Dubins Path Missile Interception Scenario

Aaron Hagstrom

May 2024

Summary

Let us assume we want three intercepting missiles to arrive simultaneously along Dubins paths to the circumference of a radius ρ circle centered at a moving target. The purpose of this approach is to hem in the target and thereby increase the probability of destruction.

A Dubins path is the shortest path from a start configuration $(\mathbf{p_s}, \chi_s)$ to an end configuration $(\mathbf{p_e}, \chi_e)$ that comprises, in sequential order, an arc with radius r, a straight line, and a final arc with radius r. To determine the Dubins path for the i'th interceptor, we must compute path length for each of four possible paths to find the one of minimum length.

Specifically, we want to calculate a time-to-go σ for the three missiles arriving simultaneously and equally spaced around a maneuver horizon, where the target is assumed to be moving along a straight line at constant velocity.

If we want one missile to intercept a target, we must first set the predicted position of target equal to the position of one of the three interceptors:

$$\hat{\mathbf{p}}_T(t+\sigma) = \mathbf{p}_I(t+\sigma) \tag{1}$$

where σ is the interception time. if we approximate the target position with a straight line, we can write

$$\hat{\mathbf{p}}_T(t+\sigma) = \mathbf{p}_T(t) + \sigma \mathbf{v}_T(t) \tag{2}$$

We want the interceptor to arrive at a point within a radius ρ of the target at an angle θ from the horizontal, that is, to an endpoint position:

$$\hat{\mathbf{p}}_T(t+\sigma) + \rho R(\theta) \frac{\mathbf{v}_T}{\|\mathbf{v}_T\|}$$

Here $R(\theta)$ is a matrix for rotation around the down axis:

$$R(\theta) = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Furthermore, if we make the interceptor follow a Dubins path Δ, then, we can describe its position on the Dubins path at time σ as

$$\mathbf{p}_I(t+\sigma) = \Delta(\mathbf{p}_I(t), \hat{\mathbf{p}}_T(t+\sigma) + \rho R(\theta)\frac{\mathbf{v}_T}{\|\mathbf{v}_T\|}, R_{min}) \tag{3}$$

The input to the Dubins path is the initial position, the estimated moving target position, and the minimum radius. Now setting (2) and (3) equal, we get

$$\mathbf{p}_T(t) + \sigma\mathbf{v}_T(t) - \Delta(\mathbf{p}_I(t), \hat{\mathbf{p}}_T(t+\sigma) + \rho R(\theta)\frac{\mathbf{v}_T}{\|\mathbf{v}_T\|}, R_{min}) = \mathbf{0} \tag{4}$$

Calling the normed left-hand side of this equation $g(\sigma, t)$, We can solve numerically for σ:

$$g(\sigma, t) = \left\|\mathbf{p}_T(t) + \sigma\mathbf{v}_T(t) - \Delta(\mathbf{p}_I(t), \hat{\mathbf{p}}_T(t+\sigma) + \rho R(\theta)\frac{\mathbf{v}_T}{\|\mathbf{v}_T\|}, R_{min})\right\|$$

The goal is to find a value of σ that minimizes the function $g(\sigma, t)$. Specifically, we are solving for σ^* that minimizes $g(\sigma, t)$:

$$\sigma^* = \arg\min_{\sigma} g(\sigma, t) \tag{5}$$

This can be done with the Nelder-Mead optimization method. If we have three interceptors moving along three Dubins paths, we can calculate a σ^* value for each interceptor. Then we extend the other two interceptor paths to have the same length as the longest path (i.e. the one with the largest σ^*) and continue the simulation. This extension of paths quite a lot of other mathematics to to perform and could describe this scheduling problem if necessary. In a Python-based simulator, we would have modules for physics of the missiles and for guidance to ensure the paths are being followed as best possible.

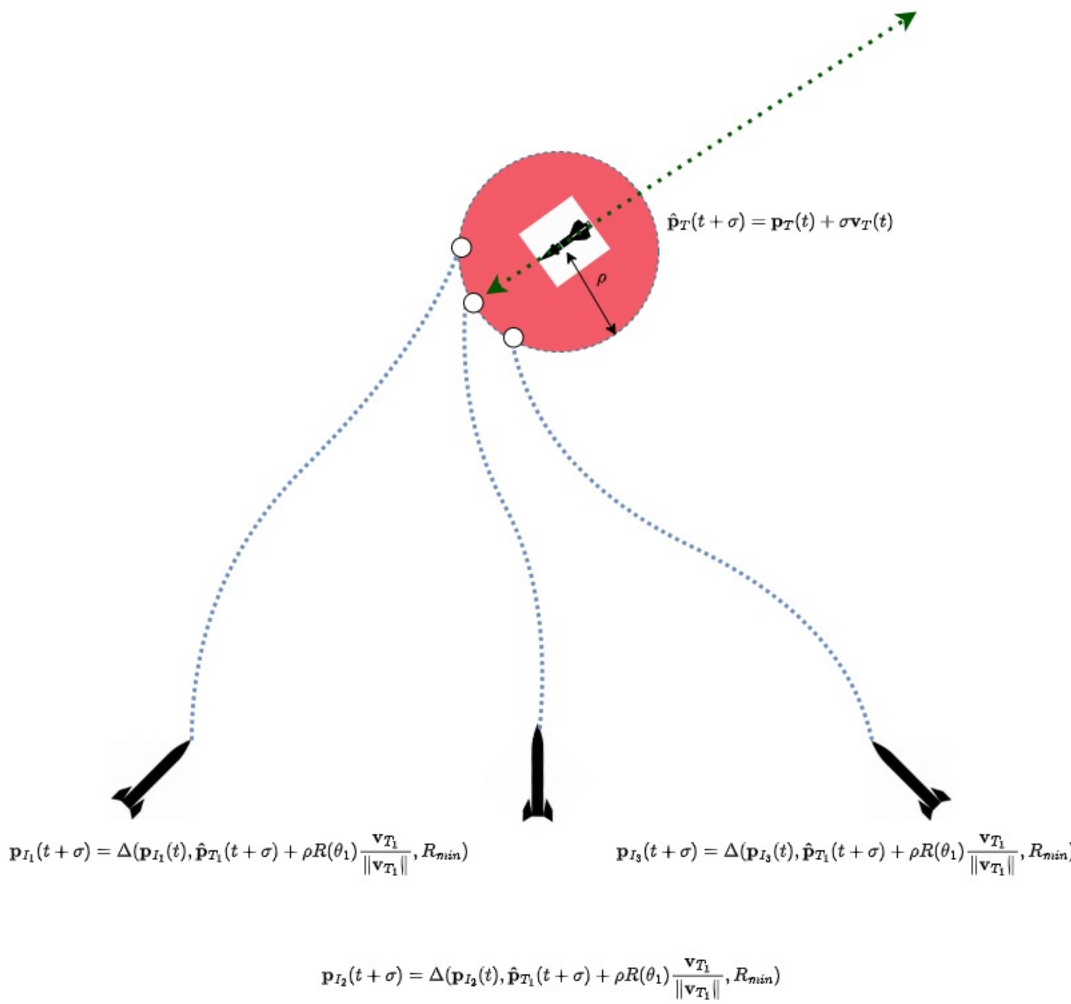

Figure 1: Missile Interception along Dubins Paths on a Maneuver Horizon