Andrew Seba  
Math/Phys 191  
5/28/16

Lab Report

# Introduction

For many modern games there will be movement on the screen in some form. To describe motion over time we need to be able to access velocity, acceleration, and the change in time. In this lab we did a linear approach to a ball being thrown up in the air. This motion can be applied to many different situations for characters, projectiles, and obstacles.

# Methods

In the lab we are given an initial velocity positive of 49.00 m/s upwards and an initial height of 2 meters and must output the current position, current velocity, and the current time. Forward Euler is the fastest and most accurate method we can use to simulate this movement. For position we can use:

The for our program is our time-step increment, and the is how far the object has traveled during the time step. We need to also update our change in velocity so that the ball can actually change speed instead of going off forever in the initial direction. Using the following we can calculate that change:

For the first couple procedures we only are dealing with the constant acceleration of gravity. So for this equation we are just adding to the velocity the acceleration multiplied by the time step. This means that if the ball is initially going up we are subtracting from the upward velocity by gravity which will slow it down till the point where it stops moving upward and changes to move downward with gravity.

We can compare this to an analytical approach using more math with precise results like the following equation for its position:

For one of the procedures we need to add in some air/wind resistance. To account for this we can update the acceleration by subtracting the gravity acceleration by the wind coefficient then multiply by the velocity so that the faster the object moves the more resistance is apparent.

# Results

These results are showing different time-step increments for the equations from the methods. The yellow line is the analytic data (being very accurate) and as you can see the larger the time-step the less inaccurate it becomes.

This graph shows the effects of wind resistance on the acceleration on the ball. It slows it down going up and slows it down on its way down to the ground.

# Conclusion

The Forward Euler method is a great fast way to calculate the motions of objects and represents realistic data as long as the time step between movements is smaller than 1 second, anything larger and the motion becomes unrealistic. This seems to be a great starting point for simulating motion for all directions.

# Post-Lab

1. Were there any noticeable differences in the run-times when using a different time-steps? If so, why would this ever be an issue?

When the time step was less than a tenth of a second I could see that the output took a little bit longer to show up than doing a tenth of a second or higher. This could make the in game physics slow the actual game loop down and cause the game to be less playable than before.

1. When comparing the results of the Forward Euler calculations to the analytic calculations, which time-step did best? Which did the worst?

The tenth of a second ended up being best of speed and performance than the rest. The worst accuracy was the 1 second time increment.

1. Is there any reason why you would ever select a time-step that you give you anything less than the best fit to the analytic solution?

Maybe for non-intractable objects just for visual effect maybe particle effects that need to look pretty but won’t affect gameplay at all.

1. Wind resistance is a force that causes an object in motion to slow down over time much like friction. Wind resistance, however, produces an acceleration that is proportional to the velocity rather than a constant acceleration. The faster an object is going, the stronger the force of wind resistance is. Explain the benefits of using Forward Euler to model motion under the influence of wind resistance as opposed to solving for an analytic solution.

I think that one of the benefits would be that it’s a little less computationally expensive, and that we don’t need it to have perfect orbit like motion when we are doing wind resistance to make it realistic.

1. Where else, other than a ball moving through the air, might you employ a wind resistance-like behavior?

Under water or any substance you can move through.