🎢 Halo Orbits & Invariant Manifolds

Invariant manifolds about Halo orbits and their applications in the Circular Restricted Three-body Problem.

- Joe Carpinelli
- December, 2020
- ENAE601 Final Project

Presentation Mode: Toggle Presentation

📋 Project Overview

In the context of astrodynamics, **manifolds are groups of trajectories that move toward or away from Lagrange points**. To use invariant manifolds for interplanetary travel, several concepts must be developed and built on: lagrange points, periodic and quasi-periodic orbits within the Circular Restricted Three-body Problem, and manifolds about periodic orbits [1].

Outline

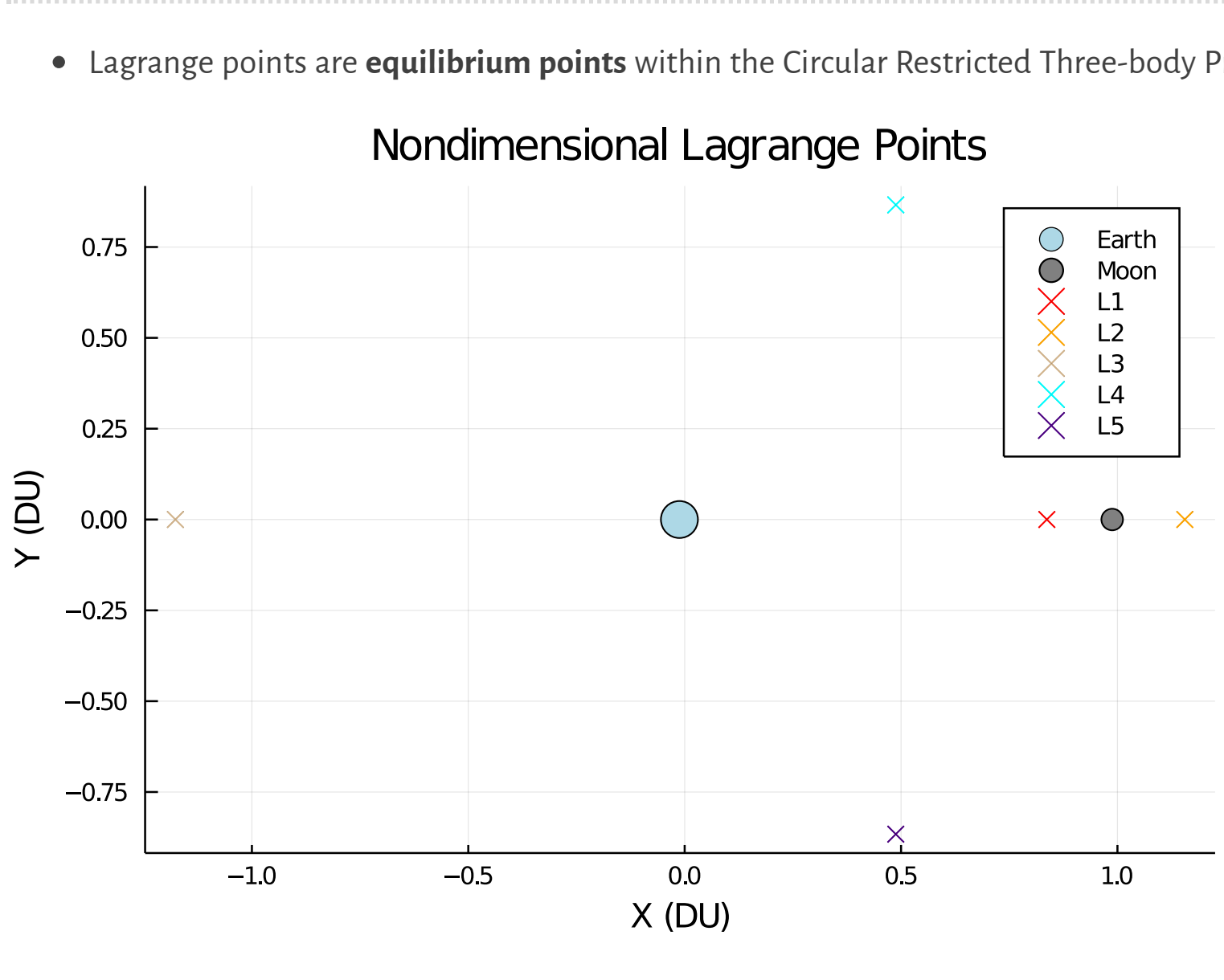
- Brief review of Lagrange points
- Periodic orbits (specifically, the subsection known as Halo orbits)
- Finding Halo orbits (both analytically, and numerically)
- Invariant manifolds about Halo orbits

Primary Reference

- Megan Rund's Masters Thesis at California Polytechnic State University [1]

Lagrange Points

- Lagrange points are **equilibrium points** within the Circular Restricted Three-body Problem



Stability at Lagrange Points

- Like equilibrium points for all nonlinear systems, Lagrange points can be **stable or unstable**
- We can find the stability of an equilibrium point by analysing **eigenvalues** of the Jacobian of the state vector

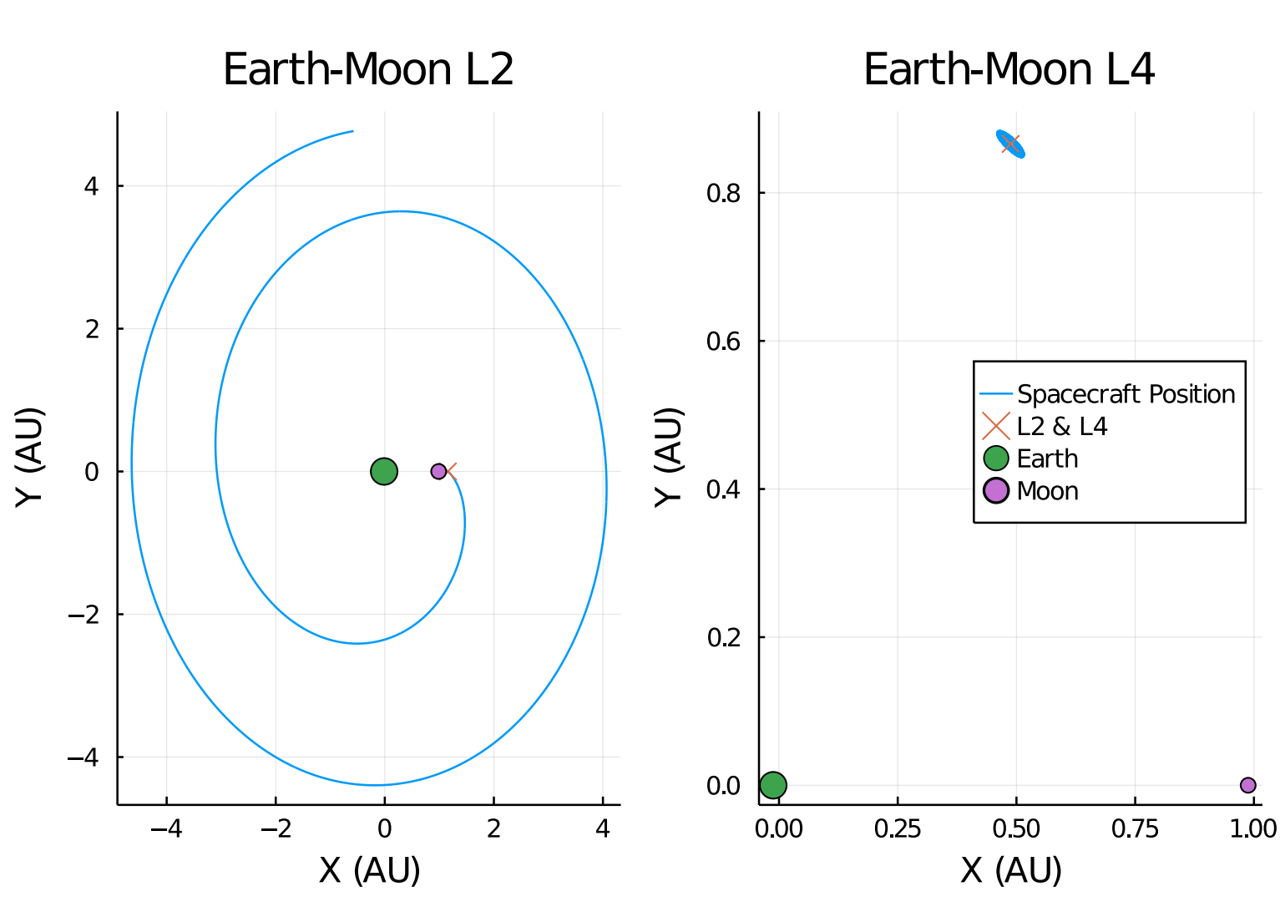
In the 2D case:

$$\begin{bmatrix}\dot\zeta\\ \dot\beta\\ \ddot\zeta\\ \ddot\eta\end{bmatrix} = \begin{bmatrix}0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ U_{xx} & U_{xy} & 0 & 2\\ U_{yx} & U_{yy} & -2 & 0\end{bmatrix}\begin{bmatrix}\zeta\\ \eta\\ \dot\zeta\\ \dot\eta\end{bmatrix}$$

$$\lambda^4 + (4 - U_{xx} - U_{yy})\lambda^2 + U_{xx}U_{yy} - U_{xy}^2 = 0$$

Stability at Lagrange Points: Examples

- Earth-Moon L4 is **stable**, and Earth-Moon L2 is **unstable**



)

Periodic Orbits

Libration Orbit Families

- Orbits about Lagrange points are known as **Libration orbits**
- **Lyapunov** orbits are two-dimensional Libration orbits
- **Lissajous** orbits are three-dimensional, semi-periodic Libration orbits
- **Halo** orbits are three-dimensional, theoretically periodic Libration orbits

Finding Halo Orbits

- The calculations required to solve for Halo orbits are numerically sensitive
- We covered the procedure for iteratively solving for Halo orbits **numerically** in **Lecture 16**; this requires an initial guess

- **How do we find an initial guess for a Halo orbit?**

😇 Analytical Halo Solution

- The Circular Restricted Three-body Dynamics can be derived using "Legendre poly- nomials", and there exists a third-order approximation (shown below) [5]
- We can choose parameters to remove **unstable** (secular) terms from the expansion

Third Order CR3BP Expansion

$$\begin{aligned}\ddot x - 2\dot y - (1 + 2c_2)x &= \frac{3}{2}c_3(2x^2 - y^2 - z^2)\\ &\quad +2c_4x(2x^2 - 3y^2 - 3z^2) + O(4),\\ \ddot y + 2\dot x + (c_2 - 1)y &= -3c_3xy - \frac{3}{2}c_4y(4x^2 - y^2 - z^2) + O(4),\\ \ddot z + c_2z &= -3c_3xz - \frac{3}{2}c_4z(4x^2 - y^2 - z^2) + O(4).\end{aligned}$$

Algorithm Inputs

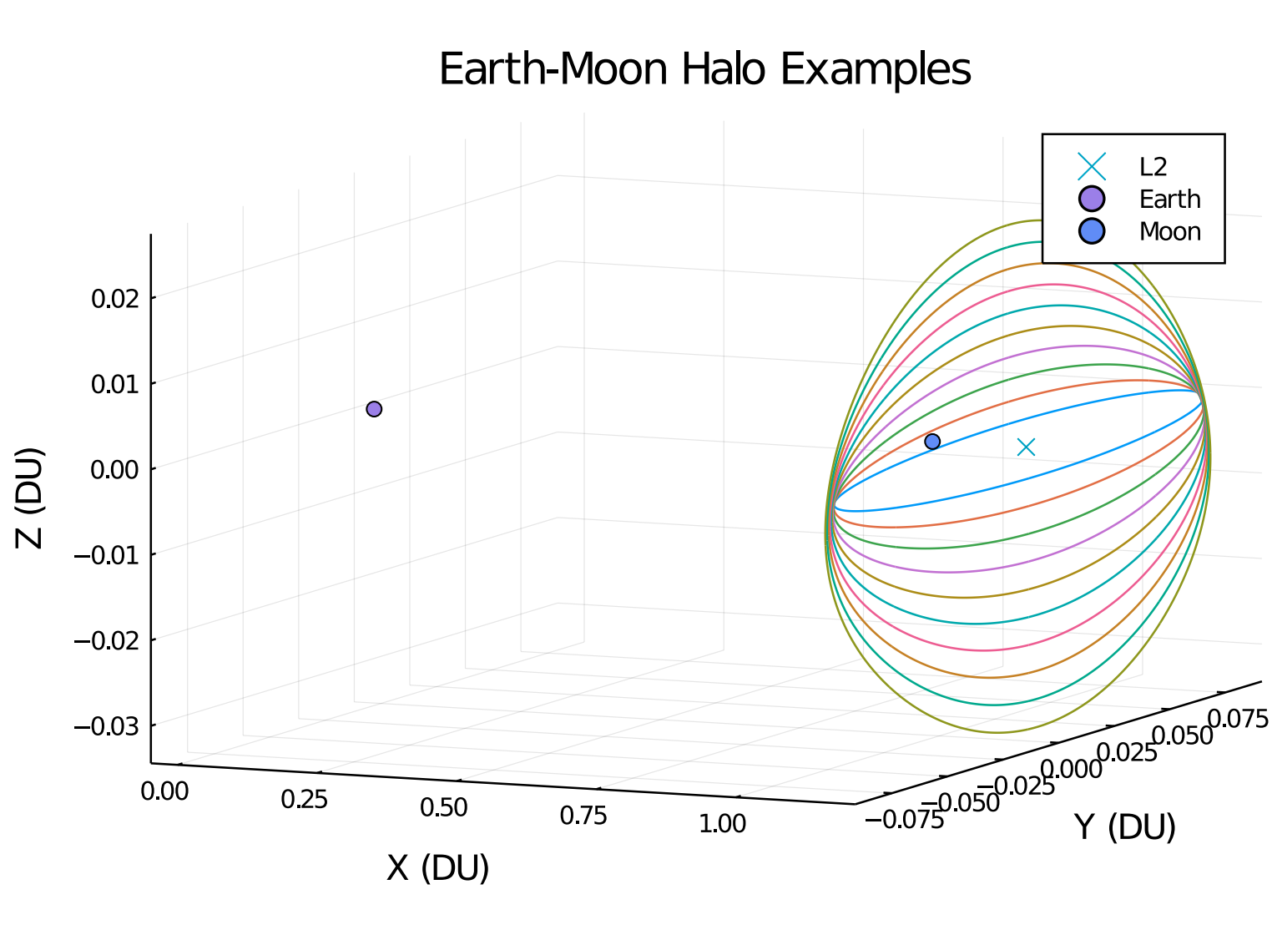
- Nondimensional mass parameter μ
- Nondimensional $Z - axis$ amplitude for the desired Halo orbit
- Lagrange point to orbit (L1 or L2)

Algorithm Outputs

- Position vector $\vec{r}_0$
- Velocity vector $\vec{v}_0$
- Estimated orbital period T

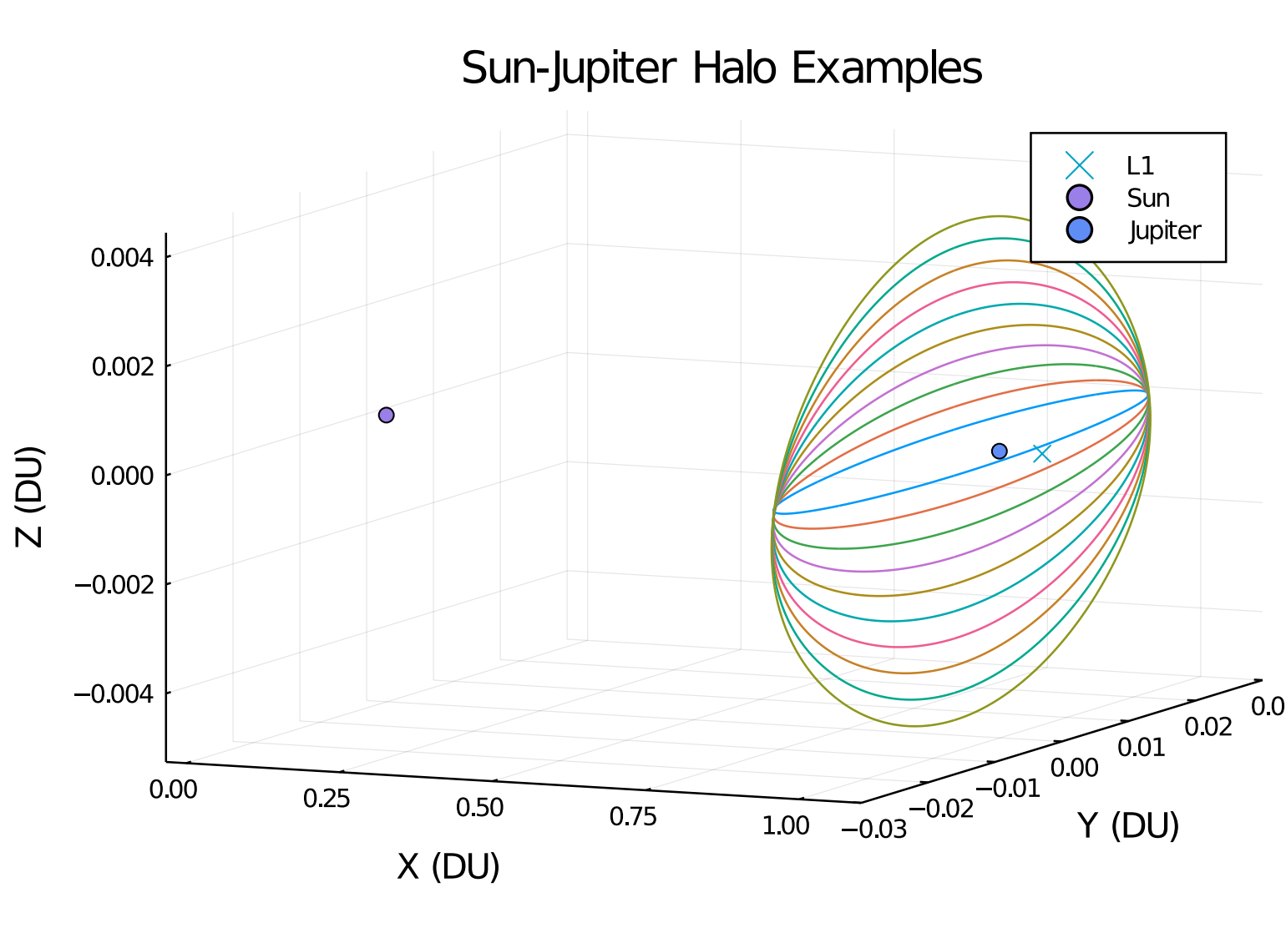
Analytical Halo Examples: Earth-Moon L2

NOT numerically propagated!



Analytical Halo Examples: Sun-Jupiter L1

NOT numerically propagated!



🧮 Numerical Halo Solution

- As discussed in **Lecture 16**, we can append the **state transition matrix** $\Phi(t_0 + t_i, t_0)$ to our state vector, and iteratively change initial conditions to **numerically find a periodic orbit** [1]

1. $\Phi = I,\ \vec{x}_0 = [x_0 \quad 0 \quad z_0 \quad 0 \quad \dot y_0 \quad 0 \quad \Phi_1 \quad \Phi_2 \quad \Phi_3 \quad \Phi_4 \quad \Phi_5 \quad \Phi_6]$
2. Propagate until $y = 0$ again; $\dot\Phi = F\Phi$ where $F = \begin{bmatrix}0 & I_3\\ U_{XX} & 2\Omega\end{bmatrix}$, U_{XX} is the matrix of second partial derivatives of potential U
3. Calculate $\begin{bmatrix}\delta x_0\\ \delta\dot y_0\end{bmatrix} = \left(\begin{bmatrix}\Phi_{41} & \Phi_{45}\\ \Phi_{61} & \Phi_{65}\end{bmatrix}\frac{1}{\dot y}\begin{bmatrix}\ddot x\\ \ddot z\end{bmatrix}\right)^{-1}\begin{bmatrix}-\dot x\\ -\dot z\end{bmatrix}$
4. Set $x_0 \leftarrow x_0 + \delta x_0$, $\dot y_0 \leftarrow \dot y_0 + \delta\dot y$ and jump back to **Step 1** (until $\dot x$, $\dot z$ are both within some tolerance of zero)

🎢 Dynamics along Halo Orbits

Manifolds Exist

- Each point along a Halo orbit is connected with an **unstable** manifold, and a **stable** manifold
- The unstable manifold **departs** the Halo orbit, and the stable manifold **arrives** at the Halo orbit

- **How can we calculate the perturbation required to shift the spacecraft onto the manifold?**

Finding Manifolds

- We can use **Eigenvectors** of the Jacobian to calculate a state perturbation which will place the spacecraft onto a manifold at each point along the Halo orbit

1. Propagate the Halo orbit for one period T, including the state transition matrix $\Phi(t_0 + t_i, t_0)$
2. Let the final state transition matrix be M; $M = \Phi(t_0 + T, t_0)$
3. Calculate eigenvectors $V^S \leftrightarrow \text{minreal}(\text{eig}(M))$, and $V^U \leftrightarrow \text{maxreal}(\text{eig}(M))$
4. For each point i along the Halo orbit...
   - $V_i^S = \Phi(t_0 + t_i, t_0)V^S$, and $V_i^U = \Phi(t_0 + t_i, t_0)V^U$
   - $X_i^S = X_i \pm \epsilon\frac{V_i^S}{|V_i^S|}$, and $X_i^U = X_i \pm \epsilon\frac{V_i^U}{|V_i^U|}$

Invariant Manifold Example

- Pulled from Rund's Thesis [1]
- Note that stable manifolds need to be propagated *backward* in time from the perturbation along the Halo orbit, becuase they converge on the Halo orbit [1]

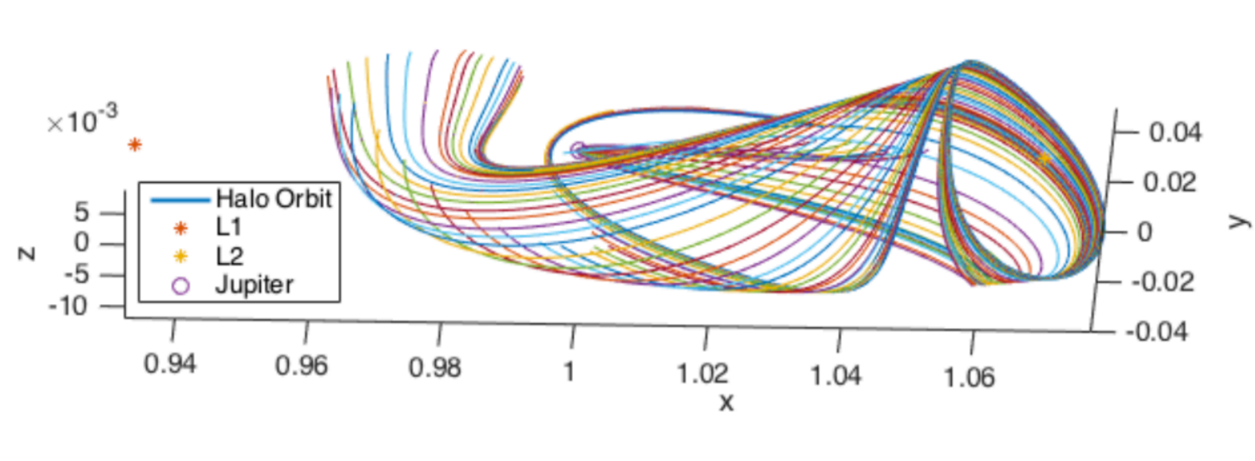


Figure 3.6: Example the stable invariant manifold for Sun-Jupiter L_2.

🪐 Manifold-based Transfer Designs

- **Design Overview** summarized from [1]

Design Overview

1. Find a desired Sun-Earth Halo orbit
2. Place the spacecraft within the **stable manifold** of this Halo orbit
3. Perturb the spacecraft from the Halo onto the **unstable manifold** to return towards Earth, and apply a maneuver to place the spacecraft on a Hyperbolic escape trajectory toward your destination planet
4. At destination planet, apply a maneuver to place spacecraft onto **stable manifold** of destination Halo orbit

🚀 Conclusions

Manifold Transfers

- Lagrange points, and periodic orbits about Lagrange points are surrounded by collections of trajectories called **manifolds**
- Manifolds can depart, or arrive at the Lagrange point / Libration orbit
- Stable manifolds can bring us to Halo orbits for free* (plus the cost to place the spacecraft onto the manifold)

Lessons Learned

- Linear (Eigenvector) analysis works in Astrodynamics too
- Calculations for iterating on Halo orbits are **extremely** numerically sensitive
- Even with more complicated models (CR3BP), we need a patch-conic-like approach for interplanetary mission design

📚 References

[1] Rund, M. S., “Interplanetary Transfer Trajectories Using the Invariant Manifolds of Halo Orbits,” , 2018.

[2] Howell, K. C., “Three-dimensional, periodic,‘halo’orbits,” Celestial mechanics, Vol. 32, No. 1, 1984, pp. 53–71.

[3] Vallado, D. A., Fundamentals of astrodynamics and applications, Vol. 12, Springer Science & Business Media, 2001.

[4] Richardson, D., “Analytical construction of periodic orbits about the collinear points of the Sun-Earth system.” asdy, 1980, p. 127.

[5] Lara, M., Russell, R., and Villac, B., “Classification of the distant stability regions at Europa,” Journal of Guidance, Control, and Dynamics, Vol. 30, No. 2, 2007, pp. 409–418.

[6] Koon, W. S., Lo, M. W., Marsden, J. E., and Ross, S. D., “Dynamical systems, the three-body problem and space mission design,” Free online Copy: Marsden Books, 2008.

[7] Williams, J., Lee, D. E., Whitley, R. J., Bokelmann, K. A., Davis, D. C., and Berry, C. F., “Targeting cislunar near rectilinear halo orbits for human space exploration,” 2017.

[8] Zimovan-Spreen, E. M., Howell, K. C., and Davis, D. C., “Near rectilinear halo orbits and nearby higher-period dynamical structures: orbital stability and resonance properties,” Celestial Mechanics and Dynamical Astronomy, Vol. 132, No. 5, 2020, pp. 1–25.

[9] NASA, NASA’s Lunar Exploration Program Overview, 2020.

[10] Carpinelli, J., “UnitfulAstrodynamics.jl,” https://juliahub.com/ui/Packages/UnitfulAstrodynamics/uJGLZ/, 2020.

⌨️ Source Code

Package Dependencies

- The following packages were used – all are available in Julia's **General** package registry

```
begin

    using Plots
    using Roots
    using PlutoUI
    using Latexify
    using StaticArrays
    using LinearAlgebra
    using ModelingToolkit
    using UnitfulAstrodynamics
    using DifferentialEquations

    μ = nondimensionalize(Earth.μ, Moon.μ)

end;
```

Finding Lagrange Points

```
f₁ = let

    f₁ = lagrangeplot(nondimensionalize(Earth.μ, Moon.μ); labels=["Earth" "Moon"])
    savefig(f₁, "media/fig1.png")

    f₁

end;
```

Unstable Lagrange Point (L2)

```
f₂ = let

    μ = nondimensionalize(Earth.μ, Moon.μ)
    r = lagrange(μ, 2)[:]
    v = [0.0, 0.0, 0.0]

    sys = NondimensionalThreeBodyState(r, v, μ, NaN * u"km", NaN * u"s")
    sols = propagate(sys, 30; save_everystep=true, reltol=1e-16, abstol=1e-16)

    x = [sols.step[i].rₛ[1] for i ∈ 1:length(sols.step)]
    y = [sols.step[i].rₛ[2] for i ∈ 1:length(sols.step)]
    z = [sols.step[i].rₛ[3] for i ∈ 1:length(sols.step)]

    f₂ = plot(x,y; label="Spacecraft Position")
    scatter!(f₂, [[i] for i ∈ lagrange(μ, 2)[1:2]]...; label="L2", markershape=:x)
    scatter!(f₂, [[i] for i ∈ sys.r₁][1:2]...; label="Earth", markersize=7)
    scatter!(f₂, [[i] for i ∈ sys.r₂][1:2]...; label="Moon")
    plot!(f₂; title="Spacecraft Propagated from Earth-Moon L2",
              xlabel="X (AU)",
              ylabel="Y (AU)")

    savefig(f₂, "media/fig2.png")

    f₂

end;
```

Stable Lagrange Point (L4)

```
f₇ = let

    μ = nondimensionalize(Earth.μ, Moon.μ)
    r = lagrange(μ, 4)[:] * (1 + 1e-3)
    v = [0.0, 0.0, 0.0]

    sys = NondimensionalThreeBodyState(r, v, μ, NaN * u"km", NaN * u"s")
    sols = propagate(sys, 1000; save_everystep=true, reltol=1e-16, abstol=1e-16)

    x = [sols.step[i].rₛ[1] for i ∈ 1:length(sols.step)]
    y = [sols.step[i].rₛ[2] for i ∈ 1:length(sols.step)]
    z = [sols.step[i].rₛ[3] for i ∈ 1:length(sols.step)]

    f₇ = plot(x,y; label="Spacecraft Position")
    scatter!(f₇, [[i] for i ∈ lagrange(μ, 4)[1:2]]...; label="L4", markershape=:x)
    scatter!(f₇, [[i] for i ∈ sys.r₁][1:2]...; label="Earth", markersize=7)
    scatter!(f₇, [[i] for i ∈ sys.r₂][1:2]...; label="Moon")
    plot!(f₇; title="Spacecraft Perturbed from Earth-Moon L4",
              xlabel="X (AU)",
              ylabel="Y (AU)")

    savefig(f₂, "media/fig2.png")

    f₇

end;
```

Plot Analytical Halo Orbit

```
function haloplot(μ, L, Z, H, str1="Mass 1", str2="Mass 2"; kwargs...)

    if L==:L1
        l = 1
    else
        l = 2
    end

    defaults = (; title="Analytical Halo Solutions",
                  xlabel="X (DU)", ylabel="Y (DU)", zlabel="Z (DU)")
    options  = merge(defaults, kwargs)

    fig = plot(; options...)
    for z ∈ Z
        r,v,T = halo_analytic(μ; L=L, Zₐ=z, hemisphere=H, steps=1000)
        x = r[:,1]
        y = r[:,2]
        z = r[:,3]
        plot!(fig, x, y, z; label=:none)

    end

    scatter!(fig, [[v] for v ∈ lagrange(μ, l)]...;
             label=string("L",L), markershape=:x)
    scatter!(fig, [-μ], [0], [0]; label=str1)
    scatter!(fig, [1-μ], [0], [0]; label=str2)

    return fig

end;
```

Northern Sun-Jupiter Halos

```
f₃ = let

    μ = nondimensionalize(Sun.μ, Jupiter.μ)
    Z = [x / 10 for x ∈ 1:10]
    f₃ = haloplot(μ, 1, Z, :northern, "Sun", "Jupiter";
                  title="Northern Analytical Halo Solutions")

    savefig(f₃, "media/fig3.png")

    f₃


end;
```

Southern Sun-Jupiter Halos

```
f₄ = let

    μ = nondimensionalize(Sun.μ, Jupiter.μ)
    Z = [x / 10 for x ∈ 1:10]
    f₄ = haloplot(μ, 1, Z, :southern, "Sun", "Jupiter";
                  title="Southern Analytical Halo Solutions")

    savefig(f₄, "media/fig4.png")

    f₄


end;
```

Northern Earth-Moon Halos

```
f₅ = let

    μ = nondimensionalize(Earth.μ, Moon.μ)
    Z = [x / 10 for x ∈ 1:10]
    f₅ = haloplot(μ, 2, Z, :northern, "Earth", "Moon";
                  title="Northern Analytical Halo Solutions")

    savefig(f₅, "media/fig5.png")

    f₅


end;
```

Southern Earth-Moon Halos

```
f₆ = let

    μ = nondimensionalize(Earth.μ, Moon.μ)
    Z = [x / 10 for x ∈ 1:10]
    f₆ = haloplot(μ, 2, Z, :southern, "Earth", "Moon";
                  title="Southern Analytical Halo Solutions")

    savefig(f₆, "media/fig6.png")

    f₆


end;
```

Numerically Produced Halo

```
f₈ = let

    r,v,T = halo(μ, L=2, Zₐ=0.05, ϕ=0.05, max_iter=20)
    sys = NondimensionalThreeBodyState(r,v,μ, NaN*u"km", NaN*u"s")
    sols = propagate(sys, T)
    x = [x.rₛ[1] for x ∈ sols.step]
    y = [x.rₛ[2] for x ∈ sols.step]
    z = [x.rₛ[3] for x ∈ sols.step]

    f₈ = plot(x,y,z; label="Spacecraft Position")
    plot!(f₈; title="Numerically Produced Earth-Moon L2 Halo",
            xlabel="X (DU)", ylabel="Y (DU)", zlabel="Z (DU)")

    savefig(f₈, "media/fig8.png")

    f₈

end;
```