AMS320HW3REPORT

william.zhang

April 2019

1 Problem Description

For this problem we wish to solve a modified formula ODE of newton law of cooling in order to calculate when the person died. We need to find a way to find to solve

$$\frac{dT}{dt} = k * (A - T^{0.999})$$

Where A is the temperature of the room and T is the currnt temperature at time T. We will use forward euler method to approximate when his body will reach 44.44

2 Algorithm

The problem with this ODE is the fact that we do not know the constant cooling factor. So we must find a way to solve this. Now there was no procedure to solve for K other than closely approximating K via analytically. We get a K of -0.0055046671 per minute. Now the entire algorithm
The following algorithm is the representation of the ODE and maiin methods

Algorithm 1: Newtonlawofcooling(Time,Temperature)

return $-0.0055046671 * (Temperature^{0.999} - 34.44$

Algorithm 2: Main()

```
h = 1
deltabeforedeath = np.log((np.power(91.11, 0.999)- 34.44) /
  (np.power(97.88,0.999) -34.44)) / -0.005504671
print(deltabeforedeath)
print("This guy died at " + str(11) + ":" + str(39.466603) )
initialtemperature = 97.88
initialtime = 0
while initialtemperature >= 44.44: do
    initialtime += h
    initialtemperature += h *
      newtonlawofcooling(initialtime,initialtemperature)
return (initialtime)
```

3 Results

The program was written in python3. The person died at 11:39:46,00 PM. In order for his body to reach 44.44 it will take him about 338 minutes for his body temperatue to reach 44.44f.

4 Analysis of problem/Algorithm

We use the forward euler method to calculate the time it takes to reach 44.44 degrees farenheight. The number of iterations can increase quadratically if we decrease the time step.

5 Problem Description

For this problem we wish to calculate the total time a bullet travels in the atmostsphere with a initial velocity of 300 and total distance traveled.We also account for wind resistance with a ode. The problem has been modified such that the system is a non a linear system. The ODE for the bullet going up is

$$m * \frac{dv}{dt} = -F_g - F_r$$

and ODE for the bullet going down is

$$m * \frac{dv}{dt} = F_g - F_r$$

6 Algorithm Design

In order to calculate the total distance. We must know when the bullet reaches zero velocity. Thanks to the forward euler method we do not need the velocity function. To calculate distance we do not need the velocity function at time T. We only need to add the distance it travels every 0.1 seconds. This is known as the reiman summ process. I have 3 methods. The main method executes the forward euler method and the equations of the ODE

Algorithm 3: Bulletup(time,velocity)

return $-9.8 - (0.00911 * np.power(velocity, 0.999))/0.1$

Algorithm 4: Bulletdown(time,velocity)

return $9.8 - (0.00911 * np.power(velocity, 0.999))/0.1$

Algorithm 5: Main()

```
vinitial = 300
timejump = 0.001 the timejump
timeinitial = 0
totaldistancetraveled = 0
while vinitial + timejump * bulletup(timeinitial, vinitial) >= 0 do
    timeinitial += timejump
    totaldistancetraveled += timejump * vinitial
    vinitial += timejump * bulletup(timeinitial,vinitial)
down = 0
while down < totaldistancetraveled do
    timeinitial += timejump
    vinitial += timejump * bulletdown(timeinitial,vinitial)
    down += timejump * vinitial
totaldistancetraveled *= 2
```

7 results

The bullet traveled close to about 3451 meters and it stayed in the air for 3400.The reader should run the program to see the numbers

8 Analysis

This problem we use the forward euler method to calculate the total time it will take for the bullet to reach the zero velocity. However, we cannot make the assumption that the time it takes to go up will be the same as going down. That is why I have to apply forward euler method twice on 2 different ODE.