PARTICLE SWARM OPTIMIZATION OF TWO-MANEUVER, IMPULSIVE TRANSFERS FROM LEO TO LAGRANGE POINT ORBITS VIA SHOOTING

Andrew J. Abraham(1), David B. Spencer(2), and Terry J. Hart(3)
(1)Lehigh University, 19 Memorial Dr. W., Bethlehem, PA, 18015, 484-860-2786, aja208@lehigh.edu. (2)Penn State University, 229 Hammond Building, University Park, PA, 16802, 814-865-4537, dbs9@psu.edu. (3)Lehigh University, 19 Memorial Dr. W., Bethlehem, PA, 18015, 610-758-4173 , teh305@lehigh.edu.

Abstract: *A two-maneuver, impulsive transfer from Low Earth Orbit (LEO) to a Lagrange Point Orbit (LPO) is optimized using Particle Swarm Optimization (PSO) and single shooting. The first maneuver departs LEO while the second enters the invariant stable manifold of the target LPO. Optimization with respect to total ΔV as well as LEO inclination is studied. It was discovered that the optimal manifold insertion location for slow transfers represents an apogee condition along the manifold. Fast transfers are also investigated and have been shown to require a lower ΔV than slow transfers for the candidate LPO used in this study.*

Keywords: *Particle Swarm Optimization, Lagrange Point Orbits, Single Shooting, Trajectory Optimization.*

1 Introduction

The study of Lagrange Point Orbits (LPO) in the Earth-Moon system is gaining popularity within the astrodynamics community. Farquahr [7, 9] noted that halo orbits could be used for communication relays as early as the late 1960's while Heppenhiemier [10] envisioned early space colonization of Lagrange points in 1978. Recent studies have focused on human operations [11, 8], observation and surveillance [6], and telerobotics. The ARTEMIS mission even orbited the Earth-Moon L_1 and L_2 Lagrange points to study Earth's magnetotail in great detail [4, 16]. Despite the interest in utilizing LPO's only a modest amount of research exists in optimizing transfers from geocentric orbits to LPO's. These transfers come in two varieties: low-thrust and high-thrust. Unfortunately, the global optimization of LPO transfers is extremely challenging due to the complexity of three body dynamics. Nevertheless, attempts of global optimization (which was assumed, but not proven) have been made for the low-thrust case by Abraham et al. [1, 2] utilizing manifold theory and evolutionary algorithms.

This present work is intended to extend Abraham's prior work to the realm of high-thrust transfers. Specifically, this study focuses on the optimization of two-maneuver transfers from Low Earth Orbit (LEO) to a desired LPO via an evolutionary algorithm known as Particle Swarm Optimization (PSO). This study will outline a method to merge direct shooting with PSO in a manner that can quickly sample the global search space and identify near optimal transfer conditions. This hybrid

PSO/Shooting approach attempts to utilize the strength of each parent technique and minimize their respective weaknesses. This is accomplished by optimizing individual “manifold insertion points” using the gradient-based, single shooting technique which is very fast and accurate. In addition to the gradient-based algorithm, an evolutionary PSO algorithm was used to identify the best “manifold insertion point” based on an objective function that uses the results of the shooting algorithm as its input. The PSO method does not require gradient information and is very adept at working with non-convex objective functions with a large number of extrema; especially the “local” PSO version. Furthermore, the PSO algorithm does not require *a priori* knowledge of the shape of the objective function. This is highly advantageous when an analytic expression of the objective function is unknown and sapling the function requires intense computational resources.

2 Problem Statement

The spacecraft begins in a LEO parking orbit that is roughly 400 km in altitude. From this orbit, a departure burn is preformed that places the spacecraft on a cislunar trajectory that will either directly cross paths with the LPO or will cross the path of the LPO’s invariant stable manifold. A second burn is performed to insert the spacecraft onto the LPO or stable manifold. In the case of manifold insertion, the spacecraft will ballistically coast along this manifold until it is automatically delivered to the target LPO.

To optimize this transfer one must first define a search space for a given LPO. In this study the search space is defined as the set of all states that comprise the invariant stable manifold of the target LPO. If an arbitrary state, $\mathbf{X}_{s.m.}$ on this manifold is selected one may use the technique of single shooting to compute a feasible, two-maneuver trajectory between the LEO and $\mathbf{X}_{s.m.}$. Indeed, this shooting algorithm represents the first stage of optimization but is insufficient to globally optimize the problem. The result of this shooting is fed into a “fitness function” which characterizes the performance of the point $\mathbf{X}_{s.m.}$. The process can then be repeated numerous times for other values of $\mathbf{X}_{s.m.}$ with the hope that after a number of points have been evaluated, the optimal point, $\mathbf{X}^*_{\text{s.m.}}$, can be identified. Of course, the method used to select values of $\mathbf{X}_{s.m.}$ from an infinite set of possibilities is of paramount importance. While it is possible to randomly select a state from the manifold, this technique represents a very crude optimization approach. Instead the method of PSO is used to inject an element of intelligence into an otherwise random process. Swarm intelligence has been successfully applied to spacecraft trajectory optimization problems in the past [2, 1, 5] and has proven to be very effective when optimizing non-convex fitness functions. The following subsections describe each element of the two-maneuver, impulsive optimization algorithm in greater detail.

2.1 System Dynamics

In this study the Circular Restricted Three Body Problem (CR3BP) is used to model the motion of the spacecraft under the simultaneous gravitational acceleration of two massive primaries. Referring to Figure 1, which uses a synodic reference frame with its origin located at the barycenter of the two primaries, the equations of motion can be expressed as

$$\begin{aligned} \ddot{x} - 2\dot{y} &= \Omega_x, \\ \ddot{y} + 2\dot{x} &= \Omega_y, \\ \ddot{z} &= \Omega_z \end{aligned} \tag{1}$$

with the subscripts of the pseudopotential

$$\Omega(x,y,z) = \frac{1}{2}\left(x^2+y^2\right) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} \tag{2}$$

indicating partial derivatives with respect to the sub-scripted variable. The distances $r_1 = \sqrt{(x+\mu)^2+y^2+z^2}$ and $r_2 = \sqrt{(x-(1-\mu))^2+y^2+z^2}$ represent the normalized distance between the spacecraft and the smaller primary (Moon) and larger primary (Earth), respectively. Note that Eqs. 1 are written in dimensionless units, that is to say that: the angular velocity of the larger and smaller primary (ω), the sum of the masses of each primary, and the distance between each primary are all intentionally set to unity. The unit of distance, [du], is the distance from the larger primary to the smaller primary, while the unit of time, [tu], is equal to the synodic period of the primaries divided by 2π . In the Earth-Moon system 1 [du] $= 3.844 \times 10^5$ [km] and 1 [tu] $= 4.348377$ [days]. The velocity unit, [vu], is simply equal to [du]/[tu]. The mass parameter is defined as

$$\mu = \frac{m_2}{m_1+m_2} \tag{3}$$

which represents the ratio of the mass of the smaller primary divided by the total mass of the system. In the case of the Earth-Moon system, this mass parameter is $\mu = 0.0121506683$.

Under ballistic conditions, exactly one constant of motion can be defined [17] for Eq. 1

$$C(x,y,z,\dot{x},\dot{y},\dot{z}) = 2\Omega(x,y,z) - \left(\dot{x}^2+\dot{y}^2+\dot{z}^2\right) = \text{constant}, \tag{4}$$

and is known as the Jacobi constant. For a given spacecraft state, $\mathbf{X} = [x,y,z,\dot{x},\dot{y},\dot{z}]^T$, there exists exactly one Jacobi constant in the CR3BP. It is also possible for the spacecraft to locate itself at an equilibrium point of Eq. 1. Five equilibrium points, known as Lagrange points, exist; with three collinear points (L_1, L_2, and L_3) and two triangular points (L_4 and L_5) . It is possible to define complex, periodic trajectories in the vicinity of each collinear Lagrange point. These periodic trajectories are known as two-dimensional Lyapunov orbits, three-dimensional halo orbits, and quasi-periodic Lissajous trajectories.

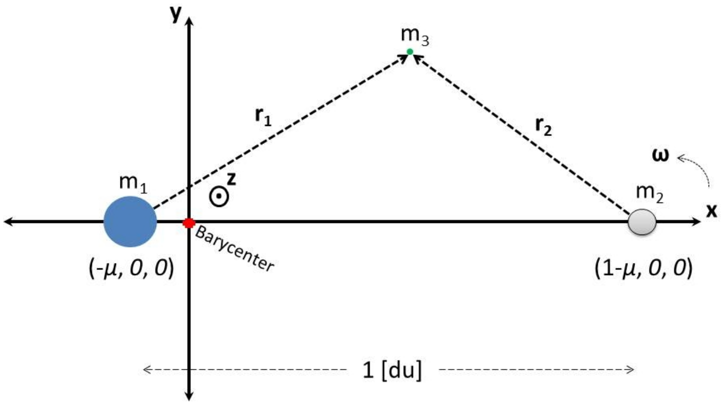

Figure 1: Coordinate System of CR3BP

2.2 Invariant Stable Manifolds

A Lagrange point orbit can be identified either via tabulated values [17] or a numeric computation that solves a two-point boundary value problem [12, 5]. The Lagrange point orbit is then defined by an arbitrary state vector (anywhere along the orbit), $\mathbf{X_{orbit}}$, and the period of the orbit, P. Under CR3BP dynamics, the state vector for the spacecraft is expressed in terms of position and velocity as

$$\mathbf{X} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{5}$$

with the controlled CR3BP expressed in terms of a first-order differential equation as

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ 2v_y + \Omega_x \\ -2v_x + \Omega_y \\ \Omega_z \end{bmatrix}. \tag{6}$$

According to Dynamical Systems Theory (DST) [13], numeric integration of Eq. 1 as well as the State Transition Matrix (STM) $\Phi_{(t,t_0)} = \frac{\partial \mathbf{X}(\mathbf{X_0},t)}{\partial \mathbf{Xo}}$, for one period, P, with the initial condition $\mathbf{X_0} = \mathbf{X_{orbit}}$ enables the calculation of the monodromy matrix, $M = \Phi_{(t=P,t_0)} = \frac{\partial \mathbf{X}(\mathbf{X_0},t=P)}{\partial \mathbf{Xo}}$ associated with the initial state, $\mathbf{X_{orbit}}$. The stable eigenvector, v, of the monodromy matrix is multiplied by a very small number, ε, where $\varepsilon = 10^{-10}$ in this study. A perturbed initial state, $X_{\mathbf{pert}} = X_{\mathbf{orbit}} \pm \varepsilon v$, is then integrated backward in time using Eq. 1. As in Abraham et al., [1, 2] the integration is terminated when the spacecraft crosses the yz-plane, from the negative x-direction. This defines a trajectory that is a member of the stable manifold of the nominal Lagrange point orbit. The nominal Lagrange point orbit can be discretized into N states defined as $\mathbf{X^{(k)}_{orbit}}$ with $k \in [1, N]$. The process is then repeated for other values of $\mathbf{X^{(k)}_{orbit}}$ which, in turn, constructs a stable manifold of N trajectories. If a spacecraft's state lies along a trajectory within this manifold then the ballistic flow forward in time will take it to the state $\mathbf{X^{(k)}_{orbit}}$ and the spacecraft will be automatically inserted into the nominal Lagrange point orbit. In this way, any state within the manifold, $\mathbf{X_{s.m.}}(\tau_{01}, k)$, can be expressed via two parameters:

1. An integer $k \in [1, N]$ that corresponds to a state on the nominal orbit $\mathbf{X^{(k)}_{orbit}}$ with N being the total number of states that represent a discretization of the orbit.

2. A time parameter τ that represents the time remaining for a ballistic flow of Eq. 1 to reach the state $\mathbf{X^{(k)}_{orbit}}$.

2.3 Single Shooting a Two-Manuver, Impulsive Transfer

For a given point on the manifold a variable time, single shooting algorithm was used to compute the magnitude and direction of two burns: one to exit the LEO and a second to enter the manifold (or LPO). The shooting was conducted in the following manner. The free variable vector

$$\mathbf{Y} = \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} \tag{7}$$

is defined as the pre-burn velocity components of the insertion point into the manifold/LPO. The magnitude of the velocity discontinuity between Y and the velocity components of $\mathbf{X}_{s.m.}$ yield Δv_{LPO}, or the amount of Δv required to insert the spacecraft into the manifold/LPO. Note that the units of Y are [vu] but can easily be converted to [km/s]. The constraint vector is defined as

$$\mathbf{F} = \begin{bmatrix} \|r - r_d\| \\ \mathbf{r} \cdot \mathbf{V} \end{bmatrix} \tag{8}$$

where r is the magnitude of the altitude of the spacecraft, $r_d = 400$ [km] is the desired altitude of the spacecraft, and $\mathbf{V}$ is the spacecraft's post-burn velocity relative to the center of the Earth. All values of $\mathbf{F}$ are geocentric and expressed in [km] and [km/s]. When $\|r - r_d\| = 0$ the spacecraft is located at the desired altitude corresponding to the target LEO. When $\mathbf{r} \cdot \mathbf{V} = \mathbf{0}$ the spacecraft's position and velocity vectors are perpendicular to each other. This is the case during apogee, perigee, or a circular orbit if only considering the two-body dynamics that dominate LEO. Since a circular parking LEO is desired, it is easy to calculate the necessary ΔV required to match $\mathbf{V}$. The two-body expression for the speed of a circular orbit is given as $V_{LEO} = \sqrt{\frac{GM}{r}}$ where GM is the gravitational parameter of the Earth and r is the known distance from the center of the Earth during the first burn. Since the spacecraft's orbital energy is maximized when the burn is performed perpendicular to the velocity vector a simple subtraction of $\|\mathbf{V}\| - V_{LEO} = \Delta V_{LEO}$ gives the ΔV necessary to exit the LEO and head towards the manifold insertion point. Other characteristics of the LEO can be verified from the state $\mathbf{X}_{LEO} = [r, V_{LEO}]^T$ such as the Keplarian orbital elements; especially eccentricity and inclination.

Given a point on the manifold and the value of $\mathbf{Y}$, one can integrate the spacecraft's path, backward in time, using Eq. 1. This integration will terminate at the perigee of the trajectory and yield the values that define $\mathbf{F}$ at the end of the trajectory. Of course the value of $\mathbf{F}$ is unlikely to be equal to the null vector even if a reasonable guess of $\mathbf{Y}$ is applied. An iterative process (Newton's method) is applied here to guide the single shooting algorithm to the correct value of $\mathbf{Y}$ that results in $\mathbf{F} = \mathbf{0}$. To use this method the Jacobian matrix must be defined

$$D\mathbf{F} = \begin{bmatrix} \frac{\partial}{\partial V_x}\|r - r_d\|, & \frac{\partial}{\partial V_y}\|r - r_d\|, & \frac{\partial}{\partial V_z}\|r - r_d\| \\ \frac{\partial}{\partial V_x}\mathbf{r} \cdot V, & \frac{\partial}{\partial V_y}\mathbf{r} \cdot V, & \frac{\partial}{\partial V_z}\mathbf{r} \cdot \mathbf{V} \end{bmatrix}. \tag{9}$$

This matrix can then be used with Newton's method to iteratively solve

$$Y_{\mathbf{new}} = \mathbf{Y} - D\mathbf{F}^T \left(D\mathbf{F} D\mathbf{F}^T\right)^{-1} \mathbf{F} \tag{10}$$

and will terminate when $\|\mathbf{F}\| \leq \varepsilon$ where $\varepsilon = 10^{-10}$ in this study (or any sufficiently small number). Typically, ten iterations or less are needed to converge on the appropriate solution however a maximum of 50 iterations are attempted before the algorithm gives up and is unable to converge during single shooting.

In order for this shooting algorithm to converge, an appropriate initial guess solution must be provided. A two phase approach to providing initial guess solutions was used. The first guess is simply a velocity that is identical to the velocity found on the manifold/LPO insertion point itself. This represents a zero ΔV for the LPO insertion and is the best case scenario for any insertion burn. From this guess the shooting algorithm generally increases the necessary ΔV and converges on a feasible solution that joins the LEO to the LPO via a transfer arc. Unfortunately, however, an optimal transfer solution is not guaranteed (where "optimal" is defined in terms of ΔV). Occasionally, for a given manifold state, the single shooting algorithm will converge to a feasible solution with a slightly higher ΔV than necessary. To compensate for this shortcoming, a recursive algorithm was written for improving the guess solution. The algorithm begins with the output $\left(\mathbf{Y}^{(1)}\right)$ of the first run of the single shooting algorithm. This output is used in Eq. 11

$$\mathbf{Y}^{(k+1)} = \left(1 - \frac{1}{4k}\right)\left(\mathbf{Y}^{(k)} - \mathbf{V}_{LPO}\right) + \mathbf{V}_{LPO} \tag{11}$$

to compute the initial guess $\left(\mathbf{Y}^{(k+1)}\right)$ that is used in the next iteration of the same shooting algorithm defined above. Eq. 11 serves to decrease the initial guess of the magnitude of the LPO insertion ΔV $\left(\Delta V_{LPO}^{(k)} = \left\|\mathbf{Y}^{(k)} - \mathbf{V}_{LPO}\right\|\right)$. Each successive iteration (with $k \in \left[\ 1,\ \ 10\ \right]$) will decrease the scaling factor of the guess ΔV. Initially, when $k = 1$, the scaling factor is 0.75 but that quickly increases to a factor of 0.975 when $k = 10$.

The shooting method and recursion method discussed above may not guarantee the minimum ΔV transfer is found between a LEO and the manifold/LPO insertion point but it performs very well the majority of the time. Indeed this single recursive single shooting method is easy to program and relatively quick in operation, requiring a handful of seconds to optimize the transfer to a single manifold/LPO insertion point. This technique, combined with PSO, proves to be a low-cost, reliable, and quick method of optimizing impulsive transfers to LPO's.

3 PSO Algorithm

3.1 Fitness Function

Once a insertion point has been evaluated using single shooting the results of that algorithm can be used to assign a "cost" to that point via an objective function known as the fitness function. The fitness function used in this study can be expressed as:

$$J\left(\mathbf{X}_{\mathbf{s.m.}}\right) = c_1 \Delta V\left(\mathbf{X}_{\mathbf{s.m.}}\right) + c_2 \left\|i\left(\mathbf{X}_{\mathbf{s.m.}}\right) - i_{desired}\right\| \tag{12}$$

where i is the two-body orbital inclination of the LEO, $\Delta V = \Delta V_{LEO} + \Delta V_{LPO}$ is the total ΔV required to transfer from LEO to the manifold/LPO insertion point, and c_1 and c_2 are weighting constants chosen by the researcher. Note that the eccentricity of the LEO was not included in the fitness function, as it had been in Abraham et al. [2, 1], because a zero eccentricity orbit was guaranteed by default if the single shooting algorithm converged. In this study, typical values of the weighting constants are $c_1 = 1$ and $c_2 = 1$ or $c_2 = 0$, depending on the importance of inclination to the researcher. When inclination was used, the value of $i_{desired} = 28^o$ with respect to the Moon's orbital plane. If deemed important, the longitude of the ascending node and the true anomaly could also be added to Eq. 12.

3.2 Parametrization of the Search Space

The Particle Swarm Optimization (PSO) algorithm requires the *a priori* definition of a "search space" where it is permitted to search for an optimal solution. In this study, the search space is defined as all states within the invariant stable manifold, $\mathbf{X}_{\mathbf{s.m.}}(\tau_{01}, k) \in W^s$ and within certain bounds. Each state is uniquely defined by exactly two parameters: k and τ. The parameter k represents an individual trajectory member of the stable manifold $(k \in W^s)$ which is generated via the method outlined in the "Invariant Stable Manifolds" section of this paper. The parameter τ_{01} is the second parameter that defines $\mathbf{X}_{\mathbf{s.m.}}$ and is defined relative to τ via a simple mapping function. The time of flight, τ, represents the amount of time required to get from an initial state to a state on the nominal Lagrange point orbit, as defined in the "Invariant Stable Manifolds" section. Unfortunately, it is impossible to define the entire stable manifold as a search space because it is infinite in nature, and a search space (by definition) must be finite. The bounds of τ, therefore, are carefully chosen such that a wide swath of relevant manifold states are captured within the search space and irrelevant manifold states are excluded.

In this study, the bounds of τ are $\tau_{L.B.} \leq \tau_{s.m.} \leq \tau_{U.B.}$ with each bound being defined in one of two ways:

Fast Transfer Search Space

- $\tau_{L.B.}$ being the time that trajectory k crosses the yz-plane located at $x = L_1$
- $\tau_{U.B.}$ being the time that trajectory k crosses first leaves the LPO (i.e. $\tau = 0$).

Slow Transfer Search Space

- $\tau_{L.B.}$ being the time that trajectory k crosses the yz-plane from the positive x direction
- $\tau_{U.B.}$ being the time that trajectory k crosses the yz-plane located at $x = L_1$.

The first search space allows the PSO algorithm to focus on the highly localized manifold that exists very near the LPO. While it is true that a spacecraft inserted into this portion of the manifold may spend a considerable amount of time in the manifold before reaching the destination LPO, this is not a problem. The majority of mission goals (aside from rendezvous) can be achieved in an LPO that is very near the target LPO, but not necessarily on it. Since the majority of Search Space 1 circles around the target LPO, a spacecraft placed into this portion of the manifold can be considered to be in a Lissajous orbit (of approximate size and shape as the target LPO) that flows into the target LPO as time progresses forward. The exploration of this search space is particularly attractive to missions that require short time of flight transfers such as manned missions to LPO's.

The second search space allows the PSO algorithm to focus on the remainder of the stable manifold. Note that any spacecraft inserted into this portion of the manifold will have to spend an appreciable amount of time coasting to the vicinity of the target LPO. The bound, $\tau_{L.B.}$ was chosen as a matter of convenience and practicality. While it is true that the trajectories of W^s continue to flow for an infinite amount of time, one needs to cut off this flow after a finite amount of time due to the limitations of computing power [1, 2]. Since the run times of the PSO method can become quite large, a smaller search space is needed to adequately converge on an optimal solution.

The values of τ are mapped to τ_{01} using the simple relationship that $\tau_{L.B.} = 1$ and $\tau_{U.B.} = 0$. Therefore, the values of τ for a given value of k are mapped to a normalized range of $0 \le \tau_{01} \le 1$. This mapping ensures that values of τ between $\tau_{L.B.}$ and $\tau_{U.B.}$are treated equally, regardless of the value of k and the time of flight between the nominal Lagrange point orbit and the yz-plane located at $x = L_1$. In a similar fashion, the values of k are mapped between $1 \le k \le N$ via a modulus function. In this study, $k = k_{desired}$ **mod** (N). This means, for example, that if $k_{desired} = N + x$ then $k = x$ assuming $0 \le x \le N$. Using this technique, no value of $k_{desired}$ is ever excluded from the search space but is instead looped back onto itself in k-space. In summary, any value of $\mathbf{X}_{\mathbf{s.m.}}(\tau_{01}, k)$ can be uniquely parametrized, in $k\tau$-space, in terms of τ_{01} and k with the boundaries of these parameters being real numbers, $0 \le \tau_{01} \le 1$ and positive integers, $1 \le k \le N$.

3.3 Particle Swarm Optimization

The PSO algorithm is simple yet powerful. It consists of N_p particles which are, initially (i.e. $j = 1$), randomly distributed throughout the search space with a position, $\chi = [\tau_{01}, k]^T$, and velocity, $\omega = \begin{bmatrix} V_\tau, & V_k \end{bmatrix}^T$. Note that the position maps to the state vector in the following manner:

$$\begin{gathered} \chi = [\tau_{01}, k]^T \Rightarrow \\ \mathbf{X}_{\mathbf{s.m.}}(\tau_{01}, k) = \begin{bmatrix} x_{s.m.}, & y_{s.m.}, & z_{s.m.}, & \dot{x}_{s.m.}, & \dot{y}_{s.m.}, & \dot{z}_{s.m.} \end{bmatrix}^T. \end{gathered} \tag{13}$$

Both the velocity and position of each particle in the search space is calculated by Equation 14 and 15, respectively:

$$\begin{gathered} \omega_i^{(j+1)} = \\ C_I\left(1 + R_1(i, j)\right)\omega_i^{(j)} + C_C R_2(i, j)\left(\psi_i^{(j)} - \chi_i^{(j)}\right) + C_S R_3(i, j)\left(\mathbf{Z}^{(j)} - \chi_i^{(j)}\right) \end{gathered} \tag{14}$$

$$\chi_i^{(j+1)} = \chi_i^{(j)} + \omega_i^{(j+1)} \tag{15}$$

with the superscripts representing the j^{th} iteration $(1 \le j \le j_{max})$ of the PSO algorithm and the subscripts representing the i^{th} particle $(1 \le i \le N_p)$. Note that $R_{1,2,3}(i, j)$ represents a random number $0 \le R_{1,2,3}(i, j) \le 1$ following a uniform distribution, and the constants C_I, C_C, C_S, represent the "Inertial," "Cognitive," and "Social" weighting coefficients, respectively. The fitness function, Eq. 12, is evaluated for each particle and $\psi_i^{(j)}$ and $\mathbf{Z}^{(j)}$ are recorded. The "personal best" value, $\psi_i^{(j)} = \begin{bmatrix} \tau_i^{(best)}, & k_i^{(best)} \end{bmatrix}$, represents the best known value of the fitness function recorded by particle i from iteration 1 to j. The "global best" value, $\mathbf{Z}^{(j)} = \begin{bmatrix} \tau^{(best)}, & k^{(best)} \end{bmatrix}$, represents the best known value of the fitness function, recorded by any particle in the swarm, from iteration 1 to j. In this way, the position and velocity of each particle can be calculated for iteration $j + 1$ based on the information contained in iteration j using Equations 14 and 15.

The inertial coefficient directs the particle's motion according to Newtonian mechanics (i.e. motion directed along the current velocity vector). The cognitive coefficient allows each particle to "remember" the best location it has visited and acts as an attractor to that location. Finally, the social coefficient allows each particle to "communicate" with the others in the swarm and attracts particle i to that location. The values of the coefficients used in this study have been inspired by the work of Pontani and Conway [14] and were only modestly modified, via trial and error [1, 2], by the authors to achieve reasonable convergence while still identifying obvious local minima. They are summarized as follows:

$$\begin{array}{l} C_I = 0.15 \\ C_C = 1.00 \\ C_S = 1.00 \end{array} .$$

3.4 Local PSO

The method outlined above is excellent for identifying local minima when only a few minima are present. Unfortunately, if a large number of local minima exist then the global PSO algorithm displays a tendency to converge on a non-optimal local minima instead of the best local minima discoverable. This algorithmic shortcoming exists because the global best solution, $Z^{(j)}$, draws other particles away from what is oftentimes the vicinity of a better local minimum. This is especially true when the depth of multiple local minima are very similar. To avoid this problem a “local” version of the PSO algorithm has been developed by Abraham et al. [2] and utilized in this study. This is accomplished by limiting the ability of the particles’ to communicate over distances grater than some cutoff distance, $\mathbf{r}_{local}$. This mirrors conditions found in nature where collaborating swarms of animals have an inability (or a retarded ability) to communicate over vast distances, thus allowing more time to explore nearby local minima. In this study, $\mathbf{Z}^{(j)}$ is modified to $\mathbf{Z}^{(j)}_{local(i)}$ by utilizing the best value of a local swarm defined as all particles within radius, $\mathbf{r}_{local}$, of particle i. If particle i can not “see” a distant particle then that distant particle has no influence over the value of $Z^{(j)}_{local(i)}$. Typically, $\mathbf{r}_{local} = \left[\begin{array}{cc} \frac{1}{20}, & \frac{1}{16}N \end{array}\right]^T$ in this study.

As the PSO algorithm evolves over j iterations, the particles in the swarm begin to collect around various local minima. It becomes possible to define a convergence metric, γ, for the entire system. This metric is defined by

$$\gamma^{(j)} = \frac{N_C^{(j)}}{N_p} \tag{16}$$

where $N_C^{(j)}$ represents the number of particles that have converged to the vicinity of $\mathbf{Z}^{(j)}_{local(i)}$. The vicinity of $\mathbf{Z}^{(j)}_{local(i)}$ is defined to be a circular area of $k\tau$-space, centered on $\mathbf{Z}^{(j)}_{local(i)}$, with a radius that is roughly 14% the size of r_{local} and an area that is roughly 2% the area of a circle of radius r_{local}. Using this definition of convergence it becomes possible to track the convergence of the PSO algorithm as a function of j and even terminate the algorithm early if a sufficient value of γ is reached.

Search Space Boundary Conditions

Occasionally, a particle attempts to exit the permissible search space to which it must be bound. In such a case, a series of rules is followed to gently guide the particle back into the search space and continue its search for the best local minima discoverable. One reason why a particle may attempt to exit the search space is because its velocity is too large. In this case, a saturation limit is imposed on the velocity vector such that $\omega \leq \omega_{max} = \left[\begin{array}{cc} \pm\tau_{max}, & \pm k_{max} \end{array}\right]^T = \left[\begin{array}{cc} \pm\frac{1}{2}, & \pm\frac{1}{2}N \end{array}\right]^T$. In general, this velocity saturation limit is very high and only acts upon extremely unreasonable velocity values. As a consequence, some values of χ still fall outside the search space. If this is the case then the τ_{01} component of position is bounded by the saturation limits

$$\tau_{01} = \begin{cases} \tau_{max} = 1 & for\, \tau_{01} > \tau_{max} \\ \tau_{min} = 0 & for\, \tau_{01} < \tau_{min} \end{cases}$$

and the k component is bounded by the modulus function $k = k_{requested} \textbf{ mod } (N)$. The velocity of this particle is also reset to $\omega = \mathbf{0}$ to prevent the particle from exiting the search space during the subsequent iteration.

4 Data and Results

In this paper, ballistic trajectories were integrated according to Eq. 1. All numeric integration was performed using MATLAB's ODE113 solver; a variable order Adams-Bashforth-Moulton PECE solver with a relative and absolute tolerance of 10^{-13}. This method was chosen because it is relatively efficient when using stringent tolerances as well as being efficient with computationally intensive functions. The computer utilized 2 parallel cores on a 3GHz Intel Core 2 Duo processor. The optimization of the search space typically required 15 to 20 hours using 300 particles and 30 iterations of the PSO algorithm.

4.1 Destination Orbit and its Associated Manifold

All data in this study has been generated using an Earth-Moon, L_1, northern halo orbit as the nominal destination orbit. This orbit has a period of $P = 2.31339$[tu] (63 days) and an initial state of

$$\mathbf{X_0} = \begin{bmatrix} 0.866224052875085 \\ 0.011670195668094 \\ 0.186912185139037 \\ 0.013870554690931 \\ 0.245270168936540 \\ 0.021792775971957 \end{bmatrix}$$

expressed in [du] and [vu]. The orbit is displayed in Figure 2 with the states being discretized into $k_{max} = N = 791$ points, with each point corresponding to a particular trajectory, k, on the invariant stable manifold.

This orbit was arbitrarily chosen as an example of how the PSO technique can optimize a transfer trajectory to a complicated and highly three dimensional LPO. According to Dynamical Systems Theory (DST) the invariant stable manifold of this northern halo orbit may be generated by the following procedure:

1. Select a point on the orbit and integrate the State Transition Matrix (STM) forward in time for one period.

2. Calculate the eigenvectors associated with the direction of the stable manifold at this point.

3. Multiply this eigenvector by a small number, ε, and add/subtract this small perturbation to the initial point.

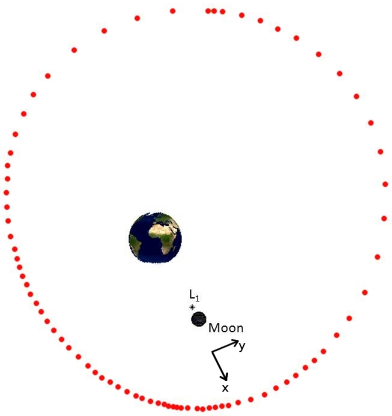

Figure 2: The nominal Earth-Moon L_1 northern halo orbit. The orbit is broken into $N = 791$ points with every 10^{th} point displayed in this figure.

4. Propagate this state vector backward in time until it crosses the x-axis from the $-x$ direction.

This procedure can be repeated numerous times; once for each point on the nominal orbit. All trajectories generated by points on the nominal orbit comprise the invariant stable manifold, W^s, of the nominal orbit. Figure 3 illustrates the invariant stable manifold of the Earth-moon, L_1, northern halo orbit.

4.2 Fast Transfer

The PSO method was used to optimize the "Fast Transfer" search space described above. This was accomplished by setting $c_1 = 1$ and $c_2 = 0$ (in the fitness function, Eq. 12) for the case of optimization with respect to any inclination, and $c_1 = 1$ and $c_2 = 1$ for the case of a 28^o desired LEO inclination. In both cases, the optimal trajectory is shown in Figure 4. Notice how the spacecraft immediately enters a Lissajous orbit that would be useful for the majority of Earth-Moon L_1 applications such as communications relay, Earth/Moon observation, and mission staging areas. As noted in Table 1, the Lissajous orbit will deliver the spacecraft into the target LPO over a period of $60 - 90$ days where it can rendezvous with another spacecraft that is on the exact same orbit. Alternatively, to speed up a desired rendezvous, the spacecraft could execute a small maneuver to decrease a rendezvous time using a modest amount of ΔV.

4.3 Slow Transfer

The PSO method was also used to optimize the "Slow Transfer" search space described above. As before, this was accomplished by setting $c_1 = 1$ and $c_2 = 0$ for the case of optimization with respect to any inclination, and $c_1 = 1$ and $c_2 = 1$ for the case of a 28^o desired LEO inclination. In both cases, the optimal trajectory is shown in Figure 5. Notice how much longer it takes for the spacecraft to reach the vicinity of L_1. Adding the values of the "TOF to Insertion" and "TOF in pre-L_1 Manifold" columns of Table 2 together it is possible to calculate the time of flight required to reach the vicinity of L_1 and directly compare it to that of a fast transfer. Disregarding LEO

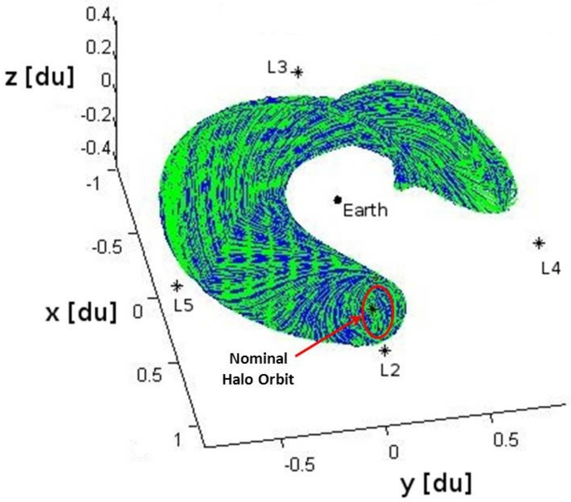

Figure 3: Invariant stable manifold of the nominal, L_1 northern halo orbit (green-blue trajectories). Note that the manifold never approaches the vicinity of Low Earth orbit.

Table 1: Summary of Fast Transfer

	J	ΔV_{LEO} [km/s]	ΔV_{halo} [km/s]	Total ΔV [km/s]	TOF to Insertion [days]	TOF in Manifold [days]	TOF Total [days]
Fast Transfer, Any Inclination	3.443	3.070	0.373	3.443	4.98	90.00	94.98
Fast Transfer, 28^o Inclination	3.585	3.069	0.444	3.513	4.89	60.94	65.83

inclination, an optimized slow transfer requires a 154.73 day flight compared with a 94.98 day flight as seen in Table 1. This means that a fast transfer could move a spacecraft to the vicinity of L_1 with approximately 100 [m/s] less ΔV and much less transfer time. Similar results exist for the 28^o inclination mission. In this case the fast transfer requires 160 [m/s] less ΔV.

It is interesting to note that Alessi et al. [3] found that the optimal manifold insertion point was manifold apogee; that is the point on the manifold that is the maximum distance away from the Earth. In this study, all optimal manifold insertion points discovered using the slow transfer search space were very near apogee. For example, the optimal insertion point found by PSO for the “Slow Transfer, Any Inclination” case was only $30,000$ km away from the manifold apogee. This is only five Earth radii in size away from the apogee condition (which is relatively small when the search space is nearly ±60 Earth radii in diameter). Even better, was the optimal insertion point for the “Slow Transfer, 28^o Inclination” case at $3,000$ km from manifold apogee. This is less than half an Earth radius and is about as close as one could hope for when matching the results from Alessi et al. Strong agreement with previously published work [3, 15] corroborates the efficacy of the PSO method as applied here.

Table 2: Summary of Slow Transfer

	J	ΔV_{LEO} [km/s]	ΔV_{halo} [km/s]	Total ΔV [km/s]	TOF to Insertion [days]	TOF in pre-L_1 Manifold [days]	TOF in post-L_1 Manifold [days]	TOF Total [days]
Slow Transfer, Any Inclination	3.548	3.040	0.507	3.548	3.89	19.04	131.80	154.73
Slow Transfer, 28^o Inclination	3.691	3.047	0.629	3.676	3.76	11.51	121.87	137.14

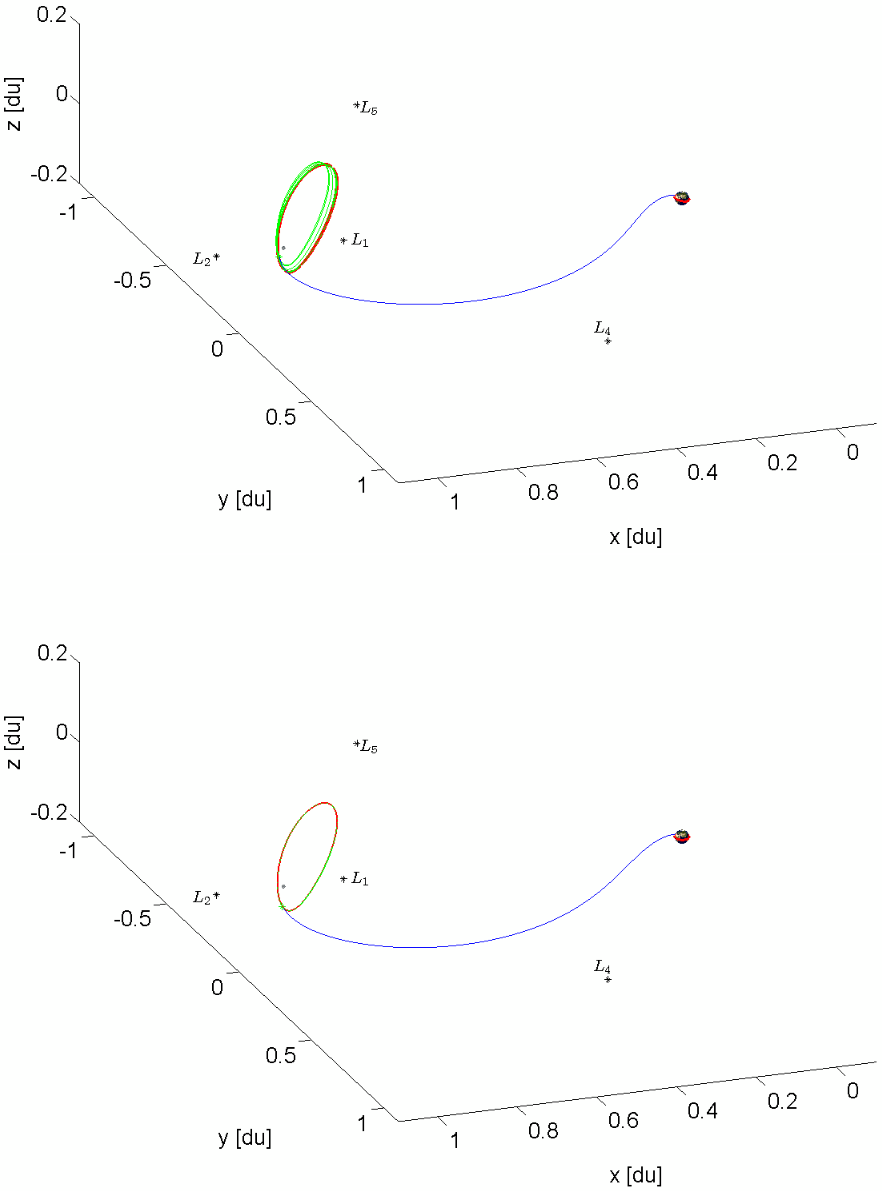

Figure 4: Fast transfer from a 400 km altitude LEO (red) to the target LPO (red) via a cislunar coast (blue) following a cislunar injection burn and a stable manifold coast (green) following a manifold injection burn. Top: Optimized for any inclination. Bottom: Optimized for 28^o inclination.

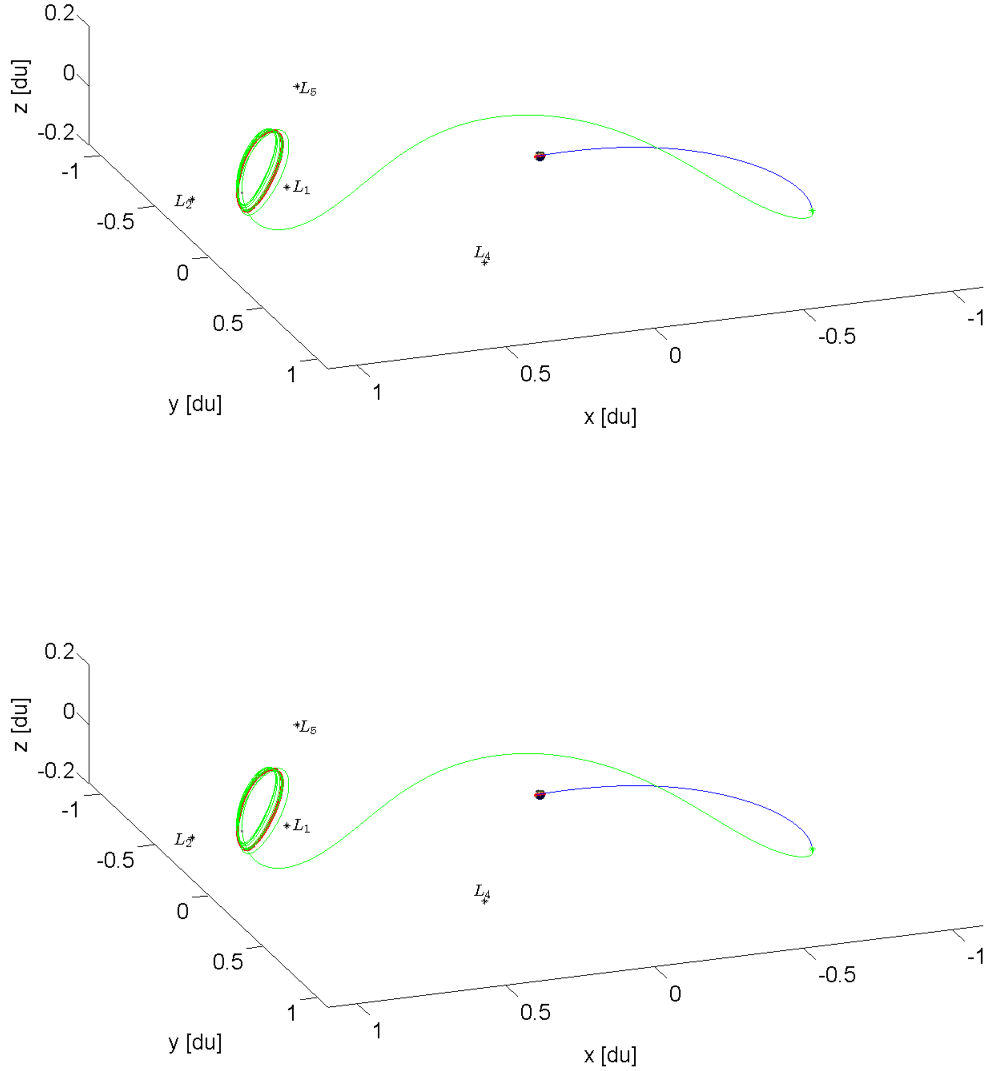

Figure 5: Slow transfer from a 400 km altitude LEO (red) to the target LPO (red) via a cislunar coast (blue) following a cislunar injection burn and a stable manifold coast (green) following a manifold injection burn. Top: Optimized for any inclination. Bottom: Optimized for 28^o inclination.

5 Summary and Conclusions

In this study, an evolutionary algorithm, called Particle Swarm Optimization, was used in conjunction with a traditional, gradient-based optimization method called Single Shooting. The fusion of a gradient-based technique with an evolutionary algorithm attempts to blend the strengths of both methods while minimizing their weaknesses. While the PSO method may be much slower than gradient-based algorithms, its robust global optimization capability when used to optimize a non-convex objective function far exceeds that of gradient-based methods; especially when the objective function is non-differentiable or nearly so. On the other hand, using a shooting method to optimize a single manifold insertion point is relatively fast (a few seconds) and can determine a very efficient two-burn transfer trajectory. While shooting does not guarantee the optimal transfer solution is found for a given insertion point, the method of iterative shooting does reduce the likelihood of a non-optimal result. The PSO/Shooting method, described above, is relatively simple to program and requires a modest amount of resources to run. Typical run-times for a desktop computer are on the order of $10-20$ hours but could be significantly reduced (by a few orders of magnitude) by utilizing a parallel computing cluster. Due to the ease of programming and reasonable run-time, the PSO/Shooting method is useful in preliminary optimization of space mission design or for trajectory pruning applications.

The optimization of the sample problem given in this study is also noteworthy. In agreement with Alessi et al., the PSO/Shooting method identified the optimal manifold insertion point as apogee when considering the slow-transfer search space. This result adds creditability to the PSO/Shooting method since Alessi et al. used an entirely different method to gather their data, yet provide similar results. In light of this published work, however, it is a bit surprising to note that the fast transfer was superior to the slow transfer in terms of both ΔV and time of flight. It is currently unknown weather this result is characteristic of all LPO or unique to the one chosen for this study. Given enough computational power it may be useful to investigate the influence of the choice in LPO over the optimization results. For example, LPO's in different systems should be studied (Earth-Moon, Sun-Earth, Sun-Mars, etc.), LPO's about the L_1, L_2, L_3, and $L_{4,5}$ points could be studied, and even various shapes and families of LPO's should be studied to see if there are any commonalities between them. Optimal manifold insertion points at apogee may be a common characteristic of most LPO's but not all. It may be useful to future mission designers to know what characteristics of a LPO lend themselves to optimal fast transfers rather than slow transfers. This is especially true for human spaceflight to LPO's.

References

[1] Andrew J. Abraham, David B. Spencer, and Terry J. Hart. Optimization of Preliminary Low-Thrust Trajectories From GEO-Energy Orbits to Earth-Moon, L1, Lagrange Point Orbits Using Particle Swarm Optimization. AAS 13-925, Hilton Head, SC, August 2013. AAS Astrodynamics Specialist Conference.

[2] Andrew J. Abraham, David B. Spencer, and Terry J. Hart. Preliminary Optimization 2-D Optimization of Low-Thrust, Geocentric-to-Halo-Orbit Transfers, via Particle Swarm Optimization. AAS 14-315, Santa Fe, NM, January 2014. AIAA Space Flight Mechanics Meeting.

[3] Elisa Alessi, Gerard Gomez, and Joseph Masdemont. Two-manoeuvres transfers between leos and lissajous orbits in the earth-moon system. *Advances in Space Research*, 45:1276–1291, 2010.

[4] Vassilis Angelopoulos. The ARTEMIS Mission. *Space Science Review*, pages 1–23, 2010.

[5] Bruce A. Conway, editor. *Spacecraft Trajectory Optimization*. Cambridge, 2010.

[6] E. Davis. Transfers to the Earth-Moon L3 Halo Orbits. In *AAS/AIAA Astrodynamics Specialist Conference*, Minneapolis, MN, August 2012.

[7] Robert W. Farquhar. A Halo-Orbit Lunar Station. *Astronautics & Aeronautics*, 10:59–63, 1972.

[8] Robert W Farquhar, David W Dunham, Yanping Guo, and James V McAdams. Utilization of Libration Points for Human Exploration in the Sun-Earth-Moon System and Beyond. *Acta Astronautica*, 55:687–700, August-November 2004.

[9] Robert W. Farquhar and Ahmed A. Kamel. Quasi-Periodic Orbit About The Translunar Libration Point. *Celestial Mechanics*, 7:458–473, 1973.

[10] T. A. Heppenheimer. Steps Toward Space Colonization: Colony Location and Transfer Trajectories. *Journal of Spacecraft and Rockets*, 15(5):305–312, September - October 1978.

[11] Lockheed Martin. Early Human L2-Farside Missions, 2010. Accessed on 3-14-2013, http://www.lockheedmartin.com/content/dam/lockheed/data/space/documents/orion/LMFarsideWhitepaperFinal.pdf.

[12] Thomas A. Pavlak. Mission Design Applications In The Earth-Moon System: Transfer Trajectories and Stationkeeping. Master's thesis, School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN, May 2010.

[13] Lawrence Perko. *Differential Equations and Dynamical Systems*. Springer, 175 Fifth Avenue, New York, NY, 10010, USA, 3rd edition, 2001.

[14] Mauro Pontani and Bruce A. Conway. Particle Swarm Optimization Applied to Space Trajectories. *Journal of Guidance, Control, and Dynamics*, 33(5):1429–1441, September-October 2010.

[15] F. Renk, M. Hechler, and E. Messerschmid. Exploration Missions in the Sun-Earth-Moon System: A Detailed View on Selected Transfer Problems. *Acta Astronautica*, 67:82–96, July-August 2010.

[16] Theodore H. Sweetser, Stephen B. Broschart, Vassilis Angelopoulos, Gregory J. Whiffen, David C. Folta, Min-Kun Chung, Sara J. Hatch, and Mark A.Woodard. ARTEMIS Mission Design. *Space Science Review*, 165:27–57, March 2012.

[17] Victor Szebehely. *Theory of Orbits: The Restricted Problem of Three Bodies*. Academic Press Inc., New York, NY, 1967.