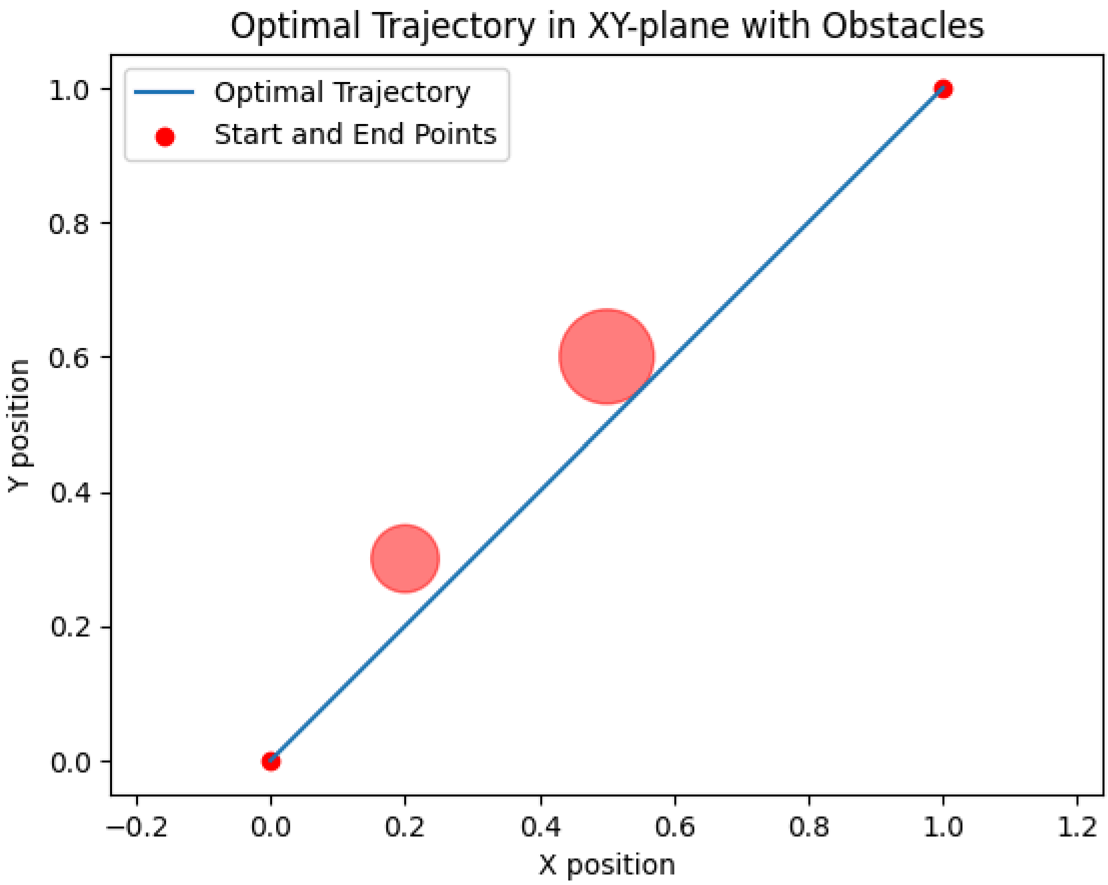
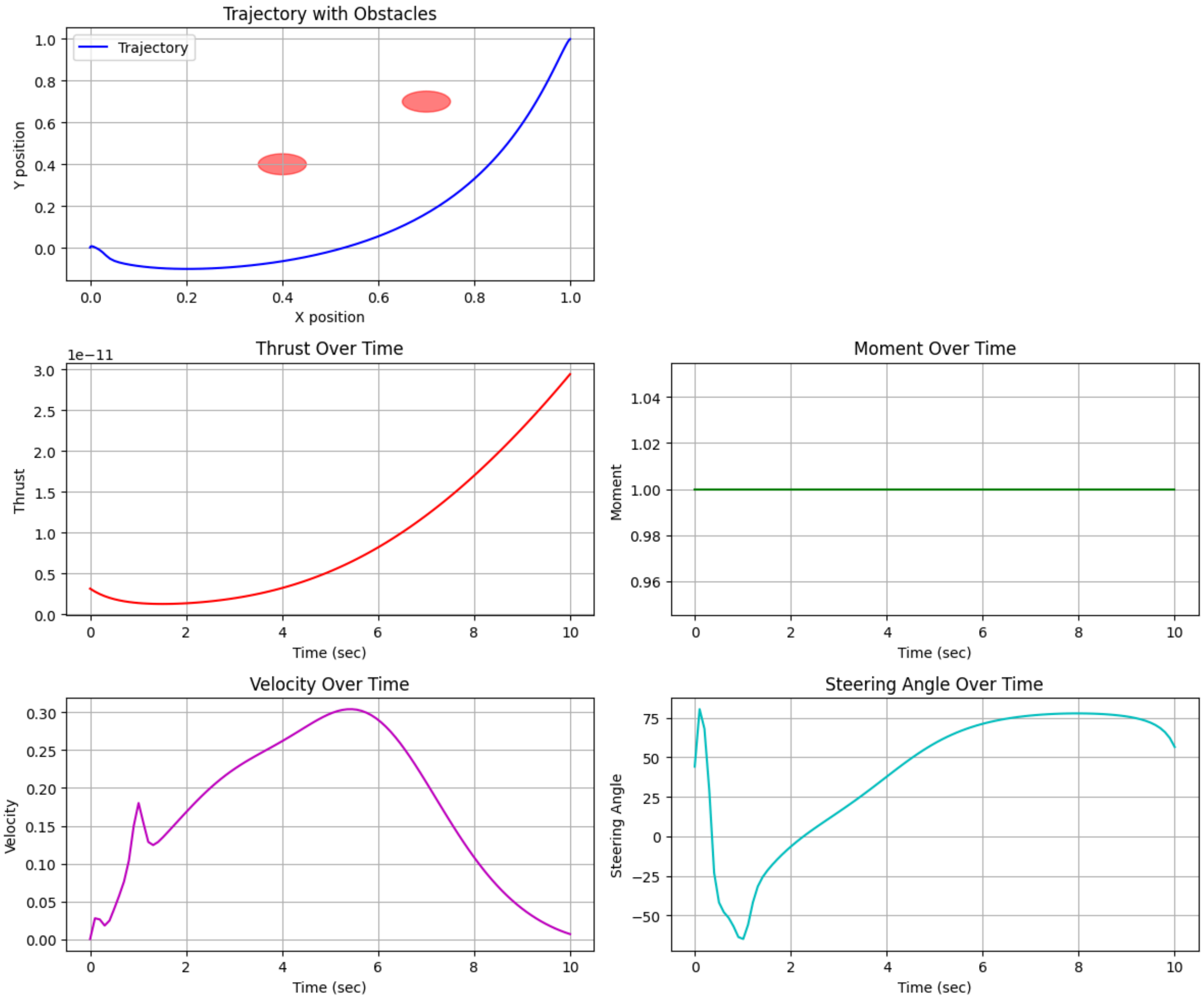
Approach 1

Unfortunately, this approach was unsuccessful at avoiding the obstacles and just barely crashes into them, which is an unacceptable safety issue for an aircraft trajectory (luckily this is a simulation). It uses a 3rd order polynomial for simplicity (other basis functions were going to be used but since the 3rd order polynomial approach failed, I didn't try other basis functions). It does start at (0,0) and stop at (1,1) though. The run time with this is approach is very low, less than a second.



Approach 2

Approach 2 was semi successful, it avoids obstacles and starts at the correct starting position of (0,0) and ends at the correct final position of (1,1), however, it usually takes the long way around the obstacles and seems to be overly cautious, which leads to a less efficient trajectory. This approach is built upon the example code from class. The neural network is slightly more complex to accommodate thrust and moment. The loss function has been modified to include the EOMs and obstacle avoidance function. In the plots, the thrust and moment are within the specified bounds. The network has 1000 iterations of training, so this approach takes a while to run, however the solution quality is better than the basis function approach (since the basis function approach failed).


Trajectory with Obstacles
Trajectory
Y position
X position
Thrust Over Time
1e−11
Thrust
Time (sec)
Moment Over Time
Moment
Time (sec)
Velocity Over Time
Velocity
Time (sec)
Steering Angle Over Time
Steering Angle
Time (sec)