Physics Sandbox

Software Design Document

Jack Ferguson

5/3/2021

CS 225, Spring 2021

Embry-Riddle Aeronautical University

Daytona Beach campus

1 Aerospace Boulevard

Daytona Beach, FL 32114

**Introduction:**

The program is an interactive physics modeling simulation. It allows the user to drag and drop different particle types (magnets, fixed collision bodies, gravity affected particles) into a sandbox type environment. The program has toggling physics, meaning the simulation can be stopped and continued as the user adds more particles.

**Problem Description:**

The particles in the sandbox have hitboxes that interact with each other, and momentum transfers. This program allows the user to save the state of the sandbox as a file, and import files to explore previously created sandboxes. The user can input values of gravity, magnets, mass, and size of particles. This type of simulation is ideal for computers. It contains rapid calculations and the behavior of the particles is described by a consistent set of equations. This idea can be applied to space travel simulations, which are becoming more and more important as humanity looks towards space for expansion.

The main problems in the program are all physics related. The interactions of the particles with each other and the environment are complex; multiple forces will be pulling different particles in different directions with different strengths. In addition, because the particles will bounce off each other and walls, they will need momentum stored. The particles will store their position, and change their position by their velocity, which is constantly calculated by this equation : v = v + at. The time in this equation and all future equations is the time passed between frames. The acceleration used in previous equation will be calculated by the equation a = F/m, where m is the mass of the particle and F is the net force acting on the particle. The net force will be calculated as a summation of all force vectors acting on the particle: FNET=.

**User Stories:**

1. As a user, I want to see the simulation in real time
2. As a user, I want to pause and play the simulation with a button
3. As a user, I want to add particles in real time
4. As a user, I want to control the size of the window
5. As a user, I want to see particles apply forces on each other
6. As a user, I want to drag and drop particles
7. As a user, I want to customize gravity in the GUI
8. As a user, I want to set particle attributes
9. As a user, I want to save the starting state of the sandbox
10. As a user, I want to save the current state of the sandbox
11. As a user, I want to create particle templates
12. As a user, I want to save particle templates

**Problem Solution:**

Each particle will store several variables, each of which describes a part of the particles motion. Each particle will store its position as a Point, which has an x and y component. This position will be modified by a velocity Vector, which will have an x component, a y component, and an angle and magnitude calculated from the mentioned components. This velocity will be modified by an acceleration Vector, which will be modified by the net force Vector. The net force Vector will be calculated from an ArrayList of force Vectors which tracks the forces currently acting on the particle.



The above UML diagram describes the classes that will be used and their interactions. The Kinematic class handles everything related to motion. This class utilizes the Point and Vector classes, which exist to simplify the math aspect of the Kinematic class and because they provide excellent generic functionality for Vector math and point storage. The Particle class extends the Kinematic class and adds Color and Name functionality. The SpecialParticle class also extends the Kinematic class, adding different Color and Name functionality, and also overriding the Kinematic updateVelocity(double) function. The user will be able to control the radius, color, and force strength of each Particle and SpecialParticle. The Simulation class controls everything. It aggregates the Kinematic class, and manages the instances of Particles as well as the simulation flow.

**References:**

I use the AP Physics C equations sheet as a condensed source of the physics equations I need

<https://apcentral.collegeboard.org/pdf/physics-cm-equations-sheet-2020.pdf>

**Appendices:**