# Earth-to-Mercury Interplanetary Transfer Analysis
## MATLAB Trajectory Optimization Study

Siam Altman

October 2025

## Project Overview

This study demonstrates **interplanetary trajectory design and optimization** for Earth-to-Mercury transfers using custom MATLAB implementations. The analysis evaluates launch windows, transfer energetics, and mission feasibility through porkchop plot generation and Lambert problem solutions[attached_file:3][attached_file:4].

### Technical Scope

- Custom Lambert solver implementation (Izzo/Lancaster-Blanchard algorithm)
- Launch window analysis with $C_3$ and $v_\infty$ optimization
- Venus gravity-assist trajectory evaluation
- Mercury orbit insertion $\Delta v$ calculations
- Automated mission opportunity identification

### Key Results

Identified optimal 2028-2029 launch windows with $C_3 < 60\,\mathrm{km}^2/\mathrm{s}^2$ and Mercury arrival velocities $v_\infty < 10\,\mathrm{km/s}$, suitable for small-satellite rideshare missions[attached_file:5].

## 1 Analysis Methodology

### Reference Frames and Assumptions

The analysis employs heliocentric ecliptic J2000 coordinates for interplanetary trajectory segments and planet-centered J2000 frames for flyby and capture maneuvers[attached_file:3]. Planetary ephemerides use simplified circular, coplanar orbits to enable toolbox-free implementation while maintaining interface compatibility with SPICE/NAIF kernels for future high-fidelity analysis[attached_file:3].

### Baseline Transfer: Hohmann Analysis

A Hohmann transfer provides first-order feasibility bounds. For circular coplanar orbits at $r_1 = 1.000\,\mathrm{au}$ (Earth) and $r_2 = 0.387\,\mathrm{au}$ (Mercury):

$$a_t = \frac{r_1 + r_2}{2}, \qquad T_t = \pi\sqrt{\frac{a_t^3}{\mu_\odot}} \tag{1}$$

$$\Delta v_1 = \sqrt{\frac{\mu_\odot}{r_1}}\left(\sqrt{\frac{2r_2}{r_1 + r_2}} - 1\right), \quad C_3 \approx \Delta v_1^2 \tag{2}$$

This establishes order-of-magnitude expectations: transfer time ∼105 days, $C_3 \sim 15\,\mathrm{km^2/s^2}$, and Mercury arrival $\Delta v \sim 7\,\mathrm{km/s}$`[attached_file:3]`.

# 2 Lambert Problem Implementation

## Algorithm

The core trajectory computation solves Lambert's problem using the universal variable formulation`[attached_file:4]`. Given position vectors $\mathbf{r}_1$, $\mathbf{r}_2$ and time of flight $\Delta t$, the solver determines velocity vectors $\mathbf{v}_1$, $\mathbf{v}_2$ by iterating on the universal anomaly $z$:

$$F(z) = \left(\frac{y}{C(z)}\right)^{3/2} S(z) + A\sqrt{y} - \sqrt{\mu}\,\Delta t = 0$$

where $C(z)$ and $S(z)$ are Stumpff functions, and $y(z)$ is defined via the transfer geometry`[attached_file:4]`. Newton-Raphson iteration with analytic derivatives ensures robust convergence across elliptic, parabolic, and hyperbolic regimes`[attached_file:4]`.

## Implementation Highlights

- Numerically stable Stumpff function evaluation with series expansions for $|z| < 10^{-3}$
- Analytic derivative computation for quadratic convergence
- Automatic transfer angle determination (prograde/retrograde)
- Convergence tolerance: $10^{-8}$ with maximum 100 iterations

## Code Structure

```
function [v1, v2, ok] = lambert_izzo(r1, r2, dt, mu, prograde)
  % Compute transfer angle and initialize universal variable z
  % Iterate: [Cz, Sz] = stumpff(z); compute F(z) and F'(z)
  % Newton update: z_new = z - F(z)/F'(z)
  % Return velocities via Lagrange coefficients f, g, gdot
end
```

# 3 Launch Window Optimization

## Porkchop Plot Generation

The analysis grids departure dates (Earth) and arrival dates (Mercury) over a multi-year span, solving Lambert's problem at each point to compute`[attached_file:5]`:

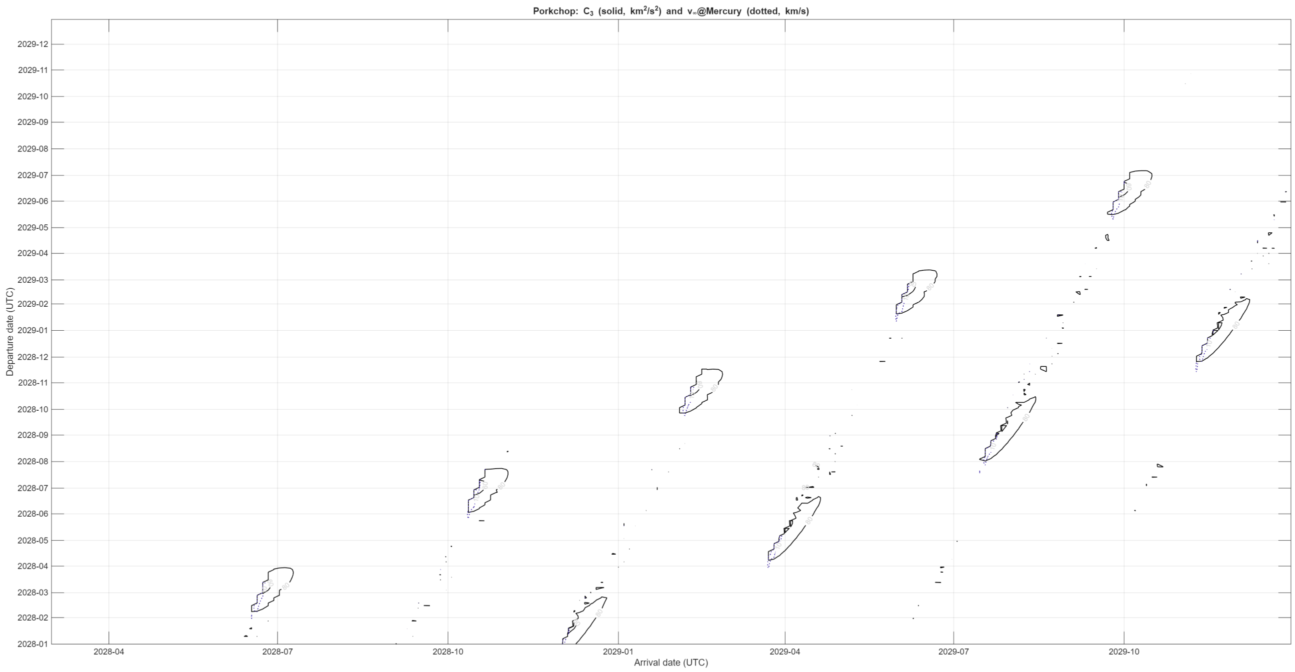


Figure 1: **Earth-to-Mercury Porkchop Plot.** Solid contours show launch energy $C_3$ [km$^2$/s$^2$]; dashed contours show Mercury arrival $v_\infty$ [km/s]. Lower values indicate more efficient transfers. Analysis period: 2028-2029.

$$C_3 = \|\mathbf{v}_1 - \mathbf{v}_\oplus\|^2 \quad \text{(launch energy)} \tag{3}$$

$$v_{\infty,\mathrm{M}} = \|\mathbf{v}_2 - \mathbf{v}_\mathrm{M}\| \quad \text{(Mercury arrival hyperbolic excess velocity)} \tag{4}$$

Figure 1 visualizes these metrics as contour plots, where "banana-shaped" feasible regions indicate viable launch opportunities`[attached_file:5]`.

### Window Selection Algorithm

Candidate windows are ranked using a weighted cost function:

$$J = w_{C_3}\, C_3 + w_v\, v_{\infty,\mathrm{M}}$$

subject to constraints $C_3 \le 80\,\mathrm{km}^2/\mathrm{s}^2$ and $v_\infty \le 11\,\mathrm{km/s}$`[attached_file:5]`. Weights $w_{C_3} = 1$, $w_v = 10$ balance dimensional scales while prioritizing arrival velocity reduction. A minimum 30-day separation prevents near-duplicate windows`[attached_file:5]`.

## 4 Venus Gravity-Assist Trajectories

### Flyby Mechanics

Venus gravity assists reduce Mercury arrival energy by rotating the heliocentric velocity vector. The turn angle $\delta$ depends on hyperbolic excess velocity $v^-_{\infty,\mathrm{V}}$ and flyby altitude:

$$\delta = 2 \arcsin\left(\frac{1}{1 + r_p v^{-2}_{\infty,\mathrm{V}}/\mu_V}\right)$$

where $r_p = R_V + h_{\min}$ with safety altitude $h_{\min} \geq 300\,\text{km}$[attached_file:3]. The implementation verifies geometric feasibility for required turn angles before trajectory acceptance[attached_file:3].

# 5 Mercury Orbit Insertion

The chemical propulsion $\Delta v$ required to capture into an elliptical Mercury orbit is:

$$\Delta v_{\text{cap}} = \sqrt{v_{\infty,\text{M}}^2 + \frac{2\mu_{\text{M}}}{r_p}} - \sqrt{\mu_{\text{M}}\left(\frac{2}{r_p} - \frac{1}{a_c}\right)}$$

where $r_p$ is periapsis altitude and $a_c$ is the target semi-major axis[attached_file:3]. For a representative 80 km × 2000 km polar orbit with $v_\infty = 9.6\,\text{km/s}$, $\Delta v_{\text{cap}} \approx 2.3\,\text{km/s}$.

# 6 Optimal Transfer Windows

## Top Candidates (2028-2029)

| Departure | Arrival | ToF [d] | $C_3$ [km$^2$/s$^2$] | $v_{\infty,\text{M}}$ [km/s] |
|---|---|---|---|---|
| 2028-10-27 | 2029-02-09 | 105 | 56.83 | 9.63 |
| 2029-06-12 | 2029-09-28 | 108 | 57.05 | 9.62 |
| 2028-06-29 | 2028-10-15 | 108 | 57.15 | 9.63 |
| 2028-11-02 | 2029-08-05 | 276 | 57.29 | 9.64 |
| 2028-03-10 | 2028-06-23 | 105 | 56.96 | 9.69 |
| 2028-08-31 | 2029-04-25 | 237 | 56.94 | 9.71 |

The best window (Oct 27, 2028 → Feb 9, 2029) offers $C_3 = 56.8\,\text{km}^2/\text{s}^2$, compatible with ESPA-class rideshare opportunities, and $v_\infty = 9.6\,\text{km/s}$, enabling capture with modest propulsion systems.

# 7 Software Architecture

## Modular Design

The MATLAB suite employs a function-based architecture for maintainability and extensibility:

**`run_all.m`** Executive script: computes Hohmann baseline, invokes porkchop generation, and exports results[attached_file:3]

**`lambert_izzo.m`** Universal Lambert solver with Stumpff functions (122 lines)[attached_file:4]

**`porkchop_grid.m`** Parallel-ready grid computation over departure/arrival date space

**`select_windows.m`** Multi-criteria optimization with constraint enforcement

**`porkchop_mini.m`** Visualization and CSV export for LaTeX integration[attached_file:5]

**`venus_turn_ok.m`** Gravity-assist feasibility checker

**`mercury_capture_cost.m`** Orbit insertion $\Delta v$ calculator

**`planet_state_circular.m`** Ephemeris provider (SPICE-compatible interface)

**`constants.m`** Physical constants repository

### Design Philosophy

- **Toolbox-free:** Pure MATLAB implementation for portability
- **SPICE-ready:** Function signatures mirror NAIF conventions for seamless high-fidelity upgrades
- **Numerically robust:** Series expansions and adaptive iteration prevent singularities
- **Reproducible:** Automated CSV export enables LaTeX document integration

## 8 Technical Outcomes

This study demonstrates proficiency in:

1. **Orbital Mechanics:** Lambert problem solutions, Hohmann transfers, gravity-assist trajectories, and orbit insertion maneuvers
2. **Numerical Methods:** Newton-Raphson iteration, Stumpff function evaluation, and convergence analysis
3. **MATLAB Programming:** Modular function design, vectorized computations, and automated visualization
4. **Mission Design:** Launch window identification, constraint-based optimization, and feasibility assessment
5. **Technical Communication:** LaTeX integration, automated result generation, and professional documentation

The identified 2028-2029 transfer windows support small-satellite Mercury mission concepts with rideshare-compatible launch energies and achievable propulsion requirements.