1. INTRODUCTION AND BACKGROUND

Controlling a rocket's orientation and velocity to safely land upright is not a trivial task. As such, in this project, gradient-based algorithms and differentiable programming are utilized to solve this optimal control problem. In particular, a neural network is designed to control the rocket thrusters to ensure the rocket lands upright with zero linear and angular velocity.

A rocket's state at time t, x(t), is described mathematically in Eq. (1) and a possible initial state, x(0), is shown visually in Figure 1. The distance to the ground is captured by the variable d(t), the vertical velocity is v(t), the angle relative to vertical is ϕ(t), and the angular velocity is ω(t). Notice the rocket has thrusters on both sides, indicating that either a positive or negative ω(t) is possible. However, there is only one vertical thruster, whose acceleration is used to balance against the downward acceleration due to gravity.

$$x(t) = \begin{bmatrix} d(t) \\ v(t) \\ \phi(t) \\ \omega(t) \end{bmatrix} \tag{1}$$

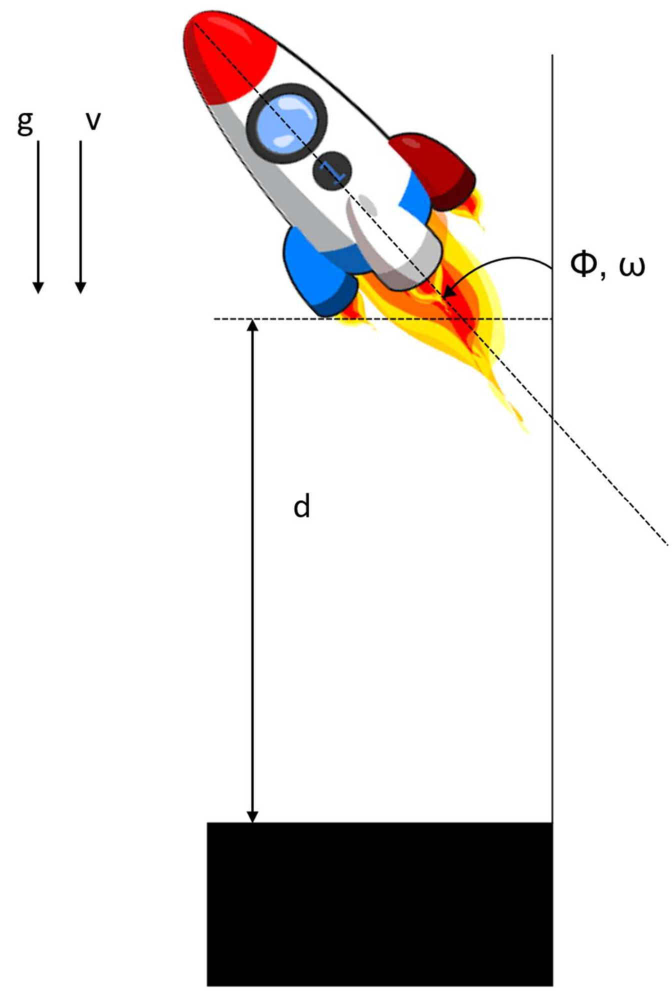

Figure 1: Initial rocket state, x(0)

The desired rocket state, at the end of the analysis period (consisting of T timepoints), x(T), should be zero, as shown in **Figure 2**. As such, a loss function, l(x(T), a(T), α(T)), is defined in Eq. (2) to quantify how close to zero x(T) is at the final timepoint T. The variable a(T) describes the vertical acceleration, considering both the downward acceleration due to gravity and the upward acceleration due to rocket thrust. The variable α(T) denotes the angular acceleration due to the side thrusters (can be positive or negative). Note that a solution with both rotational thrusters on with equal magnitude is possible, which is why it was mentioned that $\omega(t)$ could be either positive or negative, indicating that either the right or left thruster is on, but not both.

$$l\big(x(T), a(T), \alpha(T)\big) = \|x(T)\| = d(T)^2 + v(T)^2 + \phi(T)^2 + \omega(T)^2 \tag{2}$$

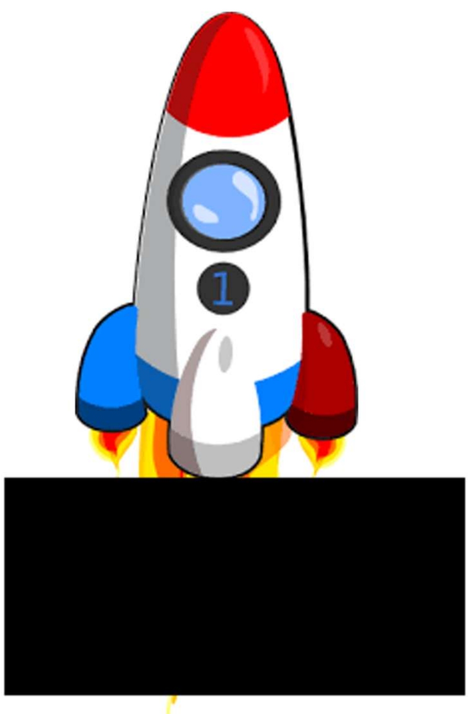

Figure 2: Final rocket state, x(T)

2. METHODOLOGY

The optimal control problem has the objective of minimizing the loss function, as shown in Eq. (3). The components of x(t) are determined by the dynamics in Eq. (4-9). Eq. (4) determines the distance from the ground, based on the velocity (determined in Eq. (5)), which is determined from the acceleration (Eq. (6)), the first action output of the neural network (hence the index 0). Similarly, Eq. (9) determines the angular acceleration (the second output of the neural network, index 1), which will determine the angular velocity in Eq. (8). Then, this angular velocity will determine the angle relative to vertical in Eq. (7). Clearly, this is an unconstrained optimization problem with respect to θ, since all states are determined by the outputs of the neural network (a(t) and α(t)).

$$\min_{\theta} \|x(T)\|^2 \tag{3}$$

$$d(t+1) = d(t) + v(t)\Delta t \;\forall\; t = 1, \dots, T-1 \tag{4}$$

$$v(t+1) = v(t) + a(t)\Delta t \;\forall\; t = 1, \dots, T-1 \tag{5}$$

$$a(t) = f_\theta\big(x(t)\big)[0] \;\forall\; t = 1, \dots, T-1 \tag{6}$$

$$\phi(t+1) = \phi(t) + \omega(t)\Delta t \;\forall\; t = 1, \dots, T-1 \tag{7}$$

$$\omega(t+1) = \omega(t) + \alpha(t)\Delta t \;\forall\; t = 1, \dots, T-1 \tag{8}$$

$$\alpha(t) = f_\theta\big(x(t)\big)[1] \;\forall\; t = 1, \dots, T-1 \tag{9}$$

The assumptions in this problem formulation are as follows:

1. The rocket does not experience any viscous drag forces.
2. The rocket's orientation into the page is vertical.
3. The rocket can overshoot the landing platform (vertically) and still land safely.
4. The rocket's initial vertical position is between [0, 5].
5. The rocket's initial velocity is between [-0.20, 0].

6. The rocket's initial angular orientation is between [-1, 1].
7. The rocket's initial angular velocity is between [-1, 1].

A batch of 100 random initial states between the bounds of initial states are optimized and selected results are presented in the following section.

3. RESULTS AND DISCUSSION

One solution presented an initial state, x0 = [0.1004, -0.1383, 0.2633, 0.8060] and required 38 gradient descent iterations to converge, shown in Figure 3. This figure shows the controller vertically overshoot the landing platform, landing velocity, and angular velocity. However, the solution converges to the angular position (vertical) without overshoot. This is not a preferred solution due to 3 states overshooting the desired final state. Furthermore, the initial state did not set up the rocket for a feasible solution: the rocket was traveling downward with a magnitude of 0.1383 at a position of 0.1004 off the platform and the rocket's vertical thruster acceleration is merely 0.18. As such, the rocket would not have been able to land without some vertical overshoot.

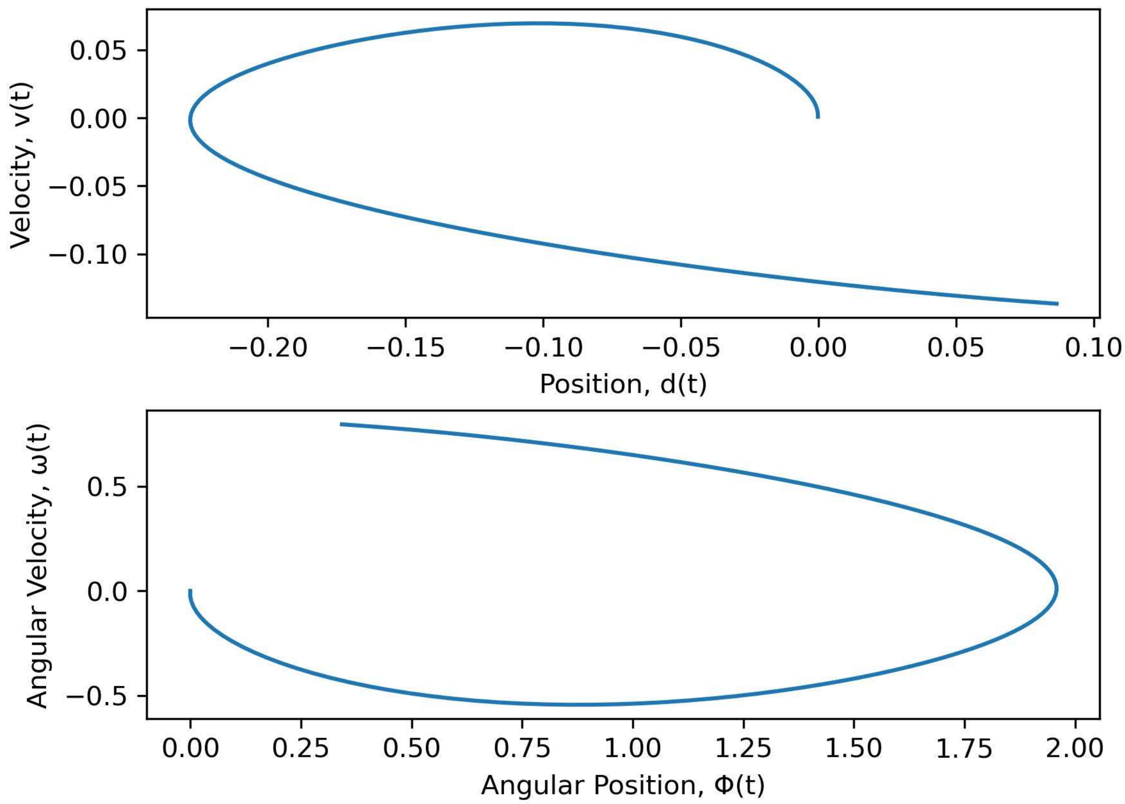

Figure 3: Convergence of initial position (0.1004), initial velocity (-0.1383), initial angular orientation (0.2633), and initial angular velocity (0.8060) to a final position, x = [0, 0, 0, 0] in 38 gradient descent steps.

Another solution, with an initial state further from the desired state (x0 = [1.8448, -0.0296, 0.9505, 0.6765]), converged in 140 gradient descent steps and the results are shown in Figure 4. Contrary to the results shown in Figure 3, this plot only overshoots the angular position and angular velocity, without overshoot of the vertical position or velocity. As such, this solution is preferred to the solution in Figure 3.

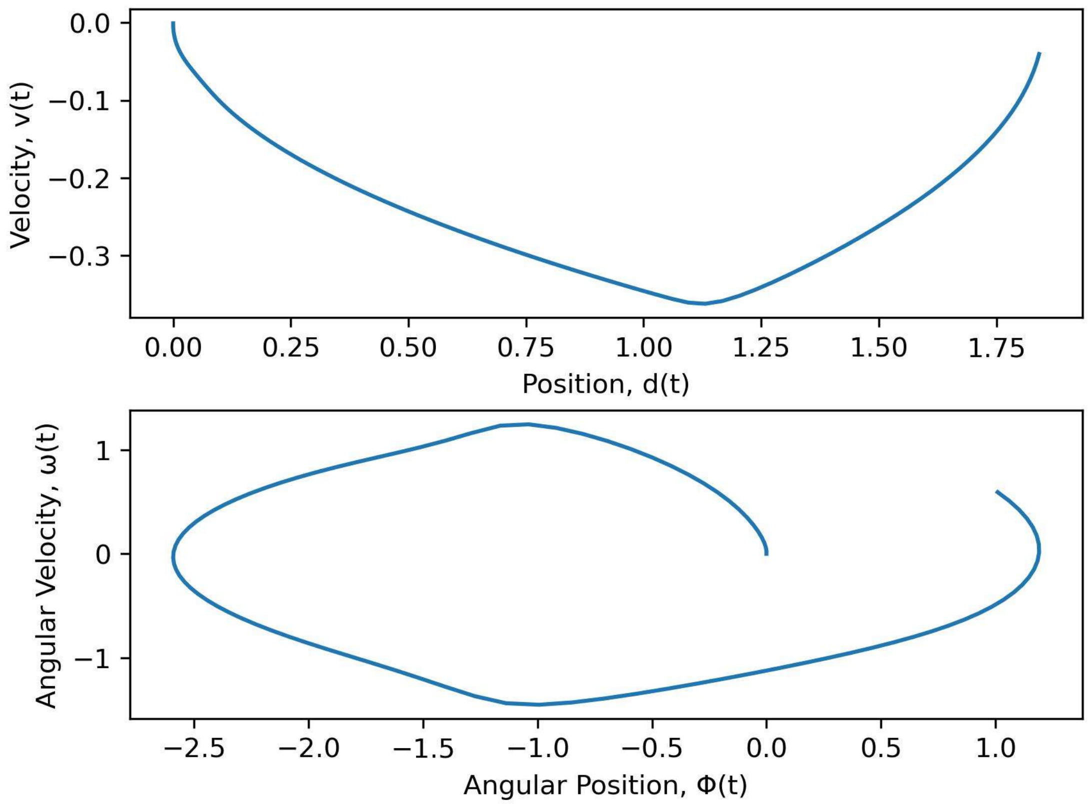

Figure 4: Convergence of initial position (1.8448), initial velocity (-0.0296), initial angular orientation (0.9505), and initial angular velocity (0.6765) to a final position, x = [0, 0, 0, 0] in 140 gradient descent steps.

The next solution, presented in Figure 5, converges in 71 gradient descent steps, and shows only overshoot in the angular velocity with an initial state of x0 = [2.0012, -0.1430, 0.4265, 0.5435].

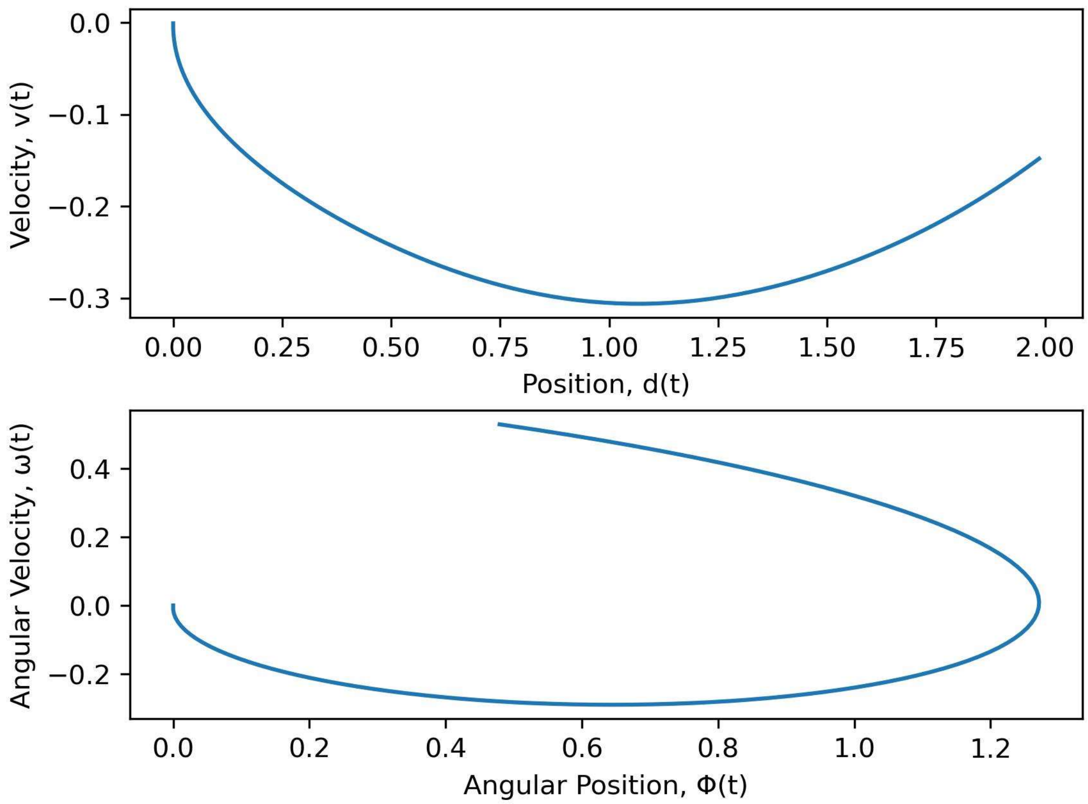

Figure 5: Convergence of initial position (2.0012), initial velocity (-0.1430), initial angular orientation (0.4265), and initial angular velocity (0.5435) to a final position, x = [0, 0, 0, 0] in 71 gradient descent steps.

The last solution presented herein has practically no overshoot in any of the states (minimal overshoot in vertical position). This solution is shown in Figure 6, converged in 66 gradient descent steps, and started at x0 = [0.2167, -0.0744, 0.8065, -0.0793].

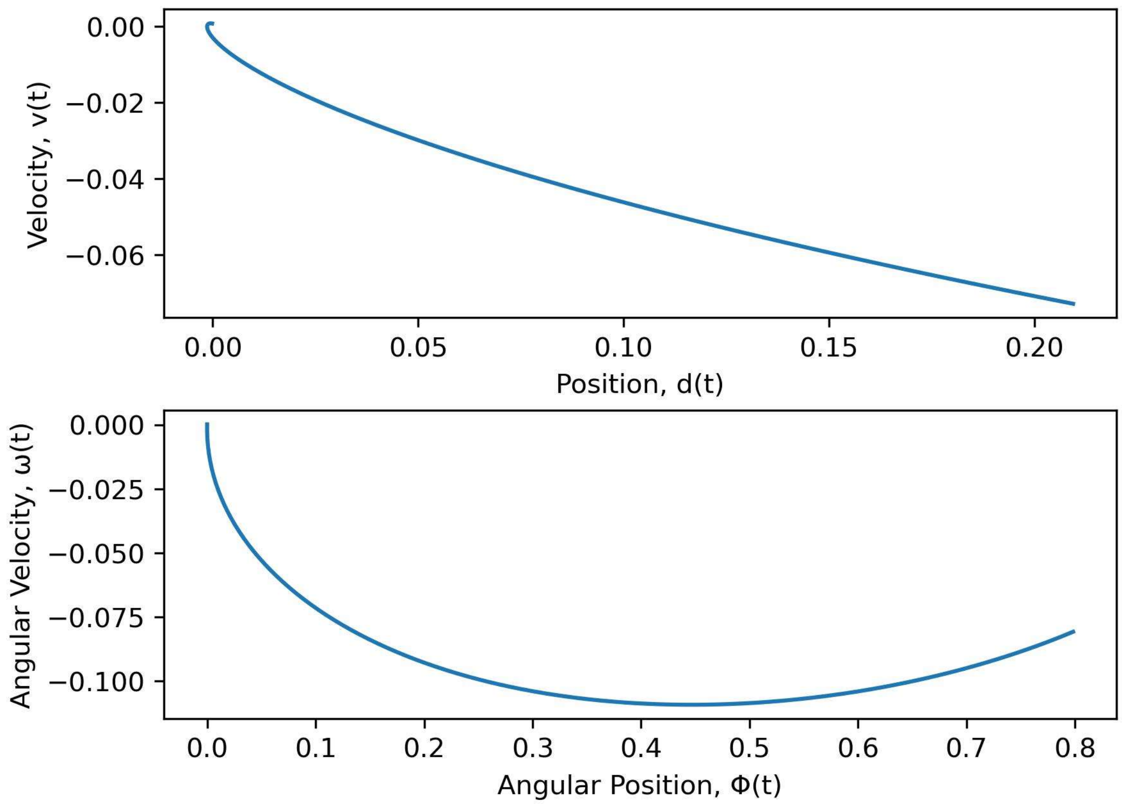

Figure 6: Convergence of initial position (0.2167), initial velocity (-0.0744), initial angular orientation (0.8065), and initial angular velocity (-0.0793) to a final position, x = [0, 0, 0, 0] in 66 gradient descent steps.

NOTE: More results can be found in the "Results_Figures" folder in this report's Github repo.