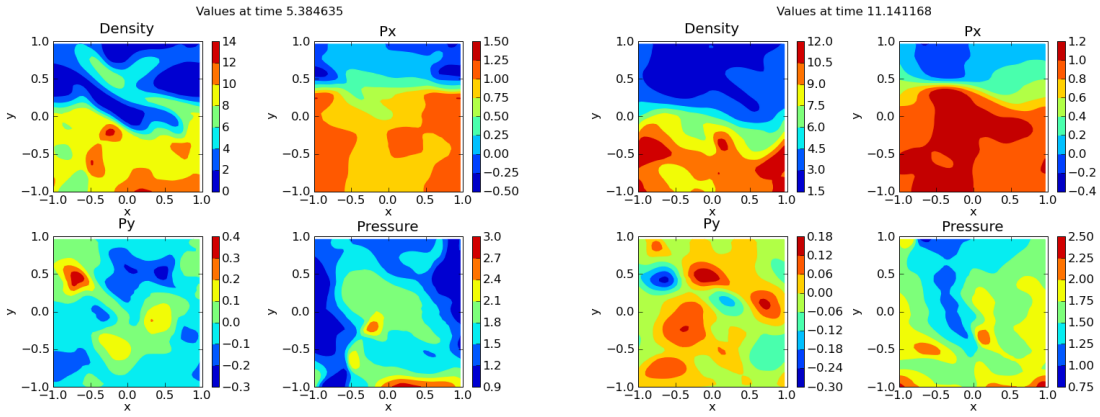

Figure 9: Again the same set up. The left four graphs show the depression of the crest that broke over, and the right four graphs show after the wave is dissipated.

denser gas. One note is that there is significant density variation in the wave, however it is clear which region is associated with the higher density gas and which is with the lower. Another thing to notice is that the height of this wave is much larger than the initial perturbation. It must have picked up energy from the opposing flow of the two different density gases.

In the next graph at t=5.38, in the left side of Figure 9, we see that the crest in Figure 8 has fallen significantly. This indicates that our assumptions were correct qualitatively. Interestingly, there is a region of large v_y at the place where the end of the crest is about to touch the rest of the high density gas. This seems like evidence of entrapment of low density gas under the crest of the high density gas, it is forced out and picks ups some velocity in the y direction. This sort of phenomena is important in the study of the breaking of water waves, so it is very interesting that this is also seen in this model.

Finally in the last graph at t= 11.14, in the right side of Figure 9 we see that the wave dissipated. There is still velocity and pressure gradients but on a smaller scale than the large wave in the previous figures. Also one notes that there has been some mixing of the two gases. In this frame there are large regions of the low density region that have a density between 3 and 4.5, which is much larger than the original density of 1 in this region. This puts a limitation to how effective this could be in modelling the air water interface.

5 Conclusions and Future Work

From these results we see that using two eulerian gases it is possible to simulate behaviour that is similar to the effects seen in the interface between air and water. For instance, with a density perturbation waves were created that propogated along the surface. Depending on the depth of the higher density gas, these waves were very similar to shallow water waves or deep water waves respectively. In addition, the breaking of a wave was simulated by a density perturbation in a Kelvin-Hemholtz setup with gravity. The high density gas formed a wave with a crest that over turned and fell, there was an entrapment and subsequent outflow of low density gas by this crest. And eventually the high density gas returned to its original level, but there was much small scale motion still going on. This really seems to capture the essence of the breaking wave. However, one problem was that there was no longer a sharp divide between the two density gases. In addition, this was a forced wave break carefully set up by the initial conditions. It would be more meaningful if this breaking could occur just from the interaction of the wave with a sloping floor for example. Further investigations, would search for such breaking behaviour and would consider densities and pressures that are more quantitatively similar to that of water and air.

References

[1] Li, J.C. and M. Lin. “Prediction of Deterministic and Random Forceon Structures by Plunging Breaking Waves”. M.L. Banner, R.H.J.

Grimshaw(Eds.). *Breaking Waves IUTAM Symposium* Sydney/Austrialia 1991. Springer-Verlag Berlin Heidelberg 1992.

[2] Rahman, Matiur. *Water Waves: Relating Modern Theory to Advanced Engineering Applications*. Claredon Press. Oxford, 1995.

[3] Lighthill James. *An Informal Introduction to Theoretical Fluid Mechanics*. Claredon Press. Oxford 1986.

[4] MacFayden, Andrew, Lectures from Computational Physics class fall 2009.

[5] Mader, Charles L., *Numerical Modeling of Water Waves*, Berkeley: University of California Press, 1988.

[6] Lubin, Pierre et al., C. R. Mecanique 331 (2003)

[7] Zrake, Jonathan. Recitations from Computational Physics class fall 2009.

[8] Chaikin, Paul. Lecture Notes from Dynamics class fall 2009.

[9] Stone, James M. "The Code Test Page". http://www.astro.princeton.edu/ jstone/tests/. 12-21-09