ARL-MR-1149 ● FEB 2026

Optimal Trajectory Design for Vehicles Navigating around Obstacles Using Pontryagin Neural Networks

by Bradley T. Burchett

NOTICES

Disclaimers

ARL-MR-1149 ● FEB 2026

Optimal Trajectory Design for Vehicles Navigating around Obstacles Using Pontryagin Neural Networks

Bradley T. Burchett
DEVCOM Army Research Laboratory

REPORT DOCUMENTATION PAGE

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED START DATE	3. DATES COVERED END DATE
February 2026	Memorandum Report	4/9/2025	1/14/2026

4. TITLE AND SUBTITLE
Optimal Trajectory Design for Vehicles Navigating around Obstacles Using Pontryagin Neural Networks

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
Bradley T. Burchett

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
DEVCOM Army Research Laboratory ATTN: FCDD-RLA-WD Aberdeen Proving Ground, MD 21005	ARL-MR-1149

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
A vehicle flying in 3D often encounters obstacles to avoid while navigating to an objective in an optimal way. Optimality implies expending the least control effort, which can be essential for drones with limited energy sources whether chemical or battery. We show how the Pontryagin Neural Network may be applied to find a path that minimizes control effort and intersects the objective at a prescribed time. Additional constraints such as final incidence angle, final glideslope, ground avoidance, and zero thrust may be easily enforced using this technique. We show examples for up to four inequality constraints with both fixed and free final time.

15. SUBJECT TERMS
Weapons Sciences, physics informed neural network, optimization, path planning, drones

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	C. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UU	27

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
Bradley T. Burchett	(520) 691-5110

STANDARD FORM 298 (REV. 5/2020)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

List of Tables

1. Introduction

In recent years, several new techniques have been proposed for optimal path planning and control. Among these are collocation methods, such as Gauss Pseudospectral[1] and theory of functional connections (TFCs).[2,3] Collocation methods transcribe systems of ordinary differential equations into systems of algebraic equations, which are then solved by least-squares. TFC uses least-squares to fit a set of basis functions such that the solution always satisfies boundary conditions. A second "free function" is used to explore the solution space for the optimal among all possible solutions. Most recently, TFC has been enhanced by replacing the free (basis) function with a single-layer neural network. This new formulation is referred to as X-TFC or extreme TFC as the network is an "extreme" learning machine—i.e., one where only the output layer weights are tuned. Combining X-TFC with Pontryagin's maximum principle results in a technique dubbed Pontryagin Neural Network (PoNN), introduced recently by d'Ambrosio et al.[4–6]

In this report, we explore the application of PoNNs to path planning for vehicles avoiding obstacles enroute to the objective. In conjunction with the Fischer–Burmeister function, PoNNs can quickly find the optimal path around obstacles by complying with both equality and inequality constraints. Optimality is defined by achieving the objective with minimal control effort and in some cases in minimum time. Due to X-TFC's guaranteed boundary condition satisfaction, the optimal trajectory always intersects the launch point and objective—i.e., no penalty on the final state is needed to ensure accuracy.

2. Euler–Lagrange Optimal Control with Path Constraints Using PoNNs

In this report, we consider the optimal control problem[7]:

$$\min_{\mathbf{u}} J = \boldsymbol{\varphi}\left(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f)\right) + \int_{\boldsymbol{t_0}}^{\boldsymbol{t_f}} \boldsymbol{L}(\boldsymbol{x}, \boldsymbol{u}, t)\, dt \tag{1}$$

$$\textbf{s.t. } \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \tag{2}$$

$$\boldsymbol{\alpha}(\boldsymbol{x}(t_0), t_0) = \mathbf{0} \tag{3}$$

$$\boldsymbol{\omega}\left(\boldsymbol{x}(t_f), t_f\right) = \mathbf{0} \tag{4}$$

$$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, t) \leq \mathbf{0} \tag{5}$$

The necessary conditions for optimality can be found by defining the Hamiltonian as

$$H = L + \boldsymbol{\lambda}^T \boldsymbol{f} + \boldsymbol{\mu}^T \boldsymbol{g} \tag{6}$$

These conditions are then found from

$$\frac{\partial H}{\partial \boldsymbol{u}} = \boldsymbol{0} \tag{7}$$

and

$$\dot{\boldsymbol{\lambda}}^T = -\frac{\partial H}{\partial \boldsymbol{x}} = -\frac{\partial L}{\partial \boldsymbol{x}} - \boldsymbol{\lambda}^T \frac{\partial f}{\partial \boldsymbol{x}} - \boldsymbol{\mu}^T \frac{\partial g}{\partial \boldsymbol{x}} \tag{8}$$

Additionally, Equation 5 is satisfied by enforcing the complementary slackness conditions

$$\mu_i \geq 0, \mu_i g_i = 0 \; \forall i = 1, \dots, p \tag{9}$$

Equation 9 may be satisfied by employing the Fischer–Burmeister function

$$\phi(a, b) = \sqrt{a^2 + b^2} + a - b \tag{10}$$

which is zero only when conditions (Equation 11) are satisfied

$$\phi(a, b) = 0 \Leftrightarrow \begin{cases} a \leq 0 \\ b \geq 0 \\ ab = 0 \end{cases} \tag{11}$$

Thus, any inequality constraints (i.e., Equation 5) may be enforced by an equality constraint $\phi(a, b) = 0$ where b is the non-negative multiplier and a is the left-hand side of Inequality 5.

3. Proof of Concept: Drone as 3 Degree-of-Freedom (3DOF) Point Mass

3.1 Dynamics

A 3DOF or point mass model may be used to predict a vehicle's trajectory with respect to the ground when only the position of the vehicle's center of gravity (CG) is of interest. Equations 12–17 describe the position and velocity of the vehicle CG with respect to a flat nonrotating earth in a ground-fixed frame[8]:

$$\dot{x} = V_x \tag{12}$$

$$\dot{y} = V_y \tag{13}$$

$$\dot{z} = V_z \tag{14}$$

$$\dot{V}_x = -\hat{C}_x\sqrt{V_x^2 + V_x^2 + V_x^2} \cdot V_x + a_x \tag{15}$$

$$\dot{V}_y = -\hat{C}_x\sqrt{V_x^2 + V_x^2 + V_x^2} \cdot V_y + a_y \tag{16}$$

$$\dot{V}_z = -\hat{C}_x\sqrt{V_x^2 + V_x^2 + V_x^2} \cdot V_z + g + a_z \tag{17}$$

where the $+g$ term indicates that altitude is positive down, $\hat{C}_x$ is a dimensional drag coefficient (1/length), and lift is part of the control effort, $\boldsymbol{a}$. Note that drag is directed opposite the flight path, $\hat{C}_x = \rho S C_X / 2m$, and $S = \pi D^2/4$. Here, the "vehicle" may be a projectile, a drone, or other aircraft. The vector $\boldsymbol{a} = \{a_x \; a_y \; a_z\}$ represents the control effort applied to the vehicle in the ground frame. We do not specify how this effort is generated but look for reasonable values in the solution.

For proof of concept, we use a vehicle with the properties listed in Table 1.

Table 1. Example vehicle properties.

Mass	Characteristic length	Drag	Gravity	Initial velocity
$m = 0.75$ kg	$D = 40$ cm	$C_X = 0.5$	$g = 9.81\ \mathrm{m/s^2}$	$\lVert V \rVert = 25\ \mathrm{m/s}$

3.2 Optimal Control: Euler–Lagrange Necessary Conditions

The optimal control problem is

$$\min_{\mathbf{u}} J = \int_{t_0}^{t_f} \boldsymbol{a}^T \boldsymbol{a}\, dt = \int_{t_0}^{t_f} u_1^2 + u_2^2 + u_3^2\, dt \tag{18}$$

$$\textbf{s.t. } \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \tag{12–17}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0}, \boldsymbol{x}(t_f) = \boldsymbol{x_f} \tag{19}$$

$$\boldsymbol{V}(t_0) = \{V_0 \cos\theta_0 \quad V_{y0} \quad -V_0 \sin\theta_0\} \tag{20}$$

$$C_{th} \leq \mathbf{0} \tag{21}$$

$$C_{hd} \leq \mathbf{0} \tag{22}$$

where the 3DOF dynamics given in Equations 12–17 determine the vehicle motion (point mass with drag). We seek a path planning algorithm that will guide the vehicle around an obstacle modeled by a sphere of radius $\boldsymbol{\rho}$ centered at $\{\boldsymbol{x_{th}}, \boldsymbol{y_{th}}, \boldsymbol{z_{th}}\}$. Thus, constraint (Equation 23) enforces a "no-fly" zone. The no-fly zone may be centered at any altitude that is above, below, or at terrain level. Thus, for some scenarios, the vehicle will want to dive below the obstacle. Equation 24 enforces a "hard deck" altitude at ground level to avoid ground impact. Since altitude, *z*, is defined as positive down, altitude must remain negative to avoid ground impact.

$$C_{\boldsymbol{th}} = \rho - \|r_{th}\|_2 = \rho - \sqrt{(x - x_{th})^2 + (y - y_{th})^2 + (z - z_{th})^2} \leq 0 \tag{23}$$

$$C_{\boldsymbol{hd}} = z_i \leq 0 \tag{24}$$

The Hamiltonian is

$$H = u_1^2 + u_2^2 + u_3^2 + \boldsymbol{\lambda}^T \boldsymbol{f} + \mu_1 C_{th} + \mu_2 C_{hd} \tag{25}$$

Applying Euler–Lagrange, we first evaluate $\frac{\partial H}{\partial \boldsymbol{u}} = 0$, which renders

$$u_1 = -\lambda_4/2\,, u_2 = -\lambda_5/2\,, u_3 = -\lambda_6/2$$

These are substituted into Equations 12–17 and the second necessary condition, $\dot{\boldsymbol{\lambda}}^T = -\frac{\partial H}{\partial \boldsymbol{x}}$, is applied resulting in

$$\dot{\lambda}_i = -\mu_{th} \frac{\partial C_{th}}{\partial x_i} - \mu_{hd} \frac{\partial C_{hd}}{\partial x_i}, i = 1,2,3, x_i \in \{x, y, z\} \tag{26}$$

$$\dot{\lambda}_4 = -\lambda_1 - \left(-\hat{C}_x V - \frac{\hat{C}_x V_x^2}{V}\right) \lambda_4 \tag{27}$$

$$\dot{\lambda}_5 = -\lambda_2 - \left(-\hat{C}_x V - \frac{\hat{C}_x V_y^2}{V}\right) \lambda_5 \tag{28}$$

$$\dot{\lambda}_6 = -\lambda_3 - \left(-\hat{C}_x V - \frac{\hat{C}_x V_z^2}{V}\right) \lambda_6 \tag{29}$$

where $\frac{\partial C_{th}}{\partial x_i} = -x_i / \sqrt{(x - x_{th})^2 + (y - y_{th})^2 + (z - z_{th})^2}$ and $\frac{\partial C_{hd}}{\partial z}$=1, all others are zero.

And, resulting in the loss functions

$$\mathcal{L}_1 = \boldsymbol{w_x} \left(\dot{\boldsymbol{V}} + \hat{C}_x \mathbf{N} \cdot \boldsymbol{V}_i - [0 \quad 0 \quad g]^T + \frac{1}{2} \boldsymbol{I} \cdot \boldsymbol{\lambda_v} \right) \tag{30}$$

$$\mathcal{L}_2 = \boldsymbol{w_{\lambda 1}} \left(\dot{\boldsymbol{\lambda}}_{\boldsymbol{x}} + \mu_{th} \frac{\partial C_{th}}{\partial \boldsymbol{x}} + \mu_{hd} \frac{\partial C_{hd}}{\partial \boldsymbol{x}} \right), \boldsymbol{x} \in \{x, y, z\} \tag{31}$$

$$\mathcal{L}_3 = \boldsymbol{w_{\lambda 2}} \left(\boldsymbol{\lambda_v} + \boldsymbol{\lambda_x} - \hat{C}_x (\mathbf{N} + \mathbf{N^{-1}} \boldsymbol{\Xi}) \boldsymbol{\lambda_v} \right), \boldsymbol{v} \in \{V_x, V_y, V_z\} \tag{32}$$

where

$$\mathbf{N} = \begin{bmatrix} V & & \\ & V & \\ & & V \end{bmatrix}, \boldsymbol{\Xi} = \begin{bmatrix} \boldsymbol{V_x^2} & & \\ & \boldsymbol{V_y^2} & \\ & & \boldsymbol{V_z^2} \end{bmatrix}$$

$$\mathcal{L}_4 = \boldsymbol{w_{th}} \left(\phi(C_{th}, \mu_{th}) \right) \tag{33}$$

$$\mathcal{L}_5 = \boldsymbol{w_{hd}}\big(\phi(C_{hd}, \mu_{hd})\big) \tag{34}$$

Where $[\boldsymbol{w_x}, \boldsymbol{w_{\lambda 1}}, \boldsymbol{w_{\lambda 2}}, \boldsymbol{w_{th}}, \boldsymbol{w_{hd}}]$ are weights that determine the relative importance of each constraint, and the total loss is found by concatenating Equations 30–34 as

$$\mathcal{L} = [\mathcal{L}_1^T, \mathcal{L}_2^T, \mathcal{L}_3^T, \mathcal{L}_4^T, \mathcal{L}_5^T] \tag{35}$$

In the PoNN scheme, states and co-states are solved through a collocation. However, to ensure that boundary conditions are satisfied, the solutions are contrived in the form of constrained expressions (CEs). These CEs consist of a free function and a linear combination of basis functions that automatically satisfy the boundary conditions, i.e.,

$$y_j^l(t) = g_j^l(t) + \sum_{\boldsymbol{k=1}}^{\boldsymbol{n_j}} \eta_{k,j} s_k^l(t) \tag{36}$$

where the l superscript denotes the lth derivative with respect to time, n_j is the number of constraints for state j, $g_j^l(t)$ is the free function, and $s_k^l(t)$ are basis functions with properties that keep the problem well-conditioned (orthogonal polynomials, neural net activation functions, etc.). For a PoNN, the free function is the output of a single-layer neural network such that

$$g_j(t) = \sum_{q=1}^{L} \beta_{j,q} \sigma(w_q^T z + b_q)$$

where σ is the activation function, L is the number of hidden units, w_q^T are the input weights, and b_q are the input biases. Input weights and biases are chosen to span the input space and are held constant. Only the output weights $\beta_{j,q}$ are trained. Since the free function is linear in $\beta_{j,q}$, this training requires little computing time and $g_j(t)$ may be written compactly as $g_j(t) = \boldsymbol{\sigma}^T \boldsymbol{\beta}_j$. Note that as a collocation method, the independent variable z is scaled such that $-1 \le z \le 1$. Thus, for time-based problems, the temporal horizon is mapped to the domain of z by

$$z = -1 + \frac{2}{t_f - t_0}(t - t_0)$$

Thus, the scaling constant $b^2 = \frac{2}{t_f - t_0}$ must be applied to time derivatives to map Equation 36 from time to z domain:

$$g_j^l(t) = \frac{d^l g_j(z)}{dt^l} = \frac{d^l g_j(z)}{dz^l}\left(\frac{dz}{dt}\right)^l = \frac{d^l g_j(z)}{dz^l}(b^2)^l = (b^2)^l \left(\frac{d^l \boldsymbol{\sigma}}{dz^l}\right)^T \boldsymbol{\beta}_j$$

In this report, the second term in Equation 36 is expanded into products of switching functions and the basis functions at a given boundary condition. These switching functions equal 1 at the time where the respective boundary condition is in effect and 0 elsewhere and are denoted as $\boldsymbol{\Omega}_i$ hereafter. See D'Ambrosio and Furfaro[4] for a full catalog of switching functions. For the application highlighted in this report, the initial and final positions and the initial velocity are specified. The final velocity is free. The $\boldsymbol{\Omega}_i$ from Table A5 of D'Ambrosio and Furfaro[4] are used in Equations 37–39, and we also present them in the Appendix, Table A-1.

Now, we can build the solution for the states and co-states in terms of constrained expressions. These CEs will be substituted into Equations 30–34 to compute the losses, and the optimal solution will be found by minimizing the losses through adjusting the free variables $(\boldsymbol{\beta}_j, \mu_{th}, \mu_{hd})$.

$$\boldsymbol{X}_i^T(z) = \boldsymbol{\beta}_{\mathbf{1}:\mathbf{3}}\left(\boldsymbol{\sigma} - \boldsymbol{\sigma}_0\boldsymbol{\Omega}_\mathbf{1} - \boldsymbol{\sigma}_f\boldsymbol{\Omega}_\mathbf{2} - \boldsymbol{\sigma}'\boldsymbol{\Omega}_\mathbf{3}\right) + \boldsymbol{X}_\mathbf{0}\boldsymbol{\Omega}_\mathbf{1} + \boldsymbol{X}_f\boldsymbol{\Omega}_\mathbf{2} + (\boldsymbol{V}_\mathbf{0}\boldsymbol{\Omega}_\mathbf{3})/\boldsymbol{b}^\mathbf{2} \tag{37}$$

$$\begin{aligned}\boldsymbol{V}_i^T(z) = \boldsymbol{b}^\mathbf{2}\big(&\boldsymbol{\beta}_{\mathbf{1}:\mathbf{3}}\left(\boldsymbol{\sigma}' - \boldsymbol{\sigma}_0\boldsymbol{\Omega}'_\mathbf{1} - \boldsymbol{\sigma}_f\boldsymbol{\Omega}'_\mathbf{2} - \boldsymbol{\sigma}'\boldsymbol{\Omega}'_\mathbf{3}\right) + \boldsymbol{X}_\mathbf{0}\boldsymbol{\Omega}'_\mathbf{1} + \boldsymbol{X}_f\boldsymbol{\Omega}'_\mathbf{2} \\ &+ (\boldsymbol{V}_\mathbf{0}\boldsymbol{\Omega}'_\mathbf{3})/\boldsymbol{b}^\mathbf{2}\big)\end{aligned} \tag{38}$$

$$\begin{aligned}\dot{\boldsymbol{V}}_i^T = \boldsymbol{b}^\mathbf{2}\big(&\boldsymbol{\beta}_{\mathbf{1}:\mathbf{3}}\left(\boldsymbol{\sigma}'' - \boldsymbol{\sigma}_0\boldsymbol{\Omega}''_\mathbf{1} - \boldsymbol{\sigma}_f\boldsymbol{\Omega}''_\mathbf{2} - \boldsymbol{\sigma}'\boldsymbol{\Omega}''_\mathbf{3}\right) + \boldsymbol{X}_\mathbf{0}\boldsymbol{\Omega}''_\mathbf{1} + \boldsymbol{X}_f\boldsymbol{\Omega}''_\mathbf{2} \\ &+ (\boldsymbol{V}_\mathbf{0}\boldsymbol{\Omega}''_\mathbf{3})/\boldsymbol{b}^\mathbf{2}\big)\end{aligned} \tag{39}$$

$$\boldsymbol{\lambda}_{xi} = \boldsymbol{\beta}_{\mathbf{4}:\mathbf{6}}\boldsymbol{\sigma} \tag{40}$$

$$\dot{\boldsymbol{\lambda}}_{xi} = b^2\boldsymbol{\beta}_{\mathbf{4}:\mathbf{6}} \tag{41}$$

$$\boldsymbol{\lambda}_{vi} = \boldsymbol{\beta}_{7:9}\left(\boldsymbol{\sigma} - \boldsymbol{\sigma}_f\right) \tag{42}$$

$$\dot{\boldsymbol{\lambda}}_{vi} = b^2\boldsymbol{\beta}_{7:9}\boldsymbol{\sigma}' \tag{43}$$

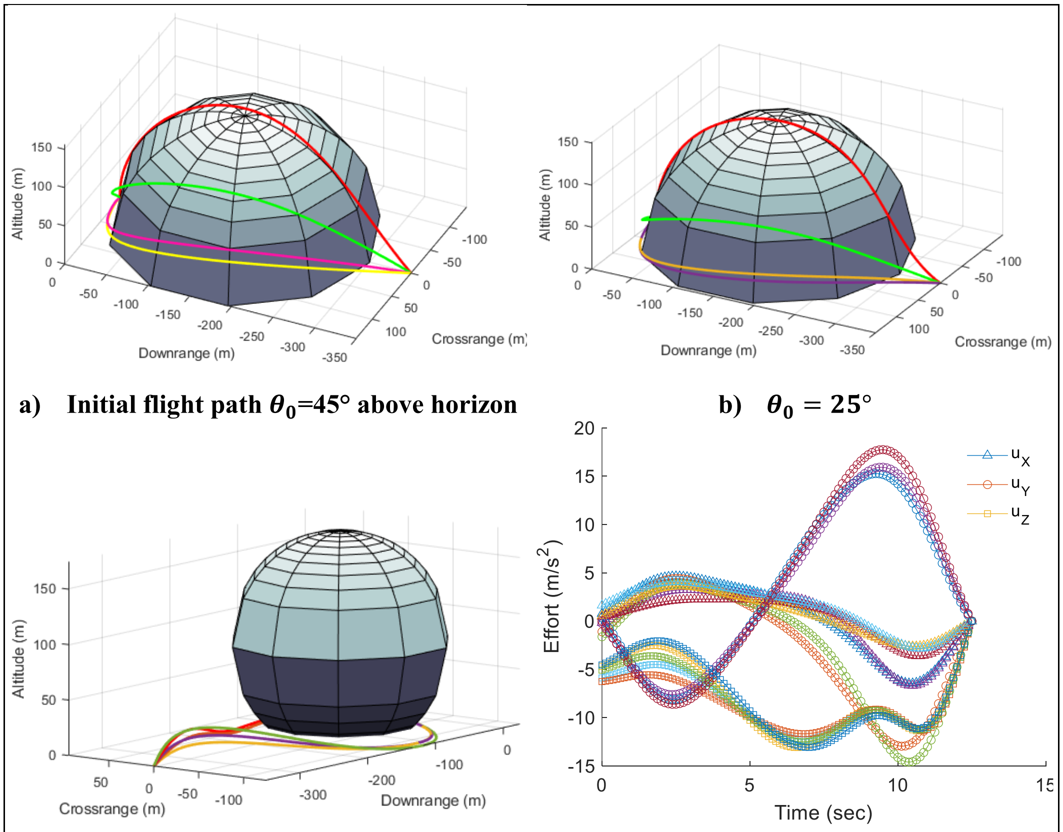
where $\boldsymbol{\sigma}$ is the vector of neural net activation functions evaluated over the z domain, and $\boldsymbol{\sigma}_0$ and $\boldsymbol{\sigma}_f$ are the activation functions evaluated at z=-1 and z=1, respectively. The second factor on the right-hand side of Equation 42 is modified to constrain $\boldsymbol{\lambda}_v\left(t_f\right) = 0$, since the velocity is unconstrained at the objective. Equations 38–43 are substituted into Equations 30–34 to evaluate the losses using the PoNN collocation. The optimal solution is then found by a Levenberg–Marquardt search for $(\boldsymbol{\beta}_j, \mu_{th}, \mu_{hd})$.

3.3 Results: Guiding Around Obstacle to Objective

In this report, we chose the activation function $\boldsymbol{\sigma}$ to be hyperbolic tangent (tansig). The input weights and biases were chosen as $w_q = 5.2$, $b_q = w_q\left(-1 + \frac{2(q-1)}{L-1}\right), q = 1, \ldots, L$. Output weights are initially set to uniform random $0 \leq \beta_{ij} \leq 1$, and multipliers μ_{th}, μ_{hd} are all initially set to unity.

First, we illustrate a basic scenario where the vehicle must navigate around a ground-level obstacle of radius 150 m and centered 150 m up range of the objective on the trajectory centerline. The vehicle is initially 350 m up range of the objective with a speed of 25 m/s plus a small cross-range velocity. The vehicle must reach the objective at t_0 = 7.5 s, while avoiding the obstacle and the ground. In Figures 1a and 1b, the vehicle starts with an initial speed determined by $V_x = 25\cos\theta_0, V_z = 25\sin\theta_0$, and V_y chosen randomly. The obstacle is located at ground level. In Figure 1a, the red trajectory has zero initial cross-range velocity and an initial flight path angle 45° above the horizon. The optimization recognized that a small vertical acceleration is required to surmount the obstacle and thus maintains centerline while skimming the top of the hemisphere. As random initial cross-range velocity is introduced, the algorithm allows the vehicle to depart the line-of-sight direct-to-objective plane and take advantage of the lower maximum altitude required to crest the sphere. The green, magenta, and yellow trajectories illustrate the solution with monotonically increasing initial cross-range velocity.

Figure 1b shows the same scenario with a decreased initial flight path angle and cross-range velocities chosen at random. Again, the red trajectory has zero initial cross-range velocity. The green, yellow, and purple trajectories show results for monotonically increasing initial cross-range velocity. In every case, optimization takes advantage of the initial velocity vector and applies just enough vertical control effort to avoid the hemisphere.



a) **Initial flight path θ_0=45° above horizon**

b) $\boldsymbol{\theta_0 = 25°}$

c) **Obstacle centered above ground, Monte-Carlo QE, hard deck enforced**

d) **Control effort from Monte-Carlo cases in pane c) in ground frame**

Figure 1. Scenario 1: Drone must remain outside obstacle sphere (depicted as solid surface). Panes a, b) obstacle centered at ground level, initial horizontal and vertical speeds held constant, initial cross-range speed chosen randomly. Pane c) obstacle centered above ground, all initial velocity components set randomly.

A second scenario was contrived to see whether the vehicle would dive beneath an obstacle centered above the ground. In Figure 1c, a 100-m radius obstacle is centered 75 m up range of the objective (i.e., 75 m above the ground on the line-of-sight plane from vehicle to objective). Thus, the objective lies below the shadow of the obstacle, and navigation over the top would require the vehicle to backtrack to reach the objective. Instead, the algorithm finds a shorter path, skirting the obstacle below its equator. Results of this scenario, this time allowing the initial flight path angle to vary, are shown in Figure 1c. The hard deck constraint is actively enforced to prevent the vehicle from striking the ground in all cases. In Figure 1d, we have plotted the control effort in the ground frame for the cases shown in Figure 1c. The vertical acceleration is limited to 1.3 g. Cases with higher initial flight path angle use less vertical lift, and instead trace a wider path around the obstacle, requiring about 1.7 horizontal g's to make the final turn back to the objective. These appear to be reasonable values for a maneuverable drone.

3.4 Results: No-Thrust

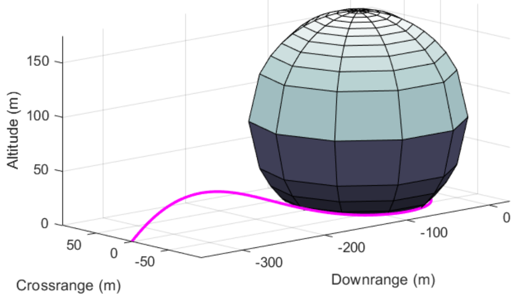
The Fischer–Burmeister approach for inequality constraints provides an opportunity to restrict control effort to transverse forces and drag (no-thrust) without having to write this constraint into the system dynamics. Assuming angle-of-attack and sideslip are small, drag acts opposite the flight path direction $\vec{\boldsymbol{V}}$. Admissible control forces for a gliding vehicle would then act normal to the flight path, or else opposite the flight path. That is, the projection of the control force on the flight path must either be zero, or parallel to drag. We can write this in terms of the velocity Lagrange multiplier as

$$\mathbf{cos}\ \boldsymbol{\vartheta} = \frac{-\vec{\boldsymbol{\lambda}}_v \cdot \vec{\boldsymbol{V}}}{\|\vec{\boldsymbol{\lambda}}_v\|\|\vec{\boldsymbol{V}}\|} \leq \mathbf{0} \tag{44}$$

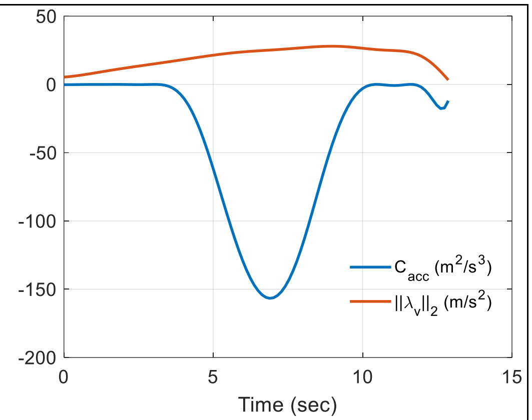
where $\boldsymbol{\vartheta}$ is the angle between the control effort and flight path. However, for $\mathbf{cos}\ \boldsymbol{\vartheta}$ to be negative, we need only ensure that the numerator is negative. Thus, we can prohibit thrust control by the constraint $C_{acc} = -\vec{\boldsymbol{\lambda}}_v \cdot \vec{\boldsymbol{V}} \leq 0$. Employing Fischer–Burmeister, we append Equation 45 to the previous set (Equations 30–34)

$$\mathcal{L}_6 = \boldsymbol{w_{acc}}\big(\phi(C_{acc}, \mu_{hd})\big) \tag{45}$$

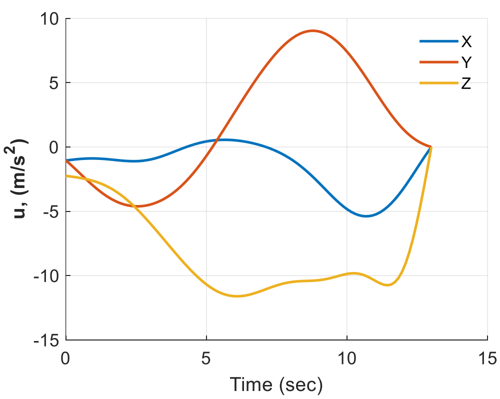
Figure 2 shows the result for a vehicle encountering an obstacle that is centered above the ground and navigating with zero thrust and a fixed final time. The initial speed was set to 40 m/s with the vehicle launched 25° above the horizon in the line-of-sight plane to the objective. In Figure 2, panes a–d, the flight time is limited to 13 s, giving the vehicle adequate, but not excessive time to reach the objective. Figure 2a depicts the optimal trajectory in relation to the obstacle. The vehicle arrests its initial upward climb and hugs the ground and obstacle to find the shortest path to reach the objective in time. Since lift is a transverse force, it is not penalized by the no-drag constraint, and the dynamics used in this demonstration do not penalize the use of lift by inducing additional drag. Figure 1b shows the numerator of the cosine of the angle between commanded acceleration and flight path (C_{acc}). This quantity must be non-positive to satisfy the no-thrust constraint and it is for the entire trajectory. The magnitude of the control vector is juxtaposed in Figure 2b with C_{acc}.



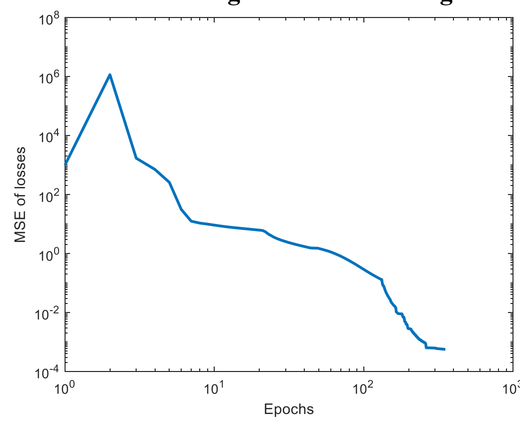
a) Zero thrust, $t_f = 13$ s, vehicle finds shortest flyable path while avoiding the ground



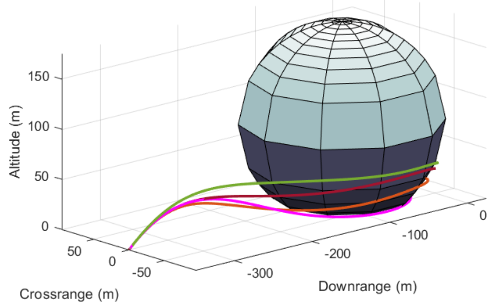
b) Checking no-thrust constraint, cosine of angle from control effort to flight path remains negative for entire flight



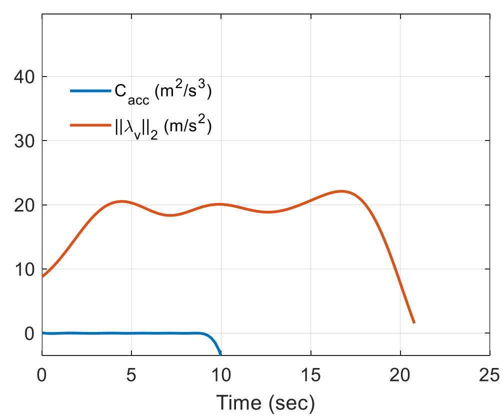
c) Control commands for 13-s flight. Downrange has minimal effort, mainly drag, vertical ≥ 1 g while avoiding ground



d) Convergence plot, residual vector norm converged in 330 iterations



e) As time horizon lengthens, control becomes less aggressive, path lengthens, trajectory closer to ballistic $t_f \in \{12, 15, 18, 21\}$

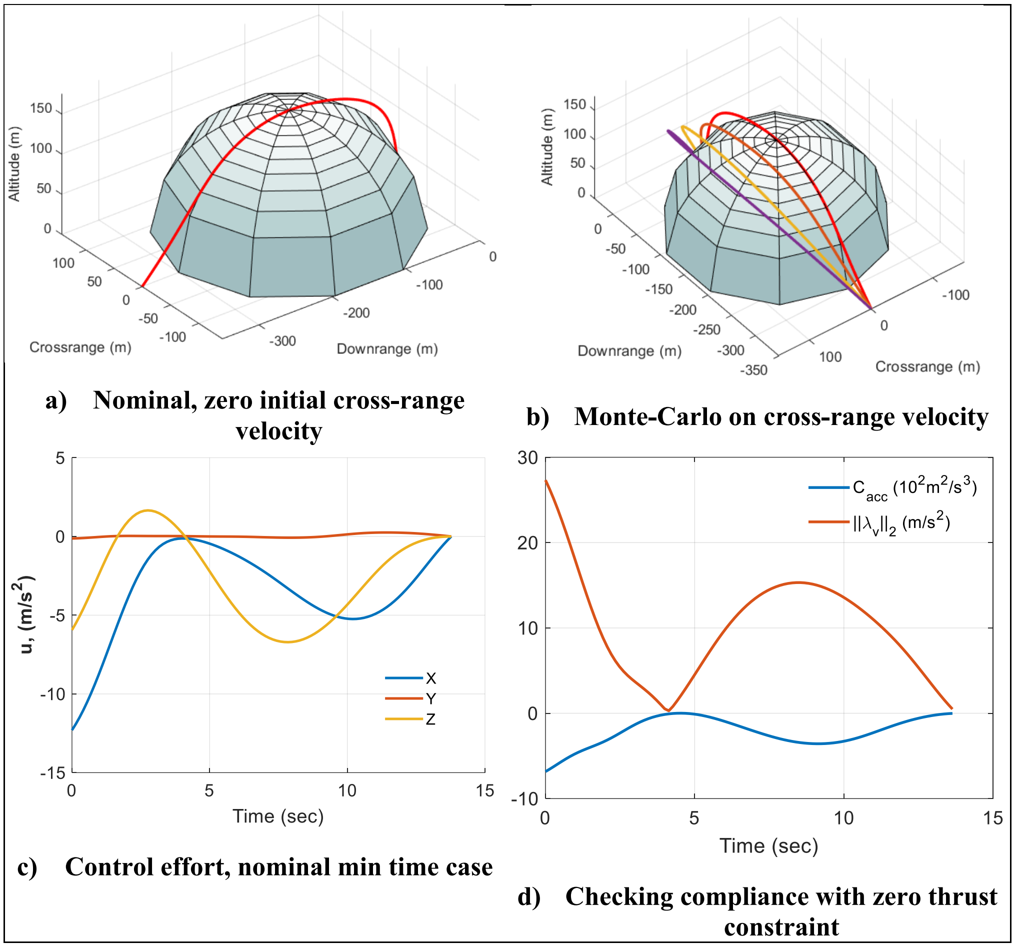


f) Magnitude of control effort with 21 s time horizon

Figure 2. Scenario 3: Zero thrust (projection of control effort on flight path must be opposite direction of flight or zero). Fixed final time. Panes a–d use a final time of 13 s. Pane e, trade study on final time. Pane f, $t_f = 21$ s. $\vec{V}_0 = 40[c_{40°}, 0, -s_{40°}]^T$, $w_v = 0.3$, $w_{\lambda r} = 0.1$, $w_{\lambda v} = 10$; $w_{th} = 9$, $w_a = 48$, $w_{acc} = 0.17$.

Figure 2c shows the control commands for the 13-s flight in the ground frame. Note that the *z* component is positive in the nadir, thus the yellow line shows only lifting forces. The initial descent is achieved by allowing gravity to halt the initial climb and pull the vehicle down to the deck. Approaching the ground, the vehicle applies enough lift to avoid ground impact. Force in the down-range direction is mostly negative, thus opposing the direction of flight. Cross-range control approaches 1 g as the vehicle skirts around the obstacle at a minimum radius; otherwise, it is applied minimally to avoid the obstacle. Figure 2d shows a sample convergence plot for Levenberg–Marquart optimization. The loss function is reduced by 9 orders of magnitude from the initial value, which is based on randomly selected PoNN output weights. In Figure 2e, we show how the algorithm adjusts the trajectory as the time constraint is relaxed. With a longer time horizon, the vehicle traces a wider disc around the obstacle, thus expending less control in the lateral direction. The longer path ensures arrival at the specified time. Figure 2f zooms in on the magnitude of control effort for the 21-s flight. Note that the peak magnitude occurs later than in Figure 1b and is closer to the approximately 20-m/s^2 mean during the initial S-turn. Thus, the wider radius reduces the total control effort.

We repeated the first scenario, allowing the algorithm to optimize for the shortest time as well as control effort, obstacle avoidance, and ground avoidance with thrust prohibited. The initial velocity was modified to give the vehicle enough initial energy to surmount the obstacle, so a no-thrust solution was tractable. The results are shown in Figure 3. In Figure 3a, the nominal trajectory with an initial velocity 45° above the horizon in the line-of-sight plane is depicted. Figure 3c shows the corresponding control effort. Note that the initial "Z-force" is upward at about 5 m/s^2. Z-force is upward for most of the flight except a short period where downward force is applied to keep close to the obstacle boundary prior to apogee. Z-force is less than gravity, however, so what is depicted is a gentle departure from a ballistic trajectory. X-force has a large initial value, however, since the flight path is pointed above the horizon, this force is transverse to the flight path plus some drag. The vehicle reduces its initial velocity to reduce the turn radius at apogee. It can always apply a modest amount of lift without violating the no-thrust policy. Figure 3d shows the magnitude of control effort and the numerator of the projection of control effort onto the flight path. The solution abides by the no-thrust restriction. In Figure 3b, we show four trajectories with minimum time as initial cross-range velocity is monotonically increased. The initial velocity vectors and computed minimum objective times are tabulated in Table 2. As the initial cross-range velocity is increased, the trajectory is inclined to ride out the initial swerve from line of sight. Essentially, this results in each subsequent trajectory taking a "great circle" type route with more pronounced incline from the vertical than its predecessor.



a) Nominal, zero initial cross-range velocity

b) Monte-Carlo on cross-range velocity

c) Control effort, nominal min time case

d) Checking compliance with zero thrust constraint

Figure 3. Case 3: Minimum time, minimum control effort, and thrust prohibited.

In Table 2, note that the absolute minimum time occurs with a large initial cross-range velocity.

Table 2. Minimum time for increasing initial cross-range velocity.

Initial velocity (m/s)			Minimum time
V_X	V_Y	V_Z	
53.7401	0.000	–53.7401	13.7877
53.7401	11.3999	–53.7401	13.7988
53.7401	27.24	–53.7401	12.0940
53.7401	34.2038	–53.7401	12.0372
53.7401	59.4484	–53.7401	13.9978

3.5 Berlin Airlift Dead Stick Landing Scenario

We have named the last scenario "Berlin Airlift" after the first major campaign of the newly founded United States Air Force (USAF) in 1948–1949. USAF transport aircraft were tasked to supply the city of west Berlin early in the Cold War due to a divided Germany and a road blockade that prevented resupply by truck. The pilots needed to maintain altitude until the last possible minute before descending into Tempelhof Airport. This scenario will emulate that situation by requiring the vehicle to avoid an obstacle placed along the glide path near the runway threshold and descend and reach favorable conditions for a safe landing. To make things more interesting, we keep the zero-thrust restriction from previous scenarios.

The vehicle starts 800 m from the designated touchdown point at 150 m above the surface. From there, it must clear a 150-m obstacle that is centered 338 m from the touch-down point. After clearing the obstacle, it must make a steep descent, then pull up (flare) to reduce the glide slope to 4° or less prior to touch down. To keep the airspeed up, we fix the final time at 16 s. The initial velocity is 76 m/s in level flight, which is 146 knots.

To enforce the glide slope restriction at the touch-down point, we devise one final constraint that limits the velocity vector depression angle to 4°. Since a downward vertical velocity is positive, we can write

$$\left\|\vec{\boldsymbol{V}}\right\| \sin \boldsymbol{\gamma}_{\mathbf{gs}} \geq \boldsymbol{V}_{\boldsymbol{z}} \geq \mathbf{0} \tag{46}$$

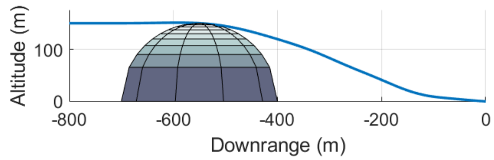
Thus,

$$C_{\gamma} = \boldsymbol{V}_{\boldsymbol{z}} - \left\|\vec{\boldsymbol{V}}\right\| \sin \boldsymbol{\gamma}_{\mathbf{gs}} \leq \mathbf{0} \tag{47}$$

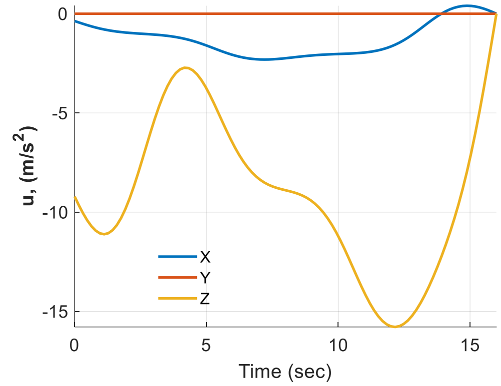
and we employ one last instance of the Fischer–Burmeister function:

$$\mathcal{L}_7 = \boldsymbol{w}_{\boldsymbol{\gamma}} \left(\phi\left(C_{\gamma}, \mu_{\gamma}\right) \right) \tag{48}$$

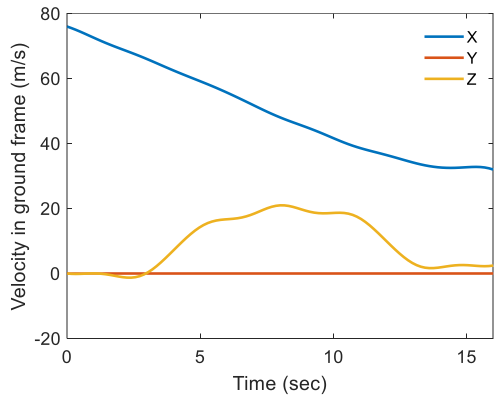
Figure 4 shows the results of this scenario after 115 optimization iterations. In Figure 4a, the optimized trajectory shows the vehicle skimming the top of the obstacle, descending rapidly, then pulling up to reach the touch-down point with a manageable glide path. Figure 4b shows the control effort in the ground frame. The commanded lift has an initial pulse above 1 g to clear the obstacle. The controller then unloads, allowing gravity to pull the vehicle into a steep dive. At 12 s, lift is increased to >1.5 g, indicating the pull up to final glide path. Drag force is increased after clearing the obstacle to prevent high speed and more lift required during the flare maneuver.



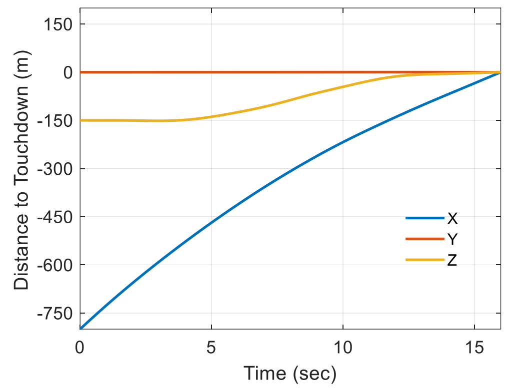
a) Zero thrust, $t_f = 16$ s, vehicle clears obstacle, then dives and flares for touchdown



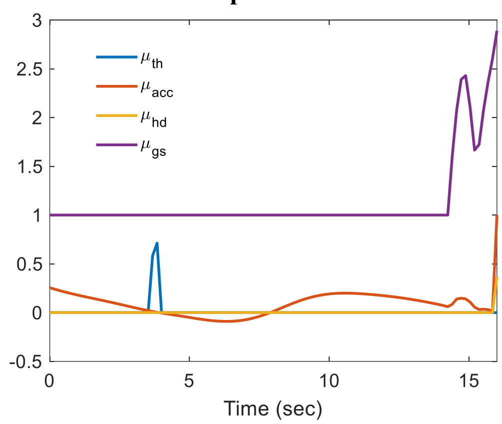
b) Control effort in ground frame. X component largely drag. Lift never crosses zero and reaches 1.5 g during flare



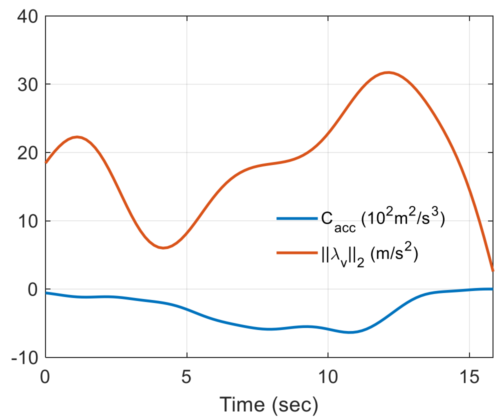
c) Velocity components in ground frame. Sink rate ramps up after clearing obstacle, then tapers off in flare



d) Distance from touchdown in 3D. Descent begins around 4.5 s after clearing obstacle.



e) All four inequality multipliers. Glide slope only tuned and effective during last 1.6 s.



f) Magnitude of control effort and satisfaction of no-thrust constraint

Figure 4. Scenario 4: Vehicle must fly a gliding trajectory over an obstacle on final approach, dive to intercept glide path, and flare to arrest the descent rate before touchdown within a final time. Final time is fixed at 16 s. w_v =0.3, $w_{\lambda r}$ =0.1, $w_{\lambda v}$=10; w_{gs}=7, w_a=48, w_{acc}=0.17, w_{gli} =1.

Figure 4c shows the velocity components in the ground frame. Forward velocity is bled off gradually due to drag. Recall that vertical velocity is positive down. There is a small climb around 2.5 s to clear the obstacle. After that, descent rate increases, reaching a peak of nearly 20 m/s. The descent rate is arrested around 13 s as the flare maneuver is executed. Figure 4d shows the three components of position in relation to the touch-down point at $\{0, 0, 0\}$.

Figure 4e depicts the Lagrange multipliers acting on the inequality constraints as functions of time. The blue line is the multiplier associated with obstacle avoidance and is only active for a brief period when the slight climb is commanded. The red line is the no-thrust restriction, which is active throughout the trajectory. The yellow line is the hard deck constraint, which becomes active only as the vehicle gets very close to the touch-down point. The purple line is the glide slope constraint, which is designed to only be active during the last 10% of flight. Thus, for good conditioning, only the points in the last 1.6 s are tuned with the rest remaining at their initial value of unity. To satisfy the glide path constraint, the pull up begins nearly 4 s prior to touchdown. All points computed in the collocation are connected to satisfy all constraints.

4. Conclusion

The PoNN method provides a powerful way to determine the optimal path for an aerospace vehicle navigating in airspace that is restricted by obstacles. This approach allows the user to define any number of restrictions as inequality constraints or boundary conditions. Using the X-TFC method, the boundary conditions are automatically satisfied. By using the Fischer–Burmeister function to enforce complementary slackness rather than the explicit inequality constraints, all constraints are treated as equality constraints. These techniques reduce the problem to a suite of loss functions that are optimized simultaneously using a nonlinear least-squares search for the PoNN network output weights and Euler–Lagrange multipliers.

In this report, we have demonstrated the success of a PoNN in solving optimization problems with up to four inequality constraints. We used a 3DOF point mass model to keep the dynamics manageable, such that the least-squares search could be powered by analytic gradient calculations. The Fischer–Burmeister technique makes it easy to enforce a no-thrust constraint on the solution without explicitly writing the dynamics to exclude thrust. Thus, gliding trajectories for reusable launch vehicles and projectiles can be found through this method.

5. References

1. Elnagar G, Kazemi MA, Razzaghi M. The pseudospectral Legendre method for discretizing optimal control problems. IEEE Trans Autom Control. 1995 Oct;38(10):1793–1796.

2. Mortari D. The theory of connections: connecting points. Mathematics. 2017;5(4):57. https://doi.org/10.3390/math5040057

3. Mortari D. Least-squares solution of linear differential equations. Mathematics. 2017;5(4):48. https://doi.org/10.3390/math5040048

4. D'Ambrosio A, Furfaro R. Learning fuel-optimal trajectories for space applications via Pontryagin neural networks. Aerospace. 2024;11:228. https://doi.org/10.3190/aerospace11028228

5. D'Ambrosio A, Schiassi E, Curti R, Furfaro R. Pontryagin neural networks with functional interpolation for optimal intercept problems. Mathematics. 2021;9:996. https://doi.org/10.3190/math9090996

6. D'Ambrosio A, Benedikter B, Furfaro R. Physics-informed Pontryagin neural networks for path-constrained optimal control problems. AIAA SciTech 2025 Forum; 2025 Jan 6–10; Orlando, FL. https://doi.org/10.2304/6.2025-199

7. Bryson AE, Ho Y. Applied optimal control. Ginn and Company; 1969.

8. Burchett BT. Methods for determining the initial state of a projectile equipped with onboard strapdown sensors. Proceedings of the AIAA Aviation Forum 2020; 2020 June 15–19; virtual event.

Appendix. Switching Function Definitions

Table A-1. Switching functions for three constraints with two constraints on f and one on f', defined on domain of z ∈ $[z_0, z_f]$.

	Initial value $\boldsymbol{\Omega_1(z)}$	Final value $\boldsymbol{\Omega_2(z)}$	Initial derivative $\boldsymbol{\Omega_3(z)}$
(.)	$\frac{(z_f - z)(z - 2z_0 + z_f)}{\Delta z^2}$	$\frac{(z - z_0)^2}{\Delta z^2}$	$\frac{(z_f - z)(z - z_0)}{\Delta z}$
$\frac{d}{dz}(.)$	$\frac{-2(z - z_0)}{\Delta z^2}$	$\frac{2(z - z_0)}{\Delta z^2}$	$\frac{(-2z + z_0 + z_f)}{\Delta z}$
$\frac{d^2}{dz^2}(.)$	$\frac{-2}{\Delta z^2}$	$\frac{2}{\Delta z^2}$	$\frac{-2}{\Delta z}$

List of Symbols, Abbreviations, and Acronyms

3D	Three-Dimensional
3DOF	3 Degrees of Freedom
CE	Constrained Expression
CG	Center of Gravity
PMP	Pontryagin's Maximum Principle
PoNN	Pontryagin Neural Network
TFC	Theory of Functional Connection
USAF	United States Air Force
X-TFC	Extreme Theory of Functional Connection

Nomenclature

$\{x,y,z\}$	vehicle cg position in ground frame [m]
$\{V_x, V_y, V_z\}$	$\boldsymbol{V}$ vehicle linear velocity in the body or no-roll frame [m/s]
$\{a_x, a_y, a_z\}$	$\boldsymbol{a}$, total acceleration vector in ground frame [m/s^2]
V	$\lVert\boldsymbol{V}\rVert_2$, speed relative to ground frame [m/s]
$\boldsymbol{f}$	system dynamics $\mathbb{R}^n$
$\boldsymbol{g}$	path constraints $\mathbb{R}^p$
$\boldsymbol{L}$	cost functional
m	vehicle mass [kg]
$\boldsymbol{x}$	system state vector $\mathbb{R}^n$
t	time [s]
$\boldsymbol{\alpha}$	constraint at initial time, t_0
$\boldsymbol{\omega}$	constraint at final time, t_f
$\boldsymbol{u}$	control vector $\mathbb{R}^m$
$\boldsymbol{\lambda}$	Lagrange multipliers (dynamics)
$\boldsymbol{\mu}$	Lagrange multipliers (path constraints)
$\boldsymbol{\Omega_i}$	Switching function
$c_\gamma, s_\gamma, t_\gamma$=	$\cos(\gamma)$, $\sin(\gamma)$, $\tan(\gamma)$, respectively

Subscript

$b2i$	ground to body frame
$b2e$	body to Euler frame
0	initial conditions
f	final (objective) conditions

Superscript

T	matrix transpose

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
1 (PDF)	DEVCOM ARL FCDD RLB CI TECH LIB