Project 1

Introduction:
The purpose of this project is to use PyTorch neural network and learning algorithms to determine a rocket's landing path by controlling thrust of the booster. The process to do this starts with setting the system state parameters, implementing the dynamics of the rocket, and using a controller for the thrust action so it can reach optimum state.

Dynamics:
Rocket state x(t) is defined as $x(t) = [dy(t), vy(t), dx(t), vx(t), \theta(t)]^T$

- $dy(t+1) = dy(t) + vx(t) * \Delta t$ - Position y
- $vy(t+1) = vy(t) + a1(t) * \cos(\theta(t)) * \Delta t$ - Velocity y
- $dx(t+1) = dx(t) + vx(t) * \Delta t$ - Position x
- $vx(t+1) = vx(t) + a1(t) * \sin(\theta(t)) * \Delta t$ - Velocity x
- $\theta(t+1) = \theta(t) + a2(t) * \Delta t$ - Theta

Note: Δt is time interval

Controller a(t) includes a1(t) for acceleration and a2(t) for angular velocity.

Closed loop controller: $a(t) = f\theta(x(t))$

Loss as a function of state and controller: L(x(t), a(t)) is set equal to 0 for all steps from t = 0 to T-1 with the final time step $L(x(T), a(T)) = dy(T)^2 + vy(T)^2 + dx(T)^2 + vx(T)^2 + \theta(T)^2$. The loss is guiding the rocket to reach x(T) = 0.

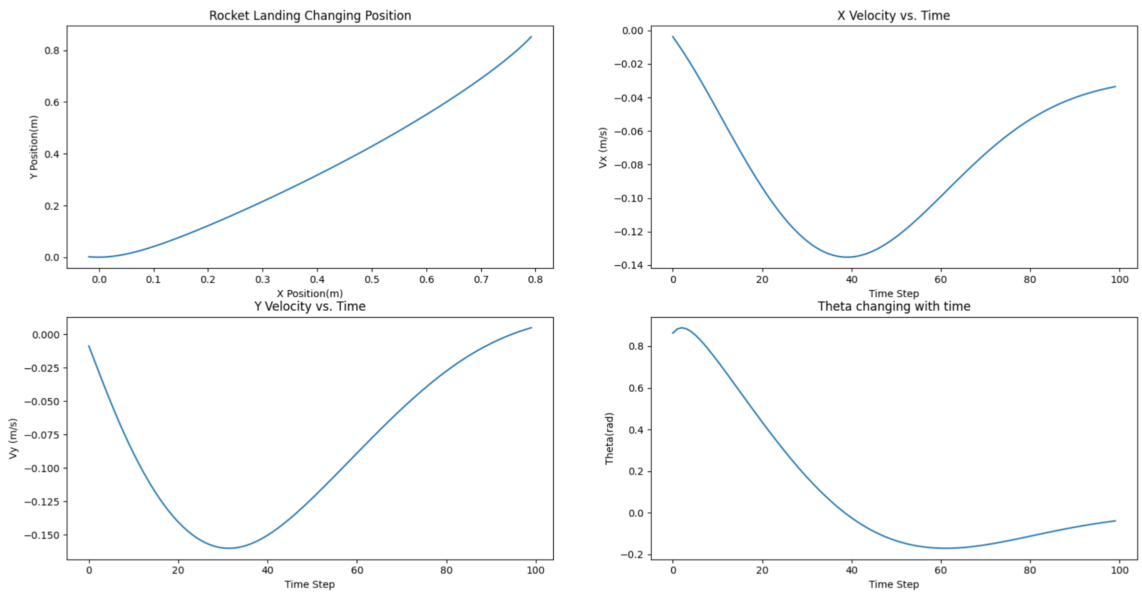
Optimization of objective function:
$$\min_{\theta} \| x(T) \|^2$$
Subject to the dynamics above.
It is a constrained problem but because each concurrent time step is a function of the previous step it is basically unconstrained with respect to θ.

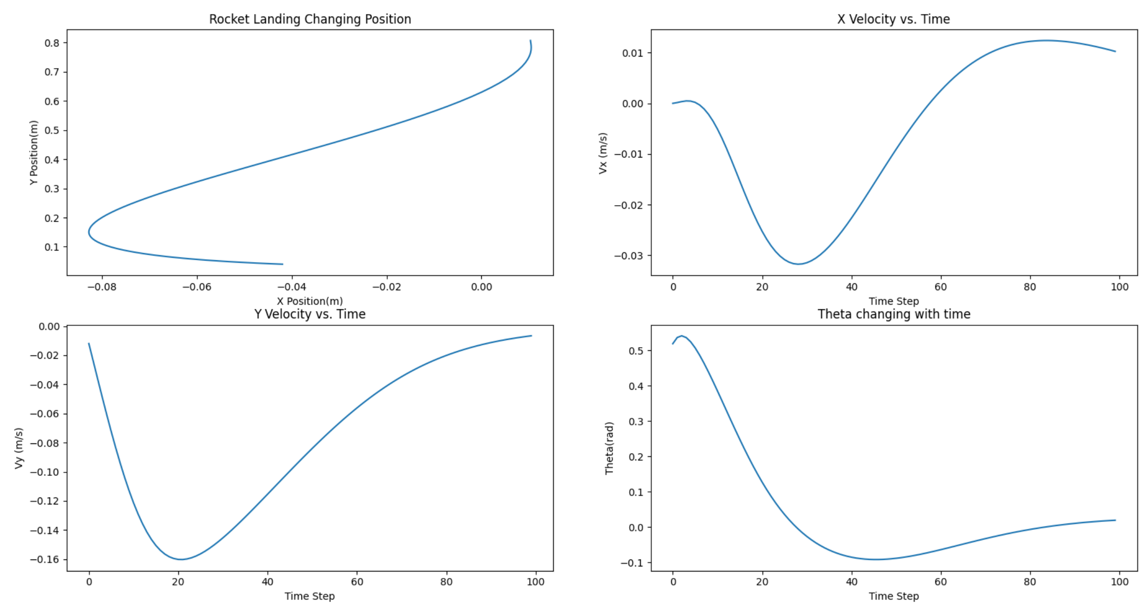
This problem was coded in Pycharm using pytorch which is necessary for building the forward
Code:
In Github

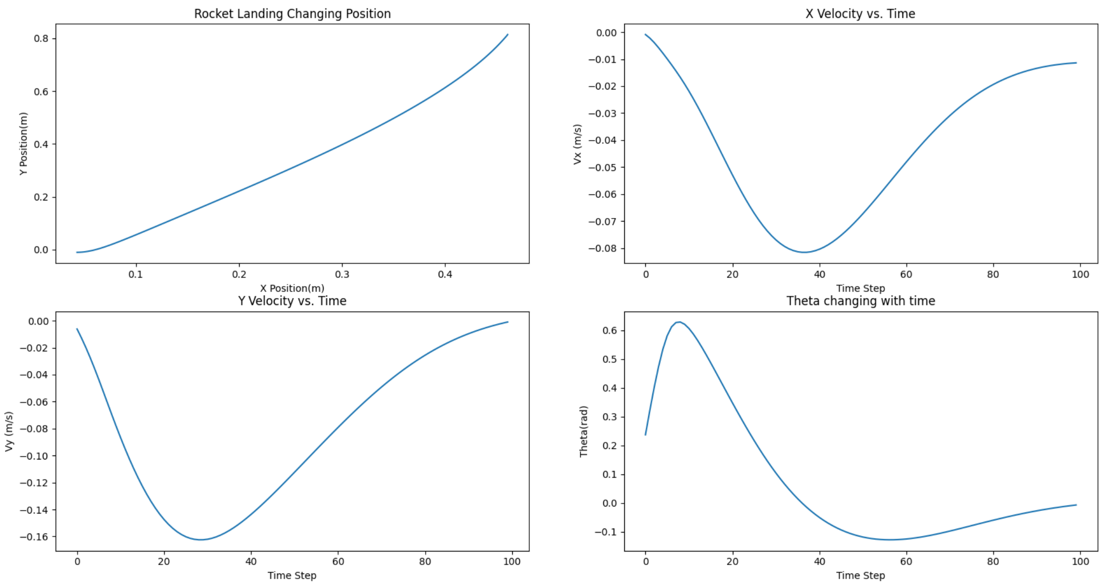
Loss:
[1] loss: 223.275
[2] loss: 117.471
[3] loss: 79.781
[4] loss: 61.175
[5] loss: 50.458

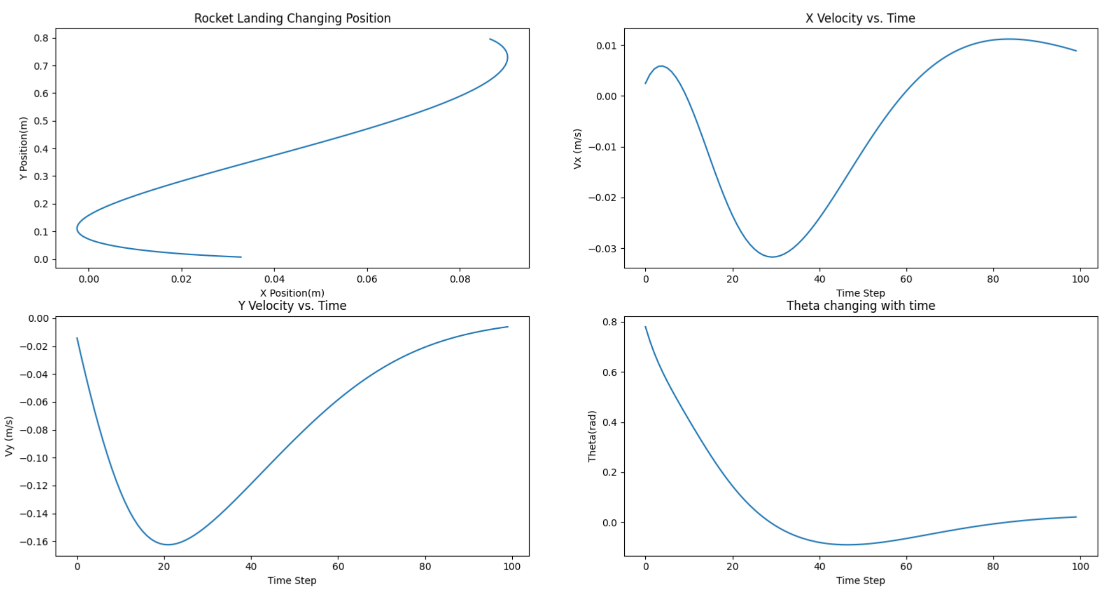
[6] loss: 43.620
[7] loss: 39.016
[8] loss: 35.454
[9] loss: 32.130
[10] loss: 27.208
[11] loss: 20.210
[12] loss: 15.321
[13] loss: 6.180
[14] loss: 4.549
[15] loss: 3.687
[16] loss: 2.972
[17] loss: 2.543
[18] loss: 2.269
[19] loss: 2.050
[20] loss: 1.848
[21] loss: 1.565
[22] loss: 1.049
[23] loss: 0.463
[24] loss: 0.343
[25] loss: 0.289
[26] loss: 0.257
[27] loss: 0.236
[28] loss: 0.220
[29] loss: 0.209
[30] loss: 0.200
[31] loss: 0.193
[32] loss: 0.187
[33] loss: 0.180
[34] loss: 0.173
[35] loss: 0.163
[36] loss: 0.133
[37] loss: 0.114
[38] loss: 0.101
[39] loss: 0.091
[40] loss: 0.084
[41] loss: 0.079
[42] loss: 0.075
[43] loss: 0.071
[44] loss: 0.068
[45] loss: 0.065
[46] loss: 0.062
[47] loss: 0.059
[48] loss: 0.055
[49] loss: 0.043

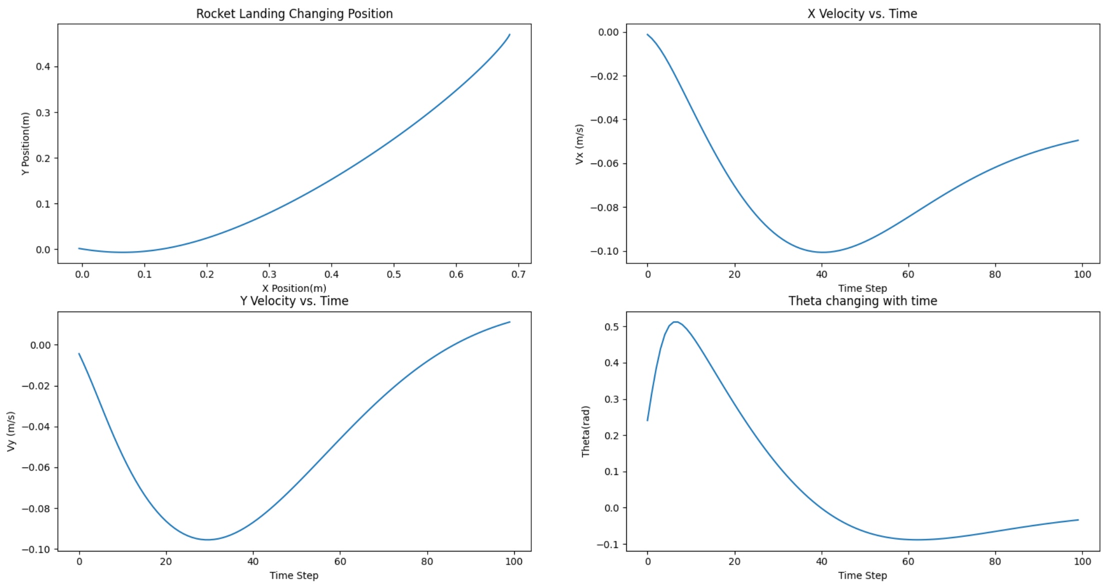
[50] loss: 0.037

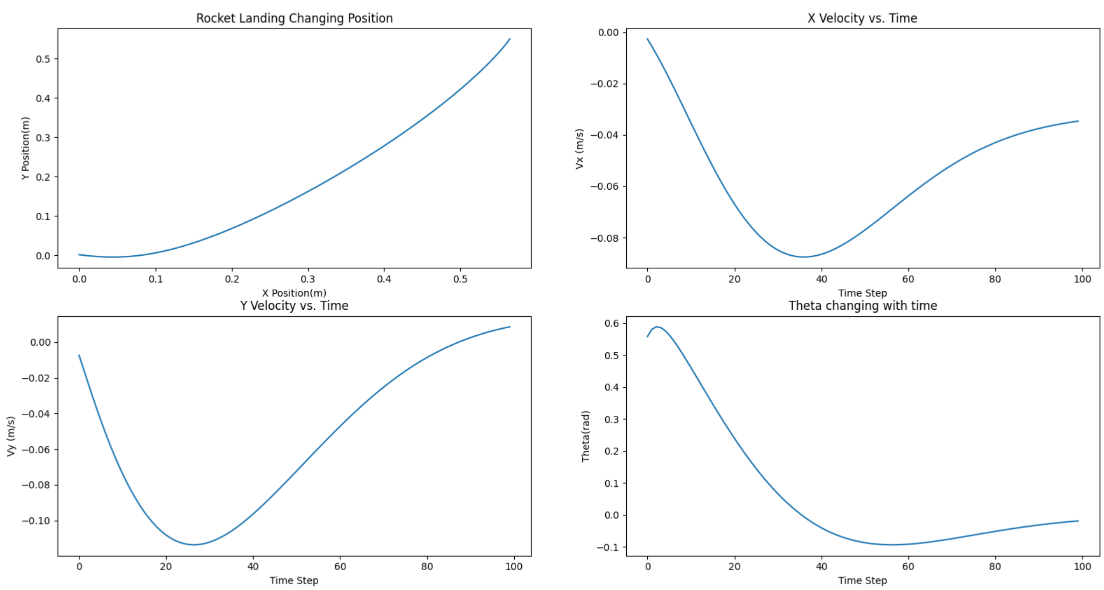

Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step

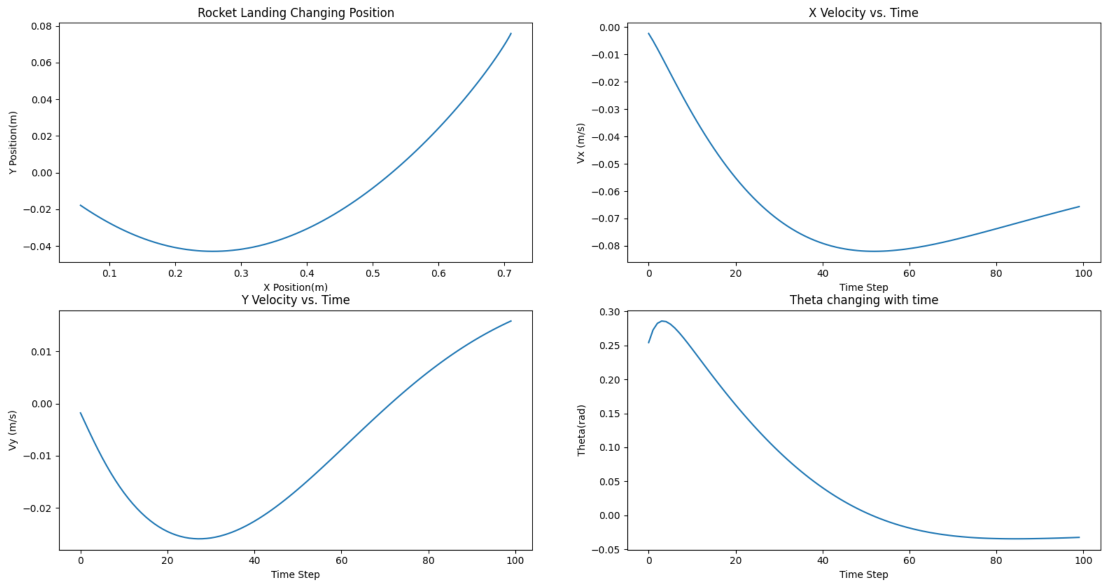

Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step

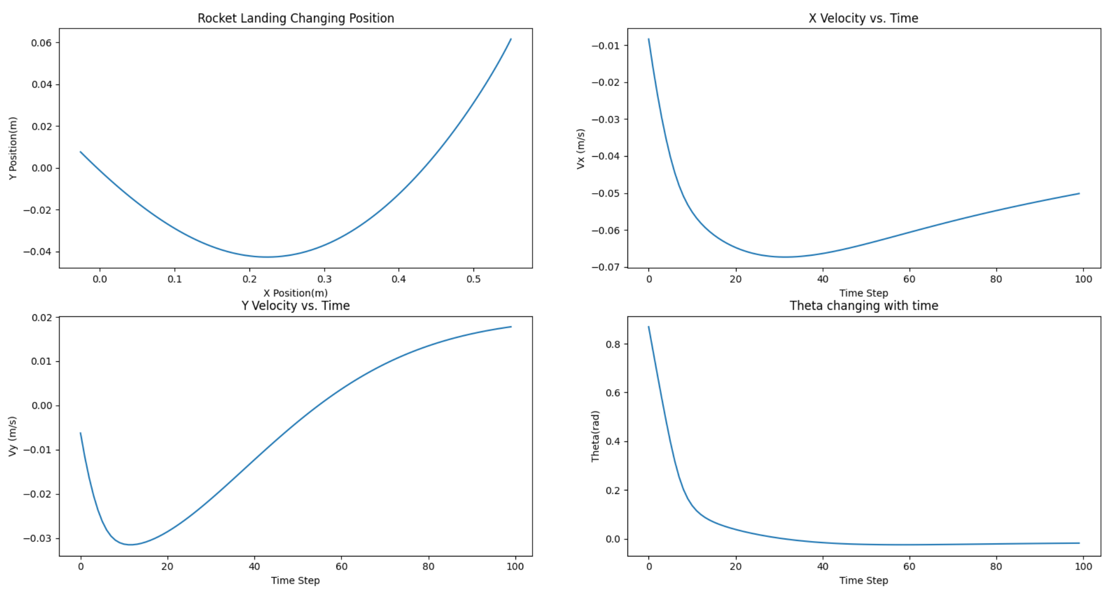

Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step

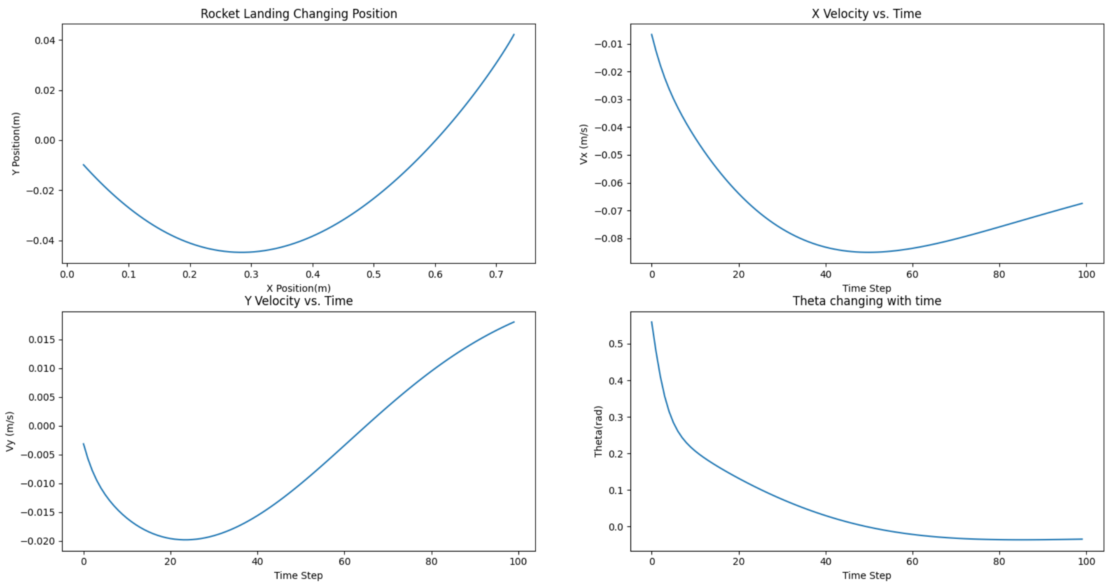

Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step

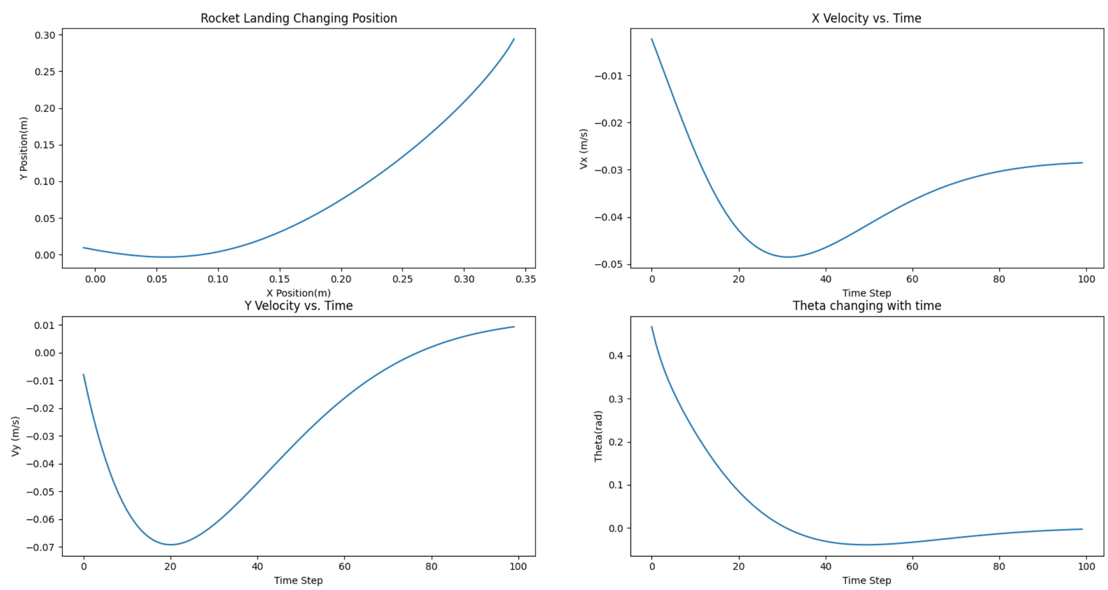

Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step


Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step


Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step


Rocket Landing Changing Position
Y Position(m)
X Position(m)
X Velocity vs. Time
Vx (m/s)
Time Step
Y Velocity vs. Time
Vy (m/s)
Time Step
Theta changing with time
Theta(rad)
Time Step





Conclusions:

The optimization converges roughly 7/10 of the time to the loss approaching 0. Adding more iterations can get it closer to 0 but it will never truly reach a loss of 0 since because of the initial states transitional losses will always be there however small. Vx and Vy converge to 0 most of the time, theta does not however and changes to allow the position and velocities to approach 0.