

# Low-Thrust Trajectory Optimization with No Initial Guess

By Nathan L. Parrish<sup>1)</sup> and Daniel J. Scheeres<sup>1)</sup>

<sup>1)</sup>Colorado Center for Astrodynamics Research, University of Colorado, Boulder, USA

(Received June 21st, 2017)

A two-stage differential corrector is applied to finding optimal low-thrust trajectories, both with two-body dynamics and with the circular restricted three body problem (CRTBP) dynamics. The first stage of the differential corrector ensures that dynamics are followed, and the second stage finds the least squares solution to minimize the sum of delta-v squared. Then, a homotopic approach is used with indirect multiple shooting to transition the delta-v squared solution to the minimum fuel mass solution, with constrained thrust. The approach is found to be robust to poor initial guesses, even converging when the initial guess consists of points pulled from a random distribution, with the constraint that the endpoints and time of flight must remain fixed. This robustness to initial guess is highly desirable especially for transfers in three-body dynamics, when it can be difficult to find a good initial guess.

**Key Words:** Optimal Control, Electric Propulsion, Multiple Shooting

## Nomenclature

$\vec{r}$	: position vector
$\vec{v}$	: velocity vector
$\vec{a}$	: acceleration vector
$\vec{u}$	: control vector
$\delta\vec{r}$	: position error vector
$t$	: time
$[J]$	: Jacobian matrix
$\mu$	: gravitational parameter
$\lambda$	: Lagrange multiplier
$\vec{\lambda}$	: “dualized” state and costate

## Subscripts

$0$	: initial
$f$	: final
$i$	: index $i$
$u$	: due to control

## Superscripts

$given$	: fixed, given value
$prop$	: value from numerical propagation

## 1. Introduction

Electric propulsion (EP) is an enabling technology for many missions because it allows a greater total change in velocity ( $\Delta v$ ) than chemical propulsion for the same or less propellant mass. EP systems can have exhaust velocities (or equivalently, specific impulse) an order of magnitude higher than chemical systems. However, the tradeoff is that the thrust generated by such systems is much lower. To change orbits with EP, the thrust may need to remain on for days or even months at a time. Chemical maneuvers in many cases can be accurately modeled as single impulsive changes in velocity, whereas low-thrust maneuvers are long-duration continuous thrust arcs. Optimizing these low-thrust transfers is a great deal more complex than optimizing impulsive-burn trajectories. Most optimization tools require a “good” initial guess of the transfer

in order to converge, but such an initial guess is not always available.

A sizable body of work exists in the literature transcribing these optimal trajectory problems into nonlinear programming (NLP) problems.<sup>1-6</sup> The typical approach is to hand the NLP problem to an industry-standard “black-box” NLP solver such as IPOPT<sup>7)</sup> or SNOPT.<sup>8)</sup> Implementations vary mostly in terms of the transcription of the optimal control problem and the enforcement of dynamics constraints.

For many trajectory optimization problems, the experience of the authors is that the vast majority of computational time is spent in the iterative linear algebra of the NLP solver. This work calls the NLP solver into question, using instead a different problem formulation and an ordinary least squares solution. The strategy employed here is to iteratively use a simple least squares solver to find optimal, feasible trajectories given only the endpoints and time of flight.

## 2. The Optimal Control Problem

The optimal control problem is defined as: minimize the Lagrange performance index

$$J = \int_{t_0}^{t_f} L[x(t), u(t), t] dt \quad (1)$$

subject to differential constraints due to the system dynamics

$$\dot{x}(t) = f[x(t), u(t), t], \quad (2)$$

path constraints (such as limiting thrust)

$$h_L \leq h[x(t), u(t), t] \leq h_U, \quad (3)$$

and endpoint constraints (such as constraining the initial and final orbits)

$$e_L \leq e[x(t_0), u(t_0), x(t_f), u(t_f), t_0, t_f] \leq e_U. \quad (4)$$

## 3. Circular Restricted Three Body Problem

The circular restricted three body problem (CRTBP) model assumes that the spacecraft is massless in comparison to the two primaries (here, the Earth and Moon). The primaries orbit their barycenter in a circular orbit.

Dimensionless units are used so that all state variables are on the order of 1. One distance unit (DU) is equal to the mean

distance between Earth and Moon, or 384747.962856037 km. The non-dimensional mass ratio  $\mu$  is equal to 0.012150585609624. A synodic reference frame is used such that the Earth is fixed at the point  $[-\mu, 0, 0]^T$ , and the Moon is fixed at the point  $[1 - \mu, 0, 0]^T$ . This reference frame is illustrated in Fig. 1.

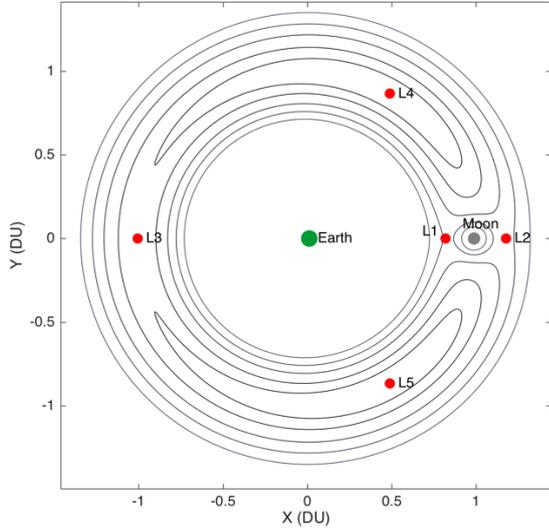


Fig. 1. The Earth-Moon synodic reference frame, with libration points  $L_1$  through  $L_5$  labeled.

The equations of motion for the system in this rotating frame are given by

$$\ddot{x} = -\left(\frac{(1-\mu)}{r_1^3}(x+\mu) + \frac{\mu}{r_2^3}(x-1+\mu)\right) + 2\dot{y} + x + a_{u,x} \quad (5)$$

$$\ddot{y} = -\left(\frac{(1-\mu)}{r_1^3}y + \frac{\mu}{r_2^3}y\right) - 2\dot{x} + y + a_{u,y} \quad (6)$$

$$\ddot{z} = -\left(\frac{(1-\mu)}{r_1^3}z + \frac{\mu}{r_2^3}z\right) + a_{u,z}. \quad (7)$$

The use of simplified dynamical models here is justified because the algorithm's effectiveness would not be affected by higher-fidelity perturbations.

#### 4. Direct vs. Indirect Methods

The main difference between direct and indirect methods is in how the control is parameterized. For direct methods, the control is specified as one or many optimization variables. For indirect methods, the control is chosen to satisfy the first-order optimality conditions derived from Pontryagin's minimum principle.

There are two objective functions that we are interested in. Both are related to the integrated control effort:

$$cost = \int L dt = \int |\vec{u}|^p dt. \quad (8)$$

Ultimately, we want to minimize the propellant mass used, for which  $p = 1$ . As an intermediate step, we will minimize the cost according to the  $p = 2$  problem. The  $p = 2$  solution is sometimes referred to as the "minimum energy" solution, but that terminology is not accurate here.

We will derive the control law for two-body dynamics. For other dynamics, the control law has the same relationship with the primer vector  $\vec{\lambda}_v$ , but the dynamics of the costates change accordingly. For two-body dynamics, the Hamiltonian is given by

$$H = |\vec{a}_u|^p + \vec{\lambda}^T \left( \begin{bmatrix} \vec{v} \\ -\frac{\mu}{r^3}\vec{r} \end{bmatrix} + \begin{bmatrix} \vec{0}_3 \\ \vec{a}_u \end{bmatrix} \right), \quad (9)$$

which can be simplified to

$$H = \vec{\lambda}^T \begin{bmatrix} \vec{v} \\ -\frac{\mu}{r^3}\vec{r} \end{bmatrix} + |\vec{a}_u|(|\vec{a}_u|^{p-1} + \vec{\lambda}_v^T \hat{a}_u). \quad (10)$$

From Pontryagin's minimum principle, we know that an optimal control law minimizes the Hamiltonian. Differentiating Eq. (10) with respect to control and setting equal to zero yields

$$p|\vec{a}_u|^{p-1} + \vec{\lambda}_v \cdot \hat{a}_u = 0. \quad (11)$$

Looking back at Eq. (10), we can see from inspection that to minimize  $H$ , we should always choose the control direction  $\hat{a}_u = -\hat{\lambda}_v$ . The magnitude of control acceleration is found by solving Eq. (11) for  $|\vec{a}_u|$ . Now the complete control law as a function of the primer vector  $\vec{\lambda}_v$  is

$$\vec{a}_u = -\left(\frac{1}{p}|\lambda_v|\right)^{\frac{1}{p-1}} \hat{\lambda}_v \quad (12)$$

for  $1 < p \leq 2$ . If  $p$  is exactly equal to 1, we make a modification to the control law by inspection of Eq. (10).

$$\vec{a}_u = \begin{cases} \vec{0}_3 & \text{if } \lambda_v < 1 \\ -u_{max}\hat{\lambda}_v & \text{if } \lambda_v > 1 \\ \text{indeterminate} & \text{if } \lambda_v = 1 \end{cases} \quad (13)$$

where  $u_{max}$  is the maximum possible control magnitude. Results in the literature<sup>9-11</sup> and the authors' own experience find that the radius of convergence is much larger when  $p = 2$  than when  $p = 1$ . Thus, a helpful strategy to solve optimal control problems is to start with the  $p = 2$  solution, then gradually reduce  $p$  down to 1, with several intermediate values in between. At each step, the converged solution from the prior step is used as the initial guess.

The evolution of the Lagrange multipliers is found with Pontryagin's minimum principle.

$$\dot{\vec{\lambda}} = -\frac{\partial H}{\partial \vec{x}} \quad (14)$$

where  $\vec{x}$  is the state vector (here, position and velocity). For brevity, the derivatives are not written here.

#### 5. Multiple Shooting

We use multiple shooting to transcribe the optimal control problem as a nonlinear programming (NLP) problem. In the direct formulation, the optimization variables traditionally consist of the state and control at each node.

As implemented here, the direct formulation has a two-stage differential corrector. The first stage chooses the control to perfectly follow the dynamics, and the second stage chooses the states to minimize the required control effort. This is described in greater detail in Section 6.

Indirect multiple shooting is very similar to direct multiple shooting in general, with the main difference being that control is computed according to the control law derived, instead of being specified explicitly as optimization variables.

## 6. Solution Method

In this work, we use the simplest optimization strategy possible: an ordinary least squares solver. We begin by solving a problem with a large radius of convergence (the  $p = 2$  problem with direct method) and gradually modify the solution until we have the more interesting problem, which has a small radius of convergence (the  $p = 1$  problem with indirect method). By solving progressively harder problems, the optimization is much more efficient than traditional implementations involving NLP solvers.

The direct method is set up to have an ‘‘inner loop’’ and an ‘‘outer loop’’. The inner loop enforces the dynamics constraints, and the outer loop minimizes the cost.

This inner/outer loop formulation can be described in the language of a two-stage differential corrector. This technique is commonly used to find three-body periodic orbits in high-fidelity dynamics simulations.<sup>12)</sup> The traditional implementation does not allow thrust, so the velocity and position are required to match at each node. In the present work, we do have some limited thrust available. Position is required to match, and the discontinuity in velocity represents a small impulsive maneuver. If these maneuvers are small enough and close enough in time, a trajectory modeled in this way can approximate a continuous-thrust maneuver. The Sims-Flanagan transcription, famous for its implementation in the software tool MALTO,<sup>13)</sup> is a common example of approximating a continuous-thrust maneuver as a series of small impulsive maneuvers.

In the inner loop, a shooting method is implemented to find the velocities at each endpoint, given the positions and the time of flight between them. For two-body point-mass dynamics, this is equivalent to solving Lambert’s problem, albeit less computationally efficient. For any arbitrary force model, a simple algorithm is implemented as follows.

For the initial guess, assume that the trajectory follows a straight line between  $\vec{r}_i$  and  $\vec{r}_{i+1}$ . Then, the velocity guess at the first node is:

$$\vec{v}_i^{guess} = \frac{(\vec{r}_{i+1}^{given} - \vec{r}_i^{given})}{t_{i+1} - t_i}. \quad (15)$$

The state defined by  $\vec{r}_i, \vec{v}_i$  is propagated via numerical integration from time  $t_i$  to time  $t_{i+1}$ , using a fixed-step integrator. A fixed-step integrator is chosen over an adaptive integrator for two reasons. First, if an intermediate solution comes near a singularity, a variable step integrator will become very slow. Although a fixed-step integrator is inaccurate near a singularity, we generally want to avoid flying a spacecraft too close to any massive body anyway. Once a low-fidelity solution has been found, we can simply add more nodes where needed to meet integration accuracy requirements. The other reason to use a fixed-step integrator is that the finite-differenced partial derivatives are more consistent. It was found that adaptive-step integration methods add numerical noise to the approximate derivatives, which significantly impairs convergence.

The position error to be removed is then given by

$$\delta\vec{r}_{i+1} = \vec{r}_{i+1}^{given} - \vec{r}_{i+1}^{prop}. \quad (16)$$

Finite differencing is used to compute the Jacobian  $[J_{inner}]$  of position error  $\delta\vec{r}_{i+1}$  with respect to the initial velocity  $\vec{v}_i$ .

The least squares update to velocity is then

$$\vec{v}_i = \vec{v}_i^{guess} - [J_{inner}]^{-1} \delta\vec{r}_{i+1}. \quad (17)$$

Once the velocity departing and arriving at each node has been found in the inner loop, the impulsive  $\Delta\vec{v}_i$  is simply the difference between the velocity going into node  $i$  and the velocity leaving node  $i$ . The outer loop minimizes the sum of the squared  $\Delta\vec{v}_i \forall i$ .

For the outer loop, finite differencing is again used to construct the Jacobian matrix. Now, the Jacobian is the partial derivative of each  $\Delta\vec{v}_i$  element with respect to each position  $\vec{r}_i$  element.

$$[J_{outer}] = \frac{\partial(\Delta\vec{v}_i)}{\partial\vec{r}_j}. \quad (18)$$

In this work, the initial and final position vectors are held fixed, and time of flight is also fixed. The outer loop Jacobian is a sparse matrix; each position vector  $\vec{r}_i$  influences three  $\Delta\vec{v}$  vectors:  $\Delta\vec{v}_{i-1}$ ,  $\Delta\vec{v}_i$ , and  $\Delta\vec{v}_{i+1}$ . The size of the Jacobian is  $3N$  rows (corresponding to the  $\Delta\vec{v}$ ) by  $3(N - 2)$  columns (corresponding to the  $\vec{r}$ , without the endpoints).

After converging on a solution to the  $p = 2$  problem with direct multiple shooting, homotopy with indirect multiple shooting is used to find the nearby solution to the  $p = 1$  problem. As implemented here, indirect multiple shooting is a single stage (as opposed to the two stages used for direct multiple shooting). At every node, the state and costate are defined as optimization variables. The ‘‘dualized’’ state is constructed from its parts.

$$\vec{X}_i = [\vec{r}_i^T, \vec{v}_i^T, \vec{\lambda}_{r,i}^T, \vec{\lambda}_{v,i}^T]^T \quad (19)$$

The objective with the indirect multiple shooting step is to simultaneously enforce the dynamics for the states and costates. The dualized state  $\vec{X}_i$  is propagated to time  $t_{i+1}$ , giving us  $\vec{X}_{i+1}^{prop}$ . The error in dynamics is then

$$\vec{\epsilon}_{i+1} = \vec{X}_{i+1} - \vec{X}_{i+1}^{prop}. \quad (20)$$

Using finite differences, the Jacobian of all the  $\vec{\epsilon}_i$  terms with respect to all the dualized states  $\vec{X}_i$  is computed. Ordinary least squares is then used iteratively to bring all  $\vec{\epsilon}_i$  below some small tolerance.

## 7. Examples & Results

The two standout features of this work are that viable solutions for many problems are found with an extremely poor initial guess, and that specialized optimization packages are not required. The following two examples will demonstrate this.

For the purposes of this paper, mass is assumed to be a constant 1000 kg. While we acknowledge this is not accurate, we note that trajectories found with this assumption are still valid – modeling the mass loss from propellant consumed would simply give the spacecraft greater control authority as time goes on.

### 7.1. Earth-Mars low-thrust rendezvous

A simple example is to design the trajectory followed by a spacecraft with electric propulsion to transfer from Earth to Mars. Two-body dynamics are used, with the Sun as the sole gravitational attractor. The trajectory is assumed to start at Earth with zero relative velocity and end at Mars with zero

relative velocity. Earth departure state is at 21 July 2020, and Mars arrival state is at 14 September 2022. For problems like this with simple dynamics, it is relatively straightforward to design a “good” initial guess intuitively. However, for the purpose of demonstrating the flexibility of the algorithm, the worst possible initial guess was used – each element of each position vector was drawn from a random normal distribution.

In the indirect formulation, the thrust available (and maximum control acceleration) is assumed to be constant. In reality, the acceleration available varies as a function of distance from the Sun, solar panel degradation, spacecraft mass, and other factors. These are neglected here because they are not necessary to demonstrate the algorithm.

### 7.1.1 Optimization of p=2 problem for Earth-Mars

The first step in optimizing is to use the direct multiple shooting method described in Section 6. It was found that this approach generally can converge from a random initial guess with about 10-15 iterations. In some cases, the algorithm can converge on solutions that are clearly suboptimal – for instance, becoming retrograde for a portion of the orbit. These suboptimal alternative families of solutions exist for most orbital transfers and are local optima of the solution space which satisfy the first-order optimality conditions. To mitigate the risk of getting “stuck” in a poor local optimum, several different random initial guesses can be used, selecting the best result afterward. If a decent initial guess is available, the iterative method will be guided to a better local optimum and converge more quickly.

Figures 1-4 show the progress of the direct multiple shooting method from a random initial guess to a converged solution in 12 iterations. The view is of the solar system from “above” – the +Z axis of the ecliptic plane.

The computation time is largely driven by the cost to numerically integrate the trajectory. Since each multiple-shooting leg is independent of the others, these can be efficiently parallelized. In addition, since a fixed-step integrator is used, we know that the computational effort of each leg is identical.

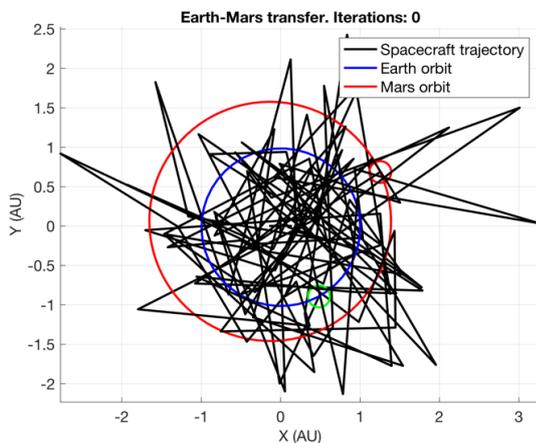


Fig. 2. The initial guess is completely random, with points drawn from a normal distribution roughly the same scale as the problem.

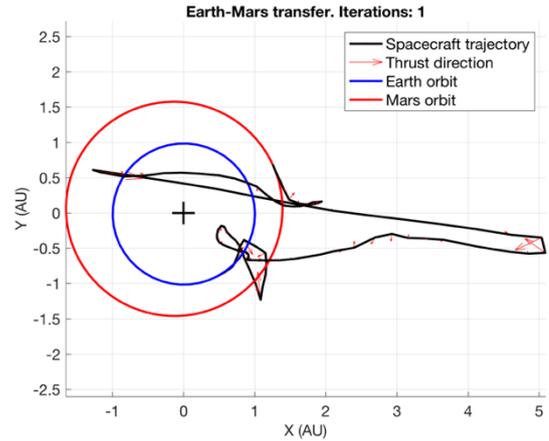


Figure 3. After a single iteration of the direct multiple shooting algorithm with least squares, some structure is apparent.

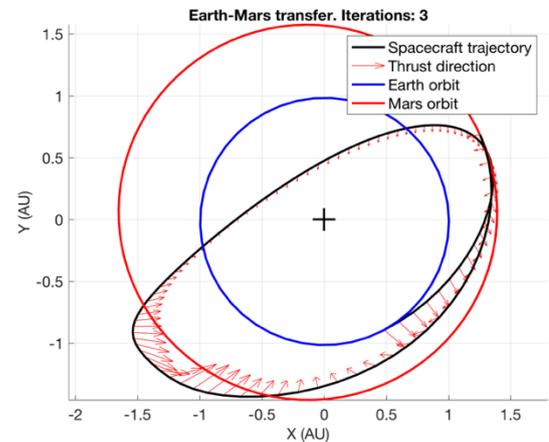


Figure 4. After 3 iterations, the current iteration is clearly in the family of the final solution.

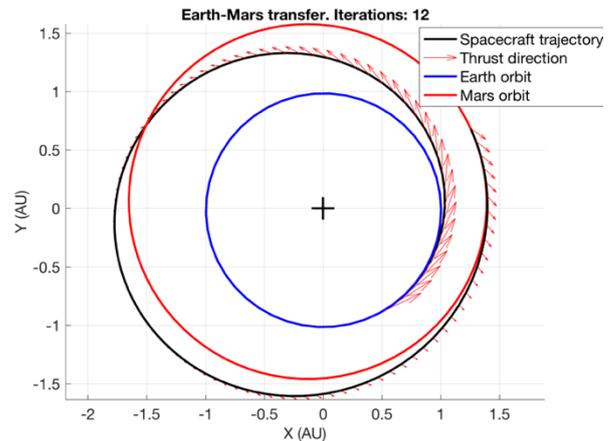


Figure 5. After 12 iterations, the problem has fully converged.

In the current implementation, a parallel for-loop is used in MATLAB, which takes about 1-2 seconds per iteration (depending on the number of legs used). Future implementations could easily distribute the propagation effort on a GPU (graphics processing unit), which we expect would

be 1-2 orders of magnitude faster than the MATLAB implementation.

### 7.1.1. Homotopy to $p = 1$ problem for Earth-Mars

When the first step described above converges, it has found an optimum state and control history for the  $p = 2$  problem, with the continuous-thrust trajectory approximated as a series of small impulsive maneuvers. We now feed this solution into an indirect multiple shooting algorithm to improve the quality of the solution. By using an indirect method, we can naturally describe the trajectory with truly continuous thrust. The thrust limit is also naturally introduced by limiting the control law output. Finally, the indirect implementation allows us to change from the  $p = 2$  solution (minimum “energy”) to the  $p = 1$  solution (minimum propellant used).

The costates  $\vec{\lambda}_i$  at each node can simply be estimated by back-solving the control law for the costates. We have the discretized control history from the direct method. Assuming the direct method did fully converge, a good estimate for the costates is given by:

$$\vec{\lambda}_{v,i} = -p\vec{a}_{u,i} \quad (21)$$

$$\vec{\lambda}_{r,i} = \frac{(\vec{\lambda}_{v,i+1} - \vec{\lambda}_{v,i})}{t_{i+1} - t_i} \quad (22)$$

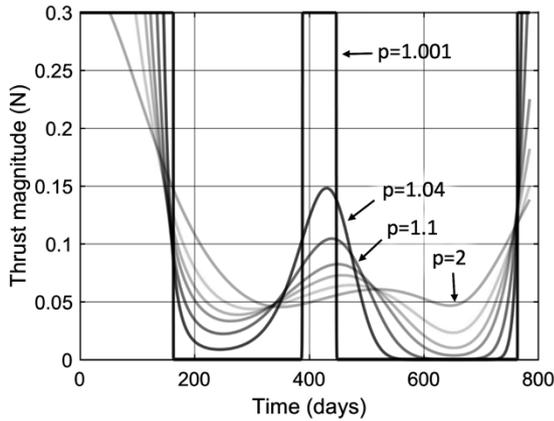


Fig. 6. Evolution of optimal thrust profiles as the cost function is changed from the  $p = 2$  problem to the  $p = 1$  problem, for the Earth-Mars transfer problem. As  $p$  is reduced, the thrust on/off times become steeper, until becoming perfectly vertical at 1. Note that the algorithm could not converge for  $p$  exactly equal to 1, but it can reach arbitrarily close to 1.

Thrust is limited to 0.3 N, which corresponds to acceleration of  $3 \times 10^{-4} \text{ m/s}^2$ . As a reference point, the Dawn spacecraft had a maximum acceleration at beginning of life of  $2.3 \times 10^{-4} \text{ m/s}^2$ .<sup>14)</sup>

## 7.2. Earth-Moon DRO to $L_2$ halo transfer

This example uses the CRTBP dynamics to transfer between two 3-body orbits: a distant retrograde orbit (DRO) about the Moon to a halo orbit about Earth-Moon  $L_2$ . The DRO is defined such that when crossing the x-axis in the +y direction, the x-component of position is 0.9 DU, and all motion is in the xy-plane. The DRO has a period of approximately 5.55 days and the Jacobi constant is approximately 3.0251. The halo orbit has a period of approximately 14.02 days and a Jacobi constant of approximately 3.0803.

### 7.2.1. Optimization of $p = 2$ problem for DRO to $L_2$

As with the Earth-Mars transfer problem, the first step is to optimize the  $p = 2$  problem using direct multiple shooting. A random initial guess was used to demonstrate the robustness of the algorithm. Figures 7-10 show snapshots of the optimization progress from the initial guess to the converged solution. These figures are generated in the synodic reference frame, with dimensionless units. Both bodies are plotted to scale. The time of flight used in this example is 20 days.

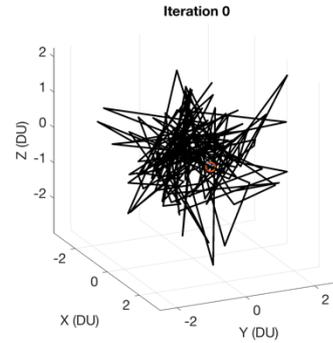


Fig. 7. The initial guess is random noise.

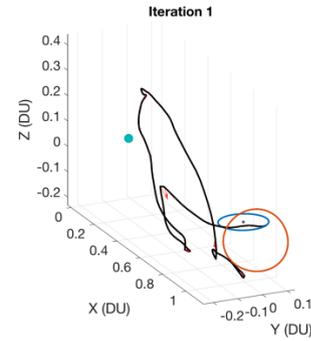


Figure 8. After a single iteration, some structure is apparent.

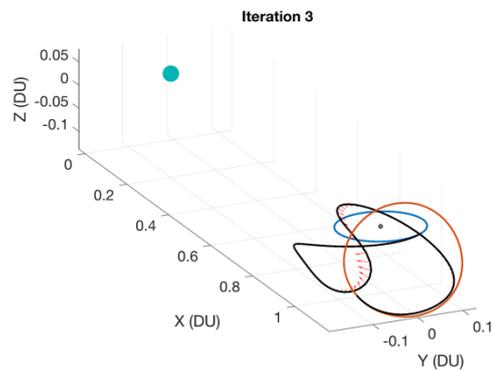


Figure 9. After 3 iterations, there is a smooth trajectory. The path is not optimal yet, though.

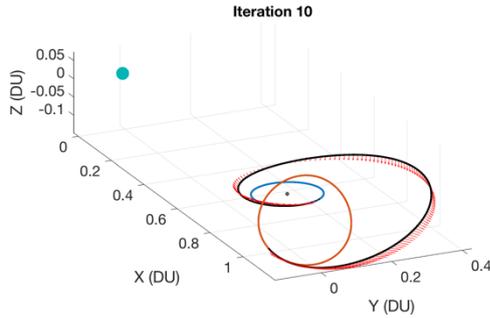


Figure 10. The solution converged in 10 iterations in this case.

As with the Earth-Mars transfer, this method is extremely robust when the endpoints and time of flight are held fixed. The rapid convergence rate lends itself well to parametric design studies, where many solutions to similar trajectories must be found. As an example, Fig. 11 shows departing the DRO and arriving on the  $L_2$  halo orbit at the same positions, but with different times of flight. This could be helpful to determine phasing maneuvers in 3-body dynamics, or to understand the trade space better.

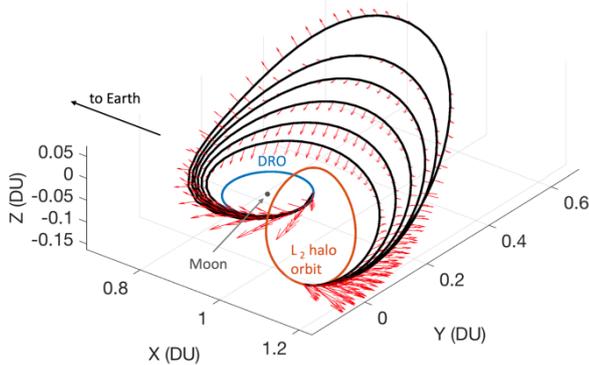


Fig. 11. An example of six trajectories from the same family with different times of flight. Time of flight here varies from 15 days to 40 days.

### 7.2.1. Homotopy to $p = 1$ problem for DRO to $L_2$

Again, we can use a homotopy approach to gradually move from the  $p = 2$  “minimum energy” solution to the  $p = 1$  minimum fuel solution, with indirect multiple shooting. It was found that between 10-20 intermediate values of  $p$  were necessary for convergence. Each intermediate value will either converge or diverge within 5-10 iterations.

## 8. Conclusion

We show that the new approaches presented compare favorably to existing methods. In comparison to the traditional approach of plugging the problem into a “black-box” NLP solver, the methods shown converge even when given no knowledge of the solution at all. This robustness to initial guess is a compelling feature, as three-body orbit transfers are challenging to design with intuition alone. Of course, if a high-quality initial guess is available, the methods shown are still valid.

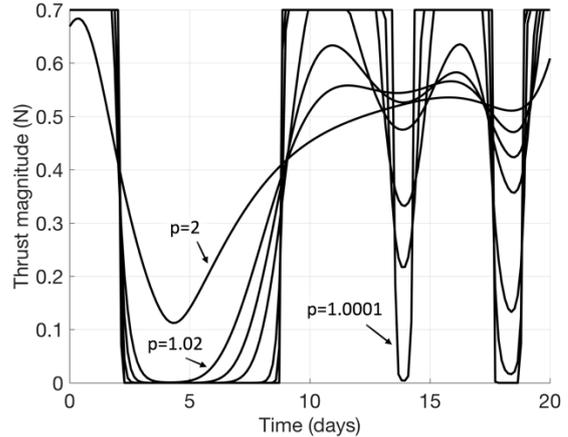


Fig. 12. Evolution of optimal thrust profiles as the cost function is changed from the  $p = 2$  problem to the  $p = 1$  problem, for the DRO- $L_2$  transfer problem.

The major limitation of this least-squares approach is that the endpoints and time of flight must remain fixed, or the problem will diverge. Future work will explore how the endpoint constraints can be opened up while still permitting convergence.

The other limitation is that the random initial guess strategy tends to lead to solutions with at most two orbital revolutions. While that is satisfactory for many missions, spacecraft with less capable propulsion systems may need to spiral for several revolutions to complete a transfer. To meet tighter constraints on thrust, a more intelligent initial guess can be used which includes more revolutions.

## Acknowledgments

This work was supported by a NASA Space Technology Research Fellowship.

## References

- 1) Sims, J., Finlayson, P., Rinderle, E., Vavrina, M., and Kowalkowski, T., “Implementation of a Low-Thrust Trajectory Optimization Algorithm for Preliminary Design,” *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2006, pp. 1–10.
- 2) Herman, J. F. C., *Improved Collocation Methods to Optimize Low-Thrust, Low-Energy Transfers in the Earth-Moon System*, 2015.
- 3) Mingotti, G., Heiligers, J., and McInnes, C., “Optimal Solar Sail Interplanetary Heteroclinic Transfers for Novel Space Applications,” *AIAA/AAS Astrodynamics Specialist Conference*, 2014, pp. 1–26.
- 4) Mingotti, G., Toppo, F., and Bernelli-Zazzera, F., “Numerical Methods to Design Low-Energy, Low-Thrust Sun-Perturbed Transfers to the Moon,” *49th Israel Annual Conference on Aerospace Sciences*, 2009.
- 5) Betts, J. T., “Very Low Thrust Trajectory Optimization Using a Direct SQP Method,” *Journal of Computational and Applied Mathematics*, vol. 120, 2000, pp. 27–40.
- 6) Betts, J. T., “Using Direct Transcription to Compute Optimal Low Thrust Transfers Between Libration Point Orbits Runge-Kutta Methods,” 2016, pp. 1–23.
- 7) Wachter, A., and Biegler, L. T., “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical Programming*, vol. 106, 2006, pp. 25–57.
- 8) Gill, P. E., Murray, W., and Saunders, M. A., “SNOPT: An SQP

- Algorithm for Large-Scale Constrained Optimization,” *SIAM Journal on Optimization*, vol. 12, 2002, pp. 979–1006.
- 9) Haberkorn, T., Martinon, P., and Gergaud, J., “Low thrust minimum fuel orbital transfer: a homotopic approach,” *Journal of Guidance, Control, and Dynamics*.
  - 10) Guo, T., Jiang, F., and Li, J., “Homotopic approach and pseudospectral method applied jointly to low thrust trajectory optimization,” *Acta Astronautica*, vol. 71, 2012, pp. 38–50.
  - 11) Haberkorn, T., Martinon, P., and Gergaud, J., “Low Thrust Minimum-Fuel Orbital Transfer: A Homotopic Approach,” *Journal of Guidance, Control, and Dynamics*, vol. 27, 2004, pp. 1046–1060.
  - 12) Parker, J. S., and Anderson, R. L., “Low-Energy Lunar Trajectory Design,” 2013.
  - 13) Sims, J. A., and Flanagan, S. N., “Preliminary design of low-thrust interplanetary missions,” *Jet Propulsion*.
  - 14) “NASA Space Science Data Coordinated Archive” Available: <https://nssdc.gsfc.nasa.gov/nmc/spacecraftDisplay.do?id=2007-043A>.