**Physics Based Curling Simulation**

Jeremy Therrien – [therrj@rpi.edu](mailto:therrj@rpi.edu)

Mark Aversa – [aversm2@rpi.edu](mailto:aversm2@rpi.edu)

**Abstract**  
This paper presents a method for creating system for modeling physical interactions between objects. A physics based system is also presented to accurately handle properties affecting objects such as friction, linear and angular momentum.

**1 Introduction**  
The system we developed will be used to create an accurate physics based Curling simulation. Past simulations have used various formulas to determine the movement of a curling rock where we use a purely physics based approach.

**2 Environment**

The curling rocks are represented by a collection of flat cylinders and the rink is represented by a rectangle.

**2.1 Curling Rocks**

Each curling rock has a vector for its position and a vector for its velocity. The diameter of the rock is 11.4 inches (28.956 cm) with a circumference of 35.814 (90.968 cm) inches, a height of 7.0367 inches (17.873 cm), and a weight of 18kg (39.683 lbs) as specified by the World Curling Federation. [1]

**2.2 Rink**The rink is represented by a rectangle with dimensions specified by the World Curling Federation (See Figure 1 for more details and terminology). The width of the rink is 15.5 ft (4.724 meters) and the length is 150 ft (45.720 meters). The distance from the backboard to the hack is 6 ft (1.829 meters). Hack specifications are ignored since they are not used. The distance from the hack line to the back line is 6 ft (1.829 meters). The distance from the back line to the tee line is 6 ft (1.829 meters). The distance from the tee line to the hog line is 21 ft (6.401 meters). The distance between each tee line is 72 ft (21.945 meters). [1]

Where the center line and tee line meet is known as the button. From the button there are four concentric circles placed at each end of the rink with the outer having a radius of 6 ft (1.829 meters), the next circle a radius of 4 ft (1.219 meters), the next circle a radius of 2 ft (0.610 meters) and the innermost circle having a minimum radius of 6 inches (15.25cm). [1]

**2.3 Broom**

The broom is used to sweep the area in front of the curling rock reducing the coefficient of friction between the rock and the ice. As a result of the lower coefficient of friction, the rock will travel in a straighter path and will curl less. In our simulation, only the shot rock can be swept.

**3 Interactions**

There are three distinct interactions that we are concerned with in this simulation: transfer of linear momentum between rock and rock, frictional forces between rock and ice, and transfer of angular momentum between rock and rock.

**3.1 Rock/Rock Linear Momentum Transfer**

For determining when two rocks collide, we use the same equation as when two spheres collide. The simulation uses an implementation of the algorithm described in “2-Dimensional Elastic Collisions without Trigonometry” [1]. This algorithm allows the calculation of new velocity and direction vectors after a collision between spheres. First, we calculate the normal between the centers of each rock:

*N = p2 – p1*

From there the tangent vector is:

*T = <-Ny, Nx>*

We then calculate the unit values of the two vectors. Next we compute each rock’s normal and tangent velocity using the dot product:

*V1N = UN • V1*

*V1T = UT • V1*

The tangent component of the velocities remains the same and we calculate the new normal component of the velocities:



The scalar value for the normal and tangent components are then multiplied by the unit normal and unit tangent, respectively, to find the new normal and tangent velocity vectors. The normal and tangent velocity vectors are then added together to create the final velocity vectors for each rock.

**3.2 Rock/Rock Angular Momentum Transfer**

Determining the angular momentum of the rocks when they collide is necessary in curling because it alters the trajectory of the rock on the ice. For determining the angular momentum we first determine the two colliding vectors and their unit value:

From there we determine the scalar product of the two unit vectors:

The trigonometric function, arccosine is then used with the dot product to find the angle between the two vectors:

A torque vector for the system is then calculated and the magnitude of the vector is taken:

The new scalar quantity is then split amongst the colliding rocks and added to their angular velocity.

**3.3 Physics**

The physics modeled in the system will contain two equations.

The first equation being that of the position of the rocks based on their velocity and acceleration due to friction.

The second equation will be the change in velocity due to the friction.

In the simulation, acceleration will always be negative, since the only energy that is being added to the system is the initial throw of the curling rock.

**3.4 Friction**

The frictional force applied to the rock is variable in the system, due to sweeping the ice with the broom. Curling ice is often sprinkled with water before a curling match to give the ice a pebbled surface. This allows the rock to grip the ice and curl better. The broom smoothes the ice in front of the rock and lowers the coefficient of friction by smoothing out these pebbles. With a lower coefficient of friction, the rock is unable to curl as much and travels in more of a straight line at a faster velocity.

**4 Time**The simulation must move through time. The rocks have velocities and positions that change over time according to the laws of motion. This raises a problem when time passes too quickly and the rocks clip into each other before a collision is detected.

**4.1 Collision Backtracking**

Physic simulations usually implement a time step value that progresses the system a certain amount of time into the future and calculates new positions and velocities for each object in the scene after the time has passed. Our simulation does this as well, and because of this we ran into a problem where the collision between rocks was registered too late and the rocks would clip and “stick” to each other. In order to solve this problem we implemented a simple backtracking algorithm that allowed us to move the rocks back to the exact point at which they collided.

We first computed the distance between the two objects:

After that we found the distance the rocks had traveled into each other based off of the radius of the rocks:

We then took that value and moved the rocks apart that distance so that they did not stick together.