Physically Based Simulation – Assignment 1

Compiling & Running:

The code is portable, so it should be easily ported to Linux. You need FLTK, and OpenGL libraries and headers to compile. For running, FLTK DLLs should do, since OpenGL DLLs are supplied with Windows

Part 1 : Ballistics Simulation

The first part implements a basic ballistics system in 3-D. The source (gun) and target (unlucky), are both in the X-Z plane, and Y is the up direction. The parameters for controlling the gun are the Azimuth and Elevation angles, both in degrees. The reference direction for the azimuth is the X-axis or North. Azimuth must be between 0-360 and Elevation 0-90. For the projectile being fired, Projectile mass, Powder mass, and powder power are the parameters. Powder power * Powder mass is the initial impulse supplied to the projectile. Gravity can be tinkered with for fun, but must remain negative for things to return to the ground ☺

Euler, Midpoint and Runge-Kutta 4 integration methods can be used.

Performance & Accuracy:

- Euler Methods take on an average: 0.002 seconds for time step 0.1 sec for the entire simulation, in 16 iterations
- Midpoint Methods take approximately: 0.0028 seconds for the same parameters, taking 15 iterations to complete
- RK4 Methods take 0.0035 seconds for 15 iterations on the same parameters

Performance wise, it seems clear that Euler Methods are the best with regard to performance, while RK4 offers most accuracy. However, Midpoint seems to be the best tradeoff giving visually indistinguishable results to RK4 while taking lesser time.

Stability/Accuracy:

Euler and Midpoint methods give similar results when using time steps of the order of 0.001 seconds. However, as the time step is increased towards 0.1 seconds, the results rapidly deviate from each other, remaining stable nonetheless. RK4 and Midpoint maintain similar results for time steps up to 0.5 seconds, after which they start to differ. The observed results are as expected from theoretical results. Euler methods start to diverge at larger time steps as the values start reaching the end of the stability region of the method.

Part 2 : Spring Mass simulator:

The spring mass simulator is a simple simulation where a mass is hung at the end of a spring, with coefficient Ks, and damping coefficient Kd.

Performance/Accuracy/Stability:

The Spring Mass simulation offers a better insight into the stability region of the integration methods. While the Euler method diverges rapidly for mid level time steps too e.g. 0.01 seconds on Ks=50 for a mass of 10 kg. Midpoint maintains stability uptil 0.5 seconds. RK4 maintains stability for even larger time steps well beyond the time step when we start to lose motion resolution