Comparison of Modeling Explosions: Eulerian vs. Lagrangian

Esther Lin
11726156

Keenan McConkey
21337167

Austin Khorram
22105150

Justin Myles
52417152

Coleman Kaps
24793151

October 8, 2020

Contents

1 Introduction

There are many approaches when it comes to describing the physics of a 2D explosion. This project was designed with two particular approaches in mind: the *Lagrangian Model of Fluids* considering the system as N particles, and the *Eulerian Model of Fluids* considering the system as a continuum. A Lagrangian model was also used to describe the motion of a circular object subject to the propagation of a pressure wave resulting from the initial explosion. In this simulation the explosion was defined to be a near-singularity of extremely high density air at $t = 0$ allowed to propagate accordingly with each time step.

We designed three different simulations with varying quantities of the same initial explosion propagating from different locations on a 2D surface, and through theoretical derivations and simulations we determined which aforementioned approach was optimal for describing the system. All of the mathematics done were discrete for implementations.

2 Objectives

There are two objectives for this project:

- To compare the Lagrangian model of fluids versus the Continuum Eulerian approach to simulating point mass explosions in a closed box
- To use the Lagrangian to model the motion of a rigid body resulting from the explosion

3 Lagrangian Model of The Explosion

In order to simulate the Lagrangian model, we tracked groups of particles instead of individual ones, following a large number of groups of particles as they propagate away from the central explosion. We divided our fluid up with a grid. Within each box, we assumed homogeneity and instant equilibrium. We also assumed diatomic gas, allowing us to use the ideal gas law.

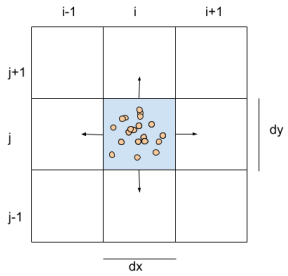

Figure 1: Groups of Particles for the Lagrangian Model

3.1 Governing Equations

We then applied the Lagrangian to each grid, using (x,y) as the centre of mass of each box. The potential energy was the tricky one to determine. We identified pressure, temperature and interactions as the major contributors.

Potential and Kinetic Energies

$$T_{i,j} = \frac{1}{2}m(\dot{x}_{i,j}^2 + \dot{y}_{i,j}^2)$$
$$U_{i,j} = U_{pressure_{i,j}} + U_{temperature_{i,j}} + U_{interactions_{i,j}}$$

Ideal Gas Law and Gay-Lussac's Law

$$PV = nRT \text{ , R is the specific gas constant}$$
$$\frac{P_1}{T_1} = \frac{P_2}{T_2}$$

Equipartition Theorem and the Maxwell–Boltzmann Distribution
We applied the equipartition theorem to each grid with 2 degrees of freedom (movement in the x and y directions).

$$\begin{aligned} T_{i,j} &= \frac{1}{2}m(\dot{x}_{i,j}^2 + \dot{y}_{i,j}^2) \\ &= kT_{i,j}, \text{ k is the Boltzmann's constant} \end{aligned}$$

3.2 Solving the Equations

Applying the thermodynamics above, we were able to write the potential energy terms for the pressure and temperature:

$$\begin{aligned} U_{pressure_{i,j}} &= P_{i,j}V_{i,j} \\ &= \frac{R}{k}[T_{i,j}] \\ U_{temperature_{i,j}} &= c_v T_{i,j} \\ &= c_v \frac{1}{k}[T_{i,j}] \end{aligned}$$

Interactions
For the interaction term, we hypothesized that if we made each box infinitesimally small, the properties (temperature, pressure, etc.) should be continuous with adjacent boxes. This allowed us to us the harmonic oscillator potential to model the interactions. We also assumed radial symmetry for the box containing the point explosion.

$$\begin{aligned} U_{interactions_{i,j}} &= \frac{1}{2}mw_{i,j_x}^2\Big(|x_{i,j} - x_{i+1,j}|^2 + |x_{i,j} - x_{i-1,j}|^2]\Big) + \frac{1}{2}mw_{i,j_y}^2\Big(|y_{i,j} - y_{i,j+1}|^2 + |y_{i,j} - y_{i,j-1}|^2\Big) \\ &= \frac{1}{2}mw_{i,j}^2\Big(\sum_{n=-1}^{1}|x_{i,j} - x_{i+n,j}|^2 + \sum_{m=-1}^{1}|x_{i,j} - x_{i,j+m}|^2\Big) \end{aligned}$$

For the boxes without the explosions, radial symmetry can not be assumed. The interaction term will be:

$$\begin{aligned} U_{interactions_{i,j}} &= \frac{1}{2}mw_{i,j_x,right}^2|x_{i,j} - x_{i+1,j}|^2 + \frac{1}{2}mw_{i,j_x,left}^2|x_{i,j} - x_{i-1,j}|^2] \\ &\quad + \frac{1}{2}mw_{i,j_y,top}^2|y_{i,j} - y_{i,j+1}|^2 + \frac{1}{2}mw_{i,j_y,bottom}^2|y_{i,j} - y_{i,j-1}|^2 \end{aligned}$$

However, this equation has too many unknowns for our implementation. We decided to enforce the interaction potential of the explosion box for all boxes, assuming symmetry in propagation of pressure, density and temperature.

Lagrangian

$$\begin{aligned} L_{i,j} &= T_{i,j} - U_{i,j} \\ &= \frac{1}{2}m(\dot{x}_{i,j}^2 + \dot{y}_{i,j}^2)\Big(1 - \frac{1}{k}(R + c_v)\Big) - \frac{1}{2}mw_{i,j}^2\Big(\sum_{n=-1}^{1}|x_{i,j} - x_{i+n,j}|^2 + \sum_{m=-1}^{1}|x_{i,j} - x_{i,j+m}|^2\Big) \\ &= \frac{1}{2}m\beta^2(\dot{x}_{i,j}^2 + \dot{y}_{i,j}^2) - \frac{1}{2}mw_{i,j}^2\Big(\sum_{n=-1}^{1}|x_{i,j} - x_{i+n,j}|^2 + \sum_{m=-1}^{1}|x_{i,j} - x_{i,j+m}|^2\Big) \end{aligned}$$

where $\beta^2 = 1 - \frac{1}{k}(R + c_v)$

From here, we solved a coupled system of equations to find the movement of a group of particles located at coordinates (i,j) moved in an oscillatory manner. A_1, B_1, A_2, B_2 where all normalized to 1 in our simulations.

$$\begin{bmatrix} x_{i,j}(t) \\ y_{i,j}(t) \end{bmatrix} = \begin{bmatrix} A_1 e^{-i\frac{w_{i,j}}{2\beta}t} + B_1 e^{i\frac{w_{i,j}}{2\beta}t} \\ A_2 e^{-i\frac{w_{i,j}}{2\beta}t} + B_2 e^{i\frac{w_{i,j}}{2\beta}t} \end{bmatrix}$$

3.3 Limitations

The limitations of this model are that we don't know enough thermodynamics and statistical mechanics to model the breadth of this scenario. Also, this model becomes incredibly computationally intensive as the number of grids are increased. However, the results of our model are sufficient for modeling smaller perturbations, compared to the Eulerian model.

It will be worth pursuing the suggestions Dr. Sigurdson gave, investigating the Lagrangian description of continuum mechanics for this problem.

4 Eulerian Model of The Explosion

Our Eulerian model of the explosion is based heavily upon a paper: *Explosion Simulation using Compressible Fluids* by Abhinav Golas et al, link: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4756053. Some pieces of the simulation have been removed or replaced by simpler processes in order to speed up the development and increase the speed that the simulation can run at. Most notably, rather that a partial time stepped Semi-Lagrangian method using Back and Forth Error Compensation and Correction, we simply solved the advection equation once, implementing finite differences. This reduces the accuracy of the simulation, especially at large time steps or after many time steps. An improvement that could be made moving forward would be to implement this algorithm.

4.1 Governing Equations

In all equations below, we represent the total vector velocity as $\boldsymbol{v}$, which has individual components labeled as follows:

$$\boldsymbol{v} = u\hat{\boldsymbol{x}} + v\hat{\boldsymbol{y}}$$

Here, $\frac{D}{Dt}$ is the material derivative. In 2-D:

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \frac{\partial x}{\partial t}\frac{\partial}{\partial x} + \frac{\partial y}{\partial t}\frac{\partial}{\partial y}$$

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y}$$

Conservation of Mass

$$\frac{D\rho}{Dt} = \rho\boldsymbol{\nabla}\cdot\boldsymbol{v}$$

Navier-Stokes Momentum Equation (ignoring gravity)

$$\rho\frac{D\boldsymbol{v}}{Dt} = -\nabla P + \mu\nabla^2\boldsymbol{v} + \frac{\mu}{3}\nabla(\boldsymbol{\nabla}\cdot\boldsymbol{v})$$

Conservation of Energy Using the first law of thermodynamics

$$\rho\frac{DN}{Dt} = k\nabla^2 T - P\boldsymbol{\nabla}\cdot\boldsymbol{v} + \Phi$$

where Φ contains the viscous dissipation terms defined below:

$$\Phi = \frac{-2\mu}{3}(\boldsymbol{\nabla}\cdot\boldsymbol{v}) + \frac{\mu}{2}\sum_{i,j\epsilon(x,y)}(\frac{\partial\boldsymbol{v}_i}{\partial j} + \frac{\partial\boldsymbol{v}_j}{\partial i})^2$$

As well, we can approximate the relationship between internal energy N and T as:

$$N = c_v T$$

Unfortunately, due to the speed and the large forces involved in an explosion, the flow is compressible and as such the temperature and pressure will change as density changes. We can approximate the change in pressure due to a change in density using the ideal gas law:

$$P = \rho RT, \text{ where R is the specific gas constant.}$$

4.2 Solving the Equations

Now that we have enough equations to define the motion, we need to go about solving them. None of these equations can be solved analytically (not easily anyway), and because we are interested in how they vary in time, finite difference solutions are a natural choice. While implicit finite differences would present an always stable solution, this yields n^2 coupled equations for each time step. As such, we elected to use explicit finite differences and controlled the time step and Δx in order to preserve stability as long as possible. Using explicit finite differences we can approximate derivatives as follows (here $\Delta y = \Delta x$):

$$\frac{\partial \psi(x,y,t)}{\partial x} = \frac{\psi(x+\Delta x, y, t) - \psi(x-\Delta x, y, t)}{2\Delta x}$$

$$\frac{\partial^2 \psi(x,y,t)}{\partial x^2} = \frac{\psi(x+\Delta x, y, t) - 2\psi(x,y,t) + \psi(x-\Delta x, y, t)}{(\Delta x)^2}$$

$$\frac{\partial^2 \psi(x,y,t)}{\partial x \partial y} = \frac{\psi(x+\Delta x, y+\Delta y, t) + \psi(x-\Delta x, y-\Delta y, t) - \psi(x+\Delta x, y-\Delta y, t) - \psi(x-\Delta x, y+\Delta y, t)}{4(\Delta x)^2}$$

$$= \frac{\psi_{1,1} + \psi_{-1,-1} - \psi_{1,-1} - \psi_{-1,1}}{4(\Delta x)^2} \tag{1}$$

Now that the tools are in place we needed to actually solve the equations for each time step. Following the referenced paper, we broke apart the convective terms from each equation and solved them separately. This allows improvement on approximating the convective terms in each equation by using specific techniques, such as the semi Lagrangian method used by Golas et al. The process is as follows:

1. Solve non convective Navier-Stokes Equation to approximate acceleration.

$$\rho \boldsymbol{a} = \rho \frac{\partial \boldsymbol{v}}{\partial t} = -\nabla P + \mu \nabla^2 \boldsymbol{v} + \frac{\mu}{3} \nabla (\boldsymbol{\nabla} \cdot \boldsymbol{v}) \tag{2}$$

$$\rho a_x = -\frac{\partial P}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + \frac{\mu}{3}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial x \partial y}\right) \tag{3}$$

$$\rho a_y = -\frac{\partial P}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) + \frac{\mu}{3}\left(\frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 u}{\partial y \partial x}\right) \tag{4}$$

2. Use the acceleration to approximate new velocities

Depending on whether you sample the velocity at the the edge of the grid square, you can find two different velocities:

$$\boldsymbol{v}_{avg,t+\Delta t} = \boldsymbol{v}_t + \boldsymbol{a}\frac{\Delta t}{2} \tag{5}$$

$$\boldsymbol{v}_{max,t+\Delta t} = \boldsymbol{v}_t + \boldsymbol{a}\Delta t \tag{6}$$

3. Approximate changes in temperature due to non-convective terms of the conservation of energy equation.

Here we can expand the equation and sub in finite differences similarly to when we approximate the acceleration. We do not explicitly expand the equation here and leave it in vector notation to keep it compact.

$$\rho c_v \Delta T = \Delta t(k\nabla^2 T - P\boldsymbol{\nabla}\cdot\boldsymbol{v} + \Phi) \tag{7}$$

$$\text{here we use: } \frac{\partial N}{\partial t} \approx \frac{\Delta N}{\Delta t} \approx \frac{c_v \Delta T}{\Delta t} \tag{8}$$

4. Similarly approximate the change in density using the conservation of mass equation.

$$\frac{\partial \rho}{\partial t} \approx \frac{\Delta \rho}{\Delta t} = \rho\boldsymbol{\nabla}\cdot\boldsymbol{v} \tag{9}$$

$$\Delta\rho = \rho\Delta t(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial x}) \tag{10}$$

5. Now we solve the advection equation for density then velocity and temperature. The advection equation can be seen below. Here we use v_{avg} to advect quantities because v_{avg} is the velocity of fluid at the centre of each grid box. It is this section that could be targeted for improvement if we wished to improve our simulation accuracy. A BFECC method is recommended by Golas et al.

$$\frac{\partial \phi}{\partial t} = -\boldsymbol{v}\cdot\nabla\phi \tag{11}$$

$$\Delta\rho_A = -\Delta t(u_{avg}\hat{\boldsymbol{x}} + v_{avg}\hat{\boldsymbol{y}})\cdot(\frac{\partial\rho}{\partial x}\hat{\boldsymbol{x}} + \frac{\partial\rho}{\partial y}\hat{\boldsymbol{y}}) = -\Delta t(u_{avg}\frac{\partial\rho}{\partial x} + v_{avg}\frac{\partial\rho}{\partial y}) \tag{12}$$

$$\Delta T_A = -\Delta t(u_{avg}\frac{\partial T}{\partial x} + v_{avg}\frac{\partial T}{\partial y}) \tag{13}$$

$$u_{t+\Delta t} = u_{max,t} - \Delta t(u_{avg}\frac{\partial u_{max}}{\partial x} + v_{avg}\frac{\partial u_{max}}{\partial y}) \tag{14}$$

$$v_{t+\Delta t} = v_{max,t} - \Delta t(u_{avg}\frac{\partial v_{max}}{\partial x} + v_{avg}\frac{\partial v_{max}}{\partial y}) \tag{15}$$

$$\rho_{t+\Delta t} = \rho_t + \Delta\rho + \Delta\rho_A \tag{16}$$

$$T_{t+\Delta t} = T_t + \Delta T + \Delta T_A \tag{17}$$

6. Update pressure using the ideal gas law.

$$P_{t+\Delta t} = \rho_{t+\Delta t} R T_{t+\Delta t} \tag{18}$$

7. Repeat for next time step.

4.3 Boundary and Initial Conditions

4.3.1 Boundary Conditions

In order to make the explosion reaction at the walls realistic, we made a few choices to handle the walls. We applied the zero slip principle to our simulation, setting the velocity of the fluid at the walls of the enclosure to zero. We also set the density of the walls to be steel $\approx 8000 \frac{kg}{m^3}$, which gives us realistic reflection.

4.3.2 Initial Conditions

By varying initial conditions we could simulate different explosives. This was simply done by adjusting density and initial pressure to an individual explosives detonation pressure and density (easily found online). For our purposes, (comparing the Lagrangian and Eulerian Model) modeling a specific explosive was less important.

4.3.3 Time Step and Cube Width

Because we did not implement BFECC on our advection equation, our time steps were required to be very small. Depending on initial conditions, Δt varied between 0.1 and 1 μs. Δx was required to be $\approx 0.2m$. Using these choices, we were able to get a stable, accurate explosion for 24 ms, $\approx 10 - 20000$ time-steps.

4.3.4 Implementation

We implemented this explosion in python, running the entire simulation first and storing it in a 4-D JSON matrix. This allowed it to be read quickly by our front end, allowing our GUI to be quite responsive.

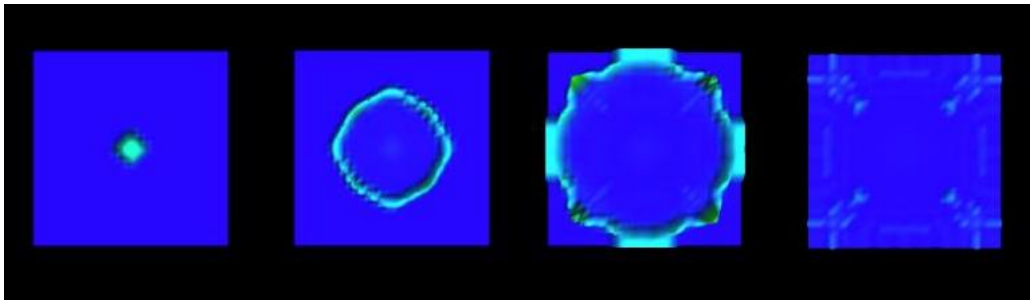

Figure 2: Propagation of the pressure resulting from a single explosion. Simulated by the Eulerian model at different moments in time

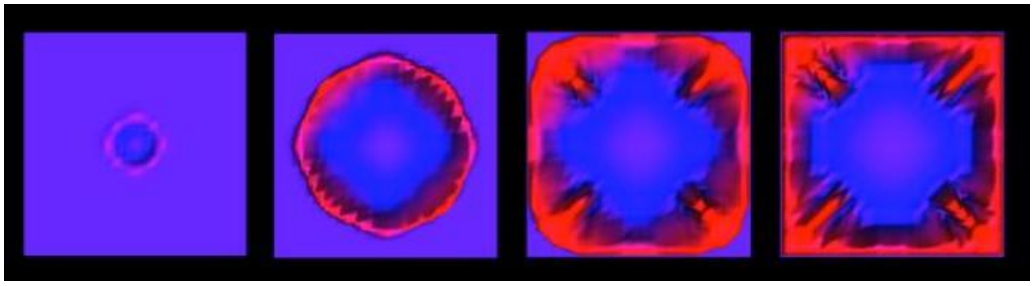

Figure 3: Propagation of the temperature resulting from a single explosion. Simulated by the Eulerian model at different moments in time

5 Lagrangian Model of The Object

To investigate the effects of an explosion on its surroundings we decided to implement a free body into our simulation to see how the explosion would effect the object's movement

5.1 Assumptions

To expediate the implementation of the free body in our explosion we made a few simplifications to the physics that governs the movement of our object

- The body is completely rigid and does not deform under the forces of the explosion
- The body is not influenced by the viscous forces of the explosion
- The body moves in the x-y plane and is not effected by gravity

In implementing a free body in the explosion we hoped to investigate the effects of a pressure potential field so it is within reason to make these assumptions

5.2 Physics

The movement of the free body depends on the pressure field created by the explosion. The pressure field can be viewed like a potential under which the object moves

5.2.1 Kinetic Energy

In this space we can have three components to the kinetic energy, motion in x, y and rotational motion, with respect to our center of mass coordinates

$$T = \frac{1}{2}m\dot{x}^2 + \frac{1}{2}m\dot{y}^2 + \frac{1}{2}I\dot{\theta}^2 \tag{19}$$

5.2.2 Lagrangian

$\mathscr{L} = T - U$
The Kinetic energy T has been defined above so now it is necessary to define the Potential Energy U

Since the Lagrange equation is defined

$$\frac{\partial \mathscr{L}}{\partial q} = \frac{\partial}{\partial t}\frac{\partial \mathscr{L}}{\partial \dot{q}} \tag{20}$$

The Kinetic energy is only dependent on the time derivative of our generalized coordinates so the generalized force $\frac{\partial \mathscr{L}}{\partial q}$ is dependent only on our potential

Therefore it isn't necessary to define our potential, rather we can just find the force (and torque) on the object resulting from the pressure field around it. This is done via finding the pressure on a finite surface of our object and the force normal to that.

Solving the Lagrange equation yields

$$\Sigma F_x = m\ddot{x} \tag{21}$$

$$\Sigma F_y = m\ddot{y} \tag{22}$$

$$\Sigma T = I\ddot{\theta} \tag{23}$$

Which is consistent with the Newtonian method for finding the motion of an object

5.3 Implementation

The implementation of our object was done in Python.
It is represented by by a matrix of ones and zeros, ones indicating where the object is and zeros where it is not. This ends up depicting a "pixelated" view of our object.

5.3.1 Center of Mass

To find the center of mass we summed over the matrix:

$$m' = \sum_{i=0}^{N} \sum_{j=0}^{N} M(i,j) \tag{24}$$

$$X_{com} = \frac{\sum_{i=0}^{N} i \sum_{j=0}^{N} M(i,j)}{m'} \tag{25}$$

giving the index of the x center of mass
The same can be done for the Y_{com} summing over the rows instead

5.3.2 Forces

Since our "object space" shares the dimensions of the explosion we can find the pressure at any surface in our object given its location by referencing the same location in explosion space

A surface on our object can be defined as the line drawn between 2 matrix elements on the outside of the object. We then can look at the elements touching this surface and look for the pressure there to find the forces. The normal to these surface determine the direction and location compared to the Center of Mass determines the torque.

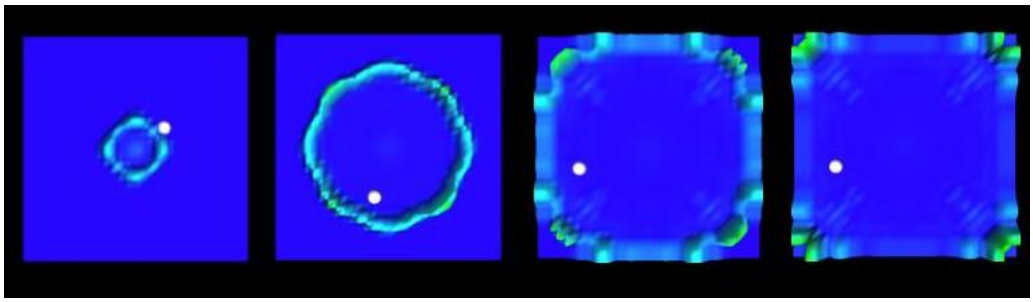

Figure 4: Movement of a spherical object resulting from the propagation of the pressure of a single explosion at different times.

5.4 Limitations

- Due to the complexity of our explosion simulation, the space could not be bigger than 32x32, this lead to the object only consisting a 3x3 square. Due to this not transforming well under rotation, ultimately rotation was not shown in our simulation although it was calculated
- Similarly due to the complexities of the fluid simulation, the time shown is on the order of milliseconds. To actually show the object moving its density was reduced to just over that of air. A heavier object, though still having high velocity by the end of our simulation, could not be seen to have moved by the time the simulation had completed

6 Simulation

In order to simulate the explosion, we set up a grid, with temperature, pressure, density and velocity known at each square of the grid. The implementation of our simulation can be divided into two parts:

6.1 Front-End

The front-end of our simulation comprises the part of the simulation that the user will be able to interact with. This is done through a website made up of the following:

- JavaScript
- HTML
- THREE.js Library

HTML and JavaScript are used to build the website that the user can interact with. THREE.js is a graphics library for JavaScript that allowed us to easily create 3D graphics using client-side GPU rendering.

6.2 Back-End

The back-end end of our simulation is where data processing is done. We set up a data server that allows us to send pre-generated data to the client, rather than having to process data as the website runs. This server architecture is set up using the following:

- Python
- JSON
- NumPy Library

Python and the NumPy Library are used to to code the data processing algorithms for our simulation. This data is formatted into a JSON file, which is sent to the client to be parsed.

7 Conclusion

Through calculations and simulations we concluded that the Eulerian model is computationally more plausible than the discrete Lagrangian approach. We would require a deeper understanding of Thermodynamics and Statistical Mechanics to be able to accurately represent the potential energy function of individual particles in the system in order to create a Lagrangian model. In the future, we should consider the continuous Lagrangian approach, as proposed by Prof. Sigurdson.

8 References

Explosion Simulation using Compressible Fluids by Abhinav Golas et al, link: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=