**IDS 6938- Simulation Techniques**

# **Instructor: Dr. Joseph Kider**

**Assignment 1**

The Jello Cube (Continuous Simulation Assignment)

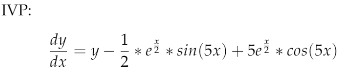
Submitted by

Kattoju Ravi Kiran

NID: ra977917

Part 1: Numerical Analysis of Integration Function

Given the following IVP:



With initial conditions y(0)=0

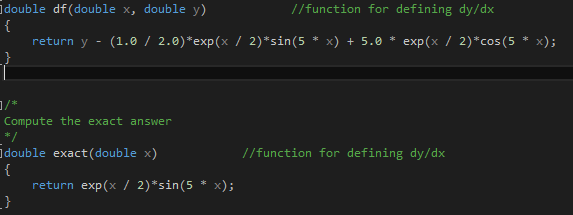
1. Solving for the exact solution:

Solving the above IVP by taking integral to get exact function gives

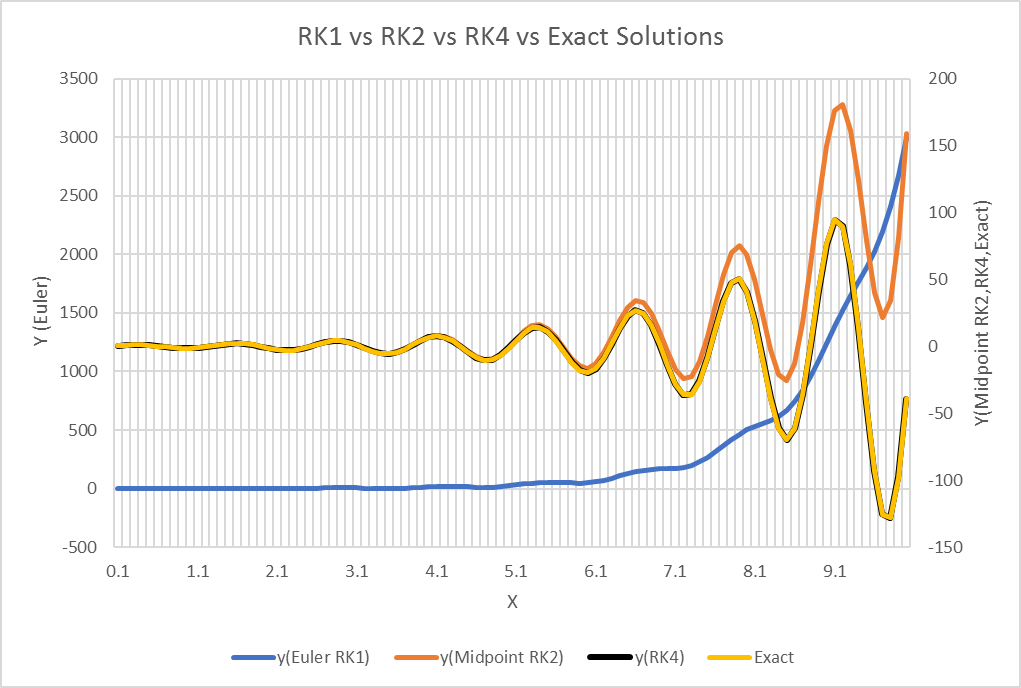
y= exp^(x/2)\*Sin(5\*x)

1. Translating formulas to **df** and **exact** functions:

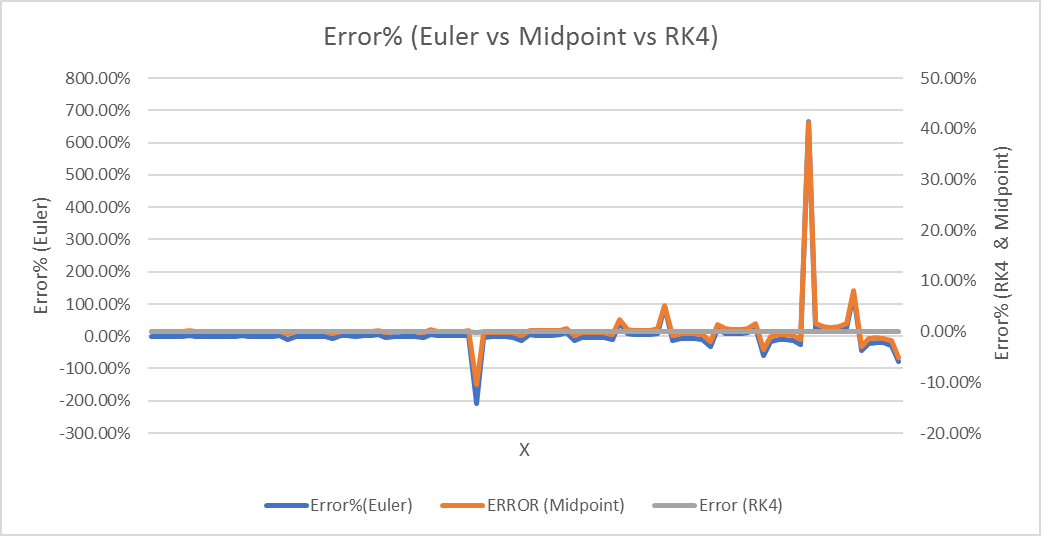
* The mathematical formulas are translated to computational code in the functions ‘**df**’ and ‘**exact’.**
* **df** function defines the original IVP given above and **exact** function defines the solution of the IVP given y(0)=0.



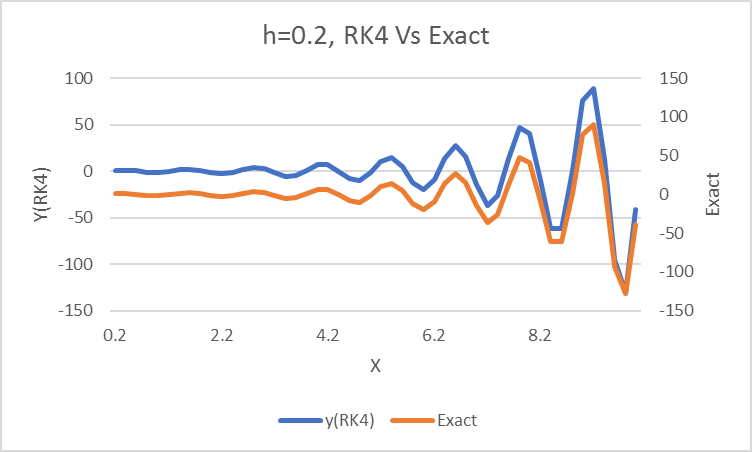
1. Running Simulation with RK1, RK2, RK4

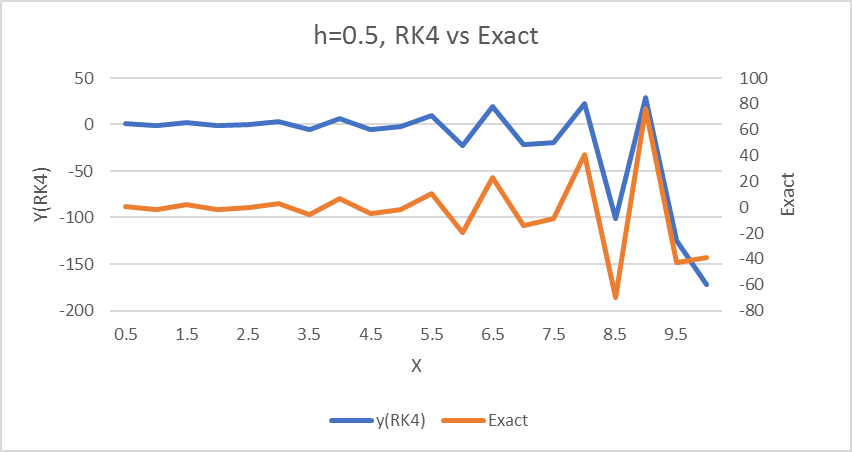


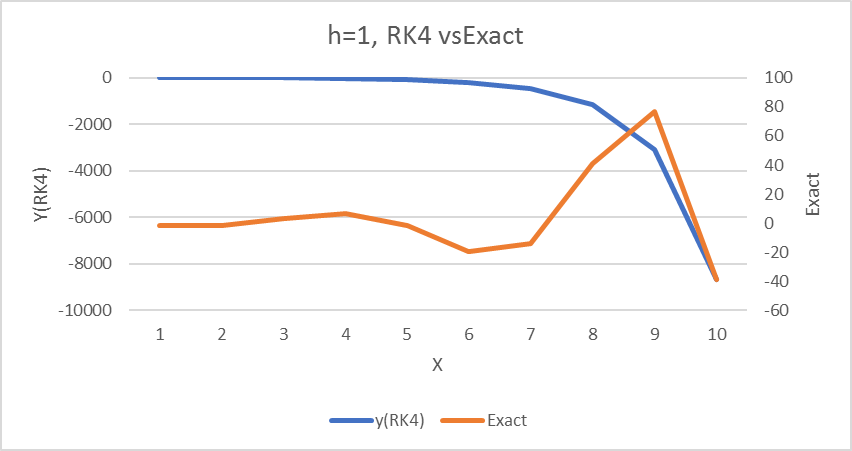
1. Plot Error percentages for RK1, RK2, RK4



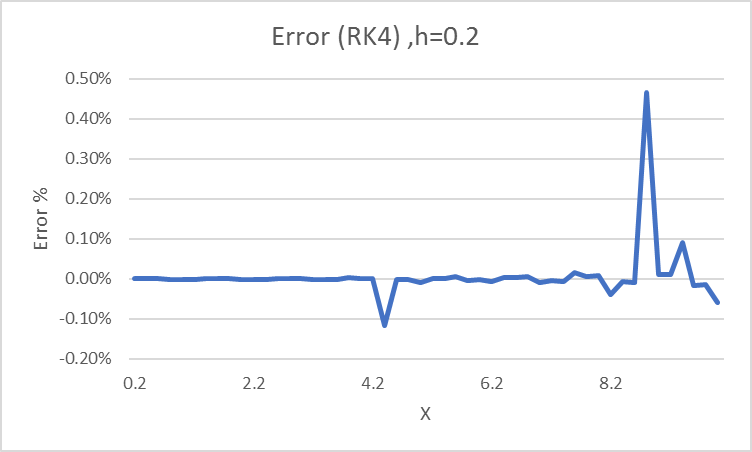
1. Varying Step Size and RK4 Plots- h-0.1, h=0.01, h=0.001

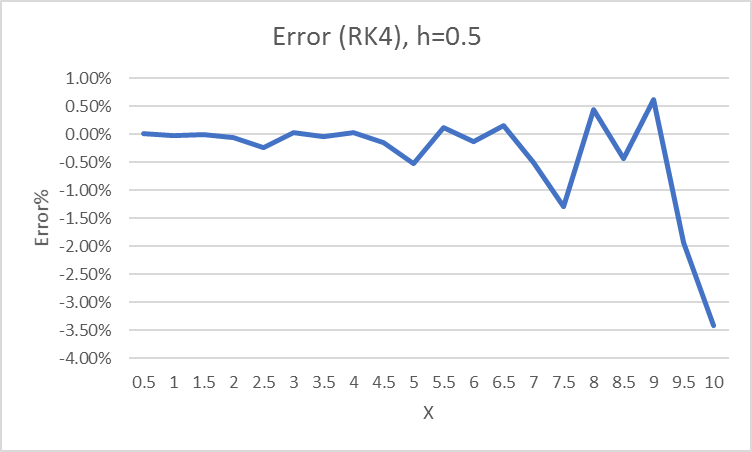


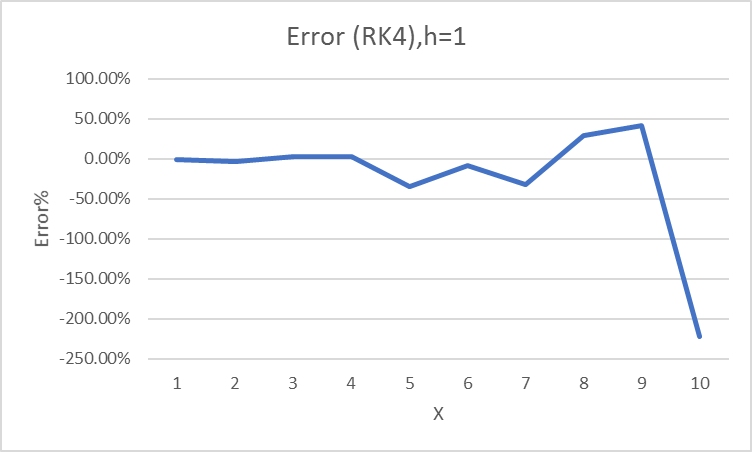




Error Plots







1. Analysis
   1. Effect of varying integration methods on accuracy

The RK4 integration method provides better accuracy compared to the Euler and the Midpoint integration methods. The Euler provides lowest accuracy amongst the three integration methods.

* 1. Effect of increasing x value on accuracy

With increasing value of x, the accuracy seems to drop. Though mildly affecting the stability of the RK4 integration method, the accuracy remains better than the Euler and the Midpoint integration methods.

* 1. Effect of varying step size on accuracy

The increase in step size has greatly affected the accuracy of the RK4 integration method. /the error % remain below -0.1% for step sizes of 0.2 and 0.5. however, increasing the step size to 1 has resulted in error% of approximately -200%.

* 1. Most accurate method

From the results, the most accurate integration method seems to be the RK4 method. This may be due to the fact that it computes and takes in to account RK1, RK2 and RK4 values in to consideration while calculating the value of y. The RK4 method produced results in close proximity to the exact y values. The error % with different step sizes and increasing values of ‘x’ provided better accuracy compared to the other 2 integration methods.

Part 2: Programming Jello Cube

* Integration Methods
  + Euler Integration

The JelloMesh::EulerIntegrate() function is called when the integration type is set to Euler Integration. This function obtains the ‘dt’ function, particle grid information and computes the position and velocity for each particle in the grid.

* + Midpoint Integration

The JelloMesh::MidpointIntegrate() function is called when the integration type is set to the Midpoint integrate method. This function obtains the ‘dt’ function, the particle grid information and uses it to calculate the forces at the midpoint and then the K1 constants. The position and velocity of each particle are adjusted based on the forces computed.

* + RK4 Integration

The JelloMesh::RK4Integrate() function is called when the integration type is set to the RK4 integrate method. This function obtains the ‘dt’ function, the particle grid information and uses it to calculate the forces at each stage using K1, K2, and K4 constants. The position and velocity of each particle are adjusted based on the forces computed using K1 at stage 1, K2 at stage 2, K4 at stage.

* Particle Forces
  + Compute Forces

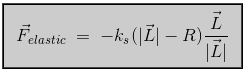
The ComputeForces() function computes the external forces on the particles in the jello mesh grid. It also computes the spring force and the damping force between particles given by Hooke’s Law.

Spring Force equation:

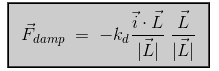
F= -Kx, where k is the spring constant and x is the spring displacement.

In this case spring force has 2 components, Elasticity given by Ks and Damping given by Kd.

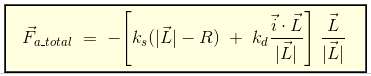
The elasticity on between 2 particles is calculated using the formula :



The damping force between 2 particles is calculated using the formula:



The total force is then calculated which is the sum of the 2 forces. This force is applied to one particle and an exact counter force is applied to the other particle to balance the forces.

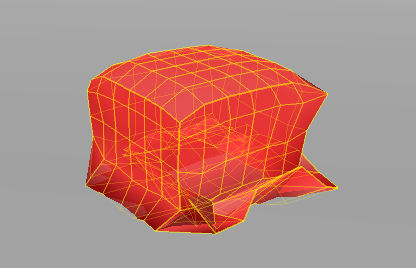
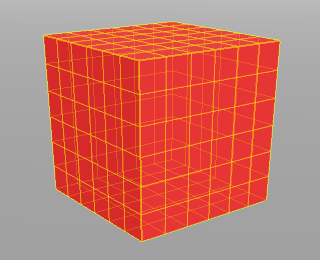


* Collision and Penetration Detection
  + Check for collisions

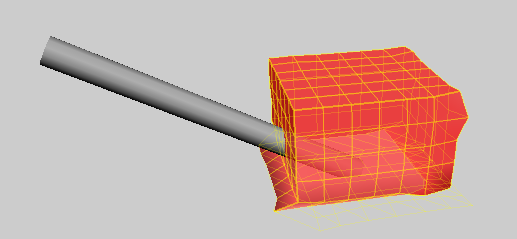
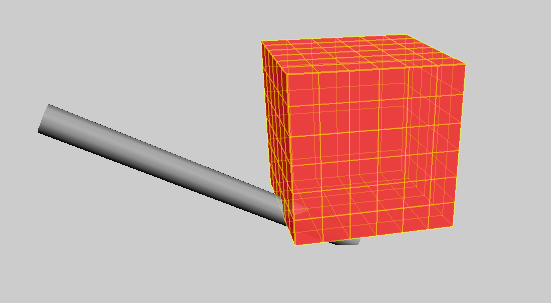
The Check for collisions function checks to see if the jello mesh has collided with the ground or objects in the world. My function has been able to detect contact and collisions both with the ground and the cylinders in the environment.

* + Floor Intersection

The floor intersection function gets the particle , intersection point information and uses it to detect if the jello mesh has made a collision ( entered the threshold region above the ground) or contact with the ground (shown in figure below). I set my threshold to 0.01 above the ground and the collision system detects entry of the Jello mesh in to the collision threshold region and starts computing forces.



* + Cylinder Intersection

The cylinder intersection function gets the particle , intersection point information and uses it to detect if the jello mesh has made a collision ( entered the threshold region above the ground) or contact with the cylinder (shown in figures below). The collision system detects jello mesh collision and contact with the cylinder. How ever, it is failing to compute forces and explodes.

* Collision and Penetration Response
  + Resolve Collisions

The Resolve Collisions function was used to calculate the resolution forces upon collision of the jello mesh particles with the ground threshold region. A penalty force was calculated using the penalty elastic force constant and the penalty damping force constant and applied to the particles in the direction of their velocity normals.

* + Resolve Contacts

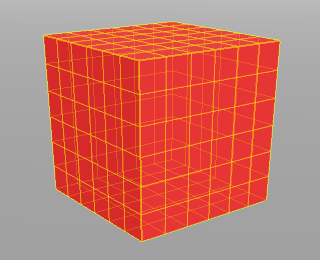
The Resolve Contacts function calculates the particle force that needs to be applied on each particle that has made contact by updating the particle position, velocity and forces subjected on the particles. The particle velocity is given by the equation:



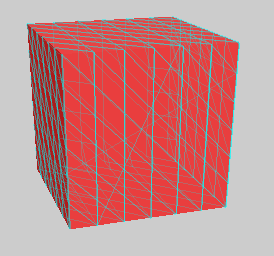
* Extra springs

I added shear and bend springs to the existing jello mesh. Structural springs were already present. Bend springs were added between alternate particles on the grid and shear springs were added between diagonal neighbouring particles.

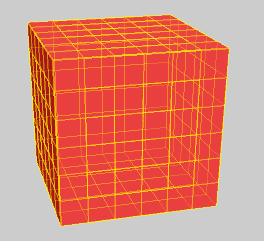
* + Structural Springs



* + Shear Springs



* + Bend Springs



* Extra Features
  + Video

Part 3: Written Questions

1. Effect of Ks and Kd on the Jello

The coefficient of elasticity Ks dictates the stiffness of the springs in the jello cube. The Ks is directly proportional to the stiffness Increasing the Ks, increases the stiffness and prevents increased displacement of the springs in the mesh on collision and impact.

The damping coefficient Kd is a dynamic spring constant and is responsible for damping or reducing the oscillations of the spring in the jello. Increase in Kd will damp the oscillations faster and vice versa.

1. Benefits and Drawbacks of Collision System

The collision system tries to simulate impact and collision of jello with other objects using a mass- spring system to give a realistic solution. However, there are circumstances where the jello exhibits unusual behavior such as bounciness, wobbling and splattering on impact with objects or the ground. These problems could be solved with finetuning and tweaking of the various spring constants for the different springs employed in the system. The benefits of this collision system may be the principle of the mass spring equations employed which may have many other applications such as designing bullet-impact simulations, Ocean wave simulations for designing wave energy harvesting systems, simulating suspension bridges to determine stresses etc.

1. Example of Mass- Spring Simulations

As mentioned above, a few examples of mass spring simulations would be simulating bullet impact on different materials for design of bullet proof vests or assessing damage potential of a bullet or a missile. It could be used to simulate wave energy in the ocean to help design wave energy harvesting devices. Another example of Mass-spring simulations could be the determining stresses in suspension bridges.

1. Realistic Behavior? Integration method

The jello explodes easily on collision and impact with objects and the ground. How ever the jello mesh has shown partially realistic effects while testing with different spring parameters with the RK4 integration method. Though the Euler and the midpoint method simulate the jello, the jello cube explodes inevitably.A more realistic behavior may be possible if the grid size is increased to have more particles and compute forces more accurately upon impact and collision.

1. Simulate Water

The same mass- spring collision system could be used to simulate water or at least the wavy nature of water and splashes. Though computationally expensive, with higher spring coefficients Ks and Kd and a large number of water particle droplets a realistic simulation of water behavior may be achieved. The efficiency of this system would depend on the Particle size, number, physics involved in particle dispersion, forces and ingression in to surrounding objects.