MODELING AND OPTIMIZATION OF EARTH-MOON TRANSFERS

Isabel Ibáñez Jiménez(1), Sofya Spiridonova(1), Ron Noomen(2)
(1) Deutsches Zentrum für Luft- und Raumfahrt (DLR), Münchener Straße 20, 82234 Weßling, Germany.
isabel.ibanezjimenez; sofya.spiridonova@dlr.de
(2) Delft University of Technology, Kluyverweg 1, 2629 HS, Delft, The Netherlands. r.noomen@tudelft.nl

Abstract—**The primary objective of this study is to develop a tool to calculate optimal transfer trajectories from a user-defined Earth-bounded orbit to a user-defined Moon-bounded orbit, using a bi-impulse direct transfer under the influence of a full dynamics model with perturbations, reflecting the physical environment. The objective is to obtain the required impulse ($\Delta\bar{V}$) efficiently so that it serves as an initial guess for further analysis. Two tools are developed to achieve this goal. The first tool employs a global optimization algorithm, a Particle Swarm Optimizer (PSO), to find an initial guess within a simplified dynamics model, exploring the user-defined search space. The second tool employs a gradient-based Sequential Linear Least SQuares Programming (SLLSQP) optimizer to refine the initial guess and include the relevant perturbations that act in real life. The capabilities of the tools are demonstrated through several test cases. The first test case involves transferring from a circular Low Earth Orbit (LEO) to a circular near-polar Low Lunar Orbit (LLO). A second and a third test case involving transfers from a LEO and a Geostationary Transfer Orbit (GTO) to an eccentric lunar orbit were also successfully evaluated. These test cases validate the functionality of the method and demonstrate its versatility in handling various scenarios. In conclusion, the developed tools provide efficient and robust solutions for optimizing direct transfers from Earth to the Moon under the influence of real-life perturbations.**

I. INTRODUCTION

The optimization of Earth-Moon transfers, including a detailed dynamics model, has been a focus of research since the start of the Moon race in 1959. It remains an area of interest within the space sector, especially in discussions about establishing a human presence in lunar orbit as a step towards exploring the Solar System. Ongoing technological and theoretical advancements present numerous opportunities for new research projects in this field.

This project aims to create a flexible tool for optimizing trajectories from any Earth-bounded orbit to any Moon-bounded orbit, using a dynamics model with user-selected perturbations. The tool seeks to offer initial estimates for required propellant, maneuvers, and optimal transfer timings, while considering a dynamics model that reflects the physical reality of such missions.

Existing studies often use models like the Circular Restricted Three-Body Problem (CR3BP) [1] [2], which offer many advantages but struggle to incorporate key perturbations like Earth's spherical harmonics or solar radiation pressure. Furthermore, these studies typically focus on specific trajectory solutions rather than offering user-friendly tools for analyzing and optimizing transfers based on auxiliary user-defined orbits.

The transfer studied throughout this project is a direct one, as it is the quickest and simplest transfer to the Moon, and hence one of the most used transfers in past and current missions. They have been extensively researched, and first-order approximations are available. Their short Time of Flight (ToF) offers advantages in terms of radiation exposure and mission control. Additionally, the spacecraft remains near Earth and the Moon, minimizing the influence of the Sun's gravity, simplifying approximations, and aiding in emergency situations. However, direct transfers require a significant ΔV, impacting mission costs compared to low-energy transfers, which make use of gravitational assists from celestial bodies to increase the orbital energy of the satellite [3]. Bi-elliptic transfers reduce ΔV but extend transfer times. Weak Stability Boundary (WSB) transfers, particularly with ballistic capture, offer improved ΔV performance [4]. Low-energy transfers access a broader range of lunar orbits, especially beneficial for periodic three-body orbits. Manifold transfers, exploiting CR3BP dynamics, decrease ΔV without significantly extending ToF, making them attractive for reaching periodic or quasi-periodic orbits. However, they require a CR3BP formulation due to manifold rotations in an inertial frame.

This work focuses on high-thrust direct two-impulse transfers, which are widely employed in initial investigations and are considered reliable by the community. This requires an initial state for trajectory optimization, which can be defined by the Keplerian elements of an Earth-bounded orbit (a, e, i, ω, Ω). The precise starting position within this orbit is determined by an additional variable that needs to be analyzed. For the final state, options include Keplerian orbits around the Moon or quasi-periodic orbits like halo or Near-Rectilinear Halo Orbits (NRHO). Keplerian orbits can be modelled in both CR3BP and ephemeris models. They are straightforward and widely used, though they are not static in an Earth-centered inertial frame. Quasi-periodic orbits exploit CR3BP dynamics, offering benefits for lunar missions but require pre-generation and storage of state information, and cannot be generated in an inertial frame. This project primarily focuses on Keplerian orbits due to their simplicity and inertial frame adaptability. Throughout this paper, an approach to find the optimal Earth-Moon direct two-impulse transfers, from a user-defined Earth-bounded orbit to a user-defined Moon-bounded orbit including a user-selected dynamics model will be analyzed, including the optimization methods, the optimization parameters, and the possibility of both global optimization methods and gradient-based optimization methods. Lastly, the optimization process does not encompass the launcher ΔV for reaching the initial orbit, though it is crucial for assessing overall mission costs.

II. Methods

The study of optimal direct two-impulsive transfers involves a hybrid approach. First, a global optimization (Section II-A) is conducted using SEMpy (Sun-Earth-Moon system in Python) [5], an open-source tool developed by ISAE-SUPAERO. Next, a gradient-based optimization is used to further refine the solution and address a more complex dynamics model (Section II-B). This optimization employs EMTOS (Earth-Moon Trajectory Optimiyation Software), a self-developed software making use of DLR internal libraries in Fortran, and the SLLSQP optimizer [6]. The methodology is depicted in Figure 1.

By using this hybrid method, the developed software will be able to find a close-to-optimal solution under simplified conditions throughout the whole search space in an acceptable time using a global optimization algorithm, and then use the obtained solution as an initial guess for a local optimization under the influence of a full dynamics model.

A. Global optimization

The global optimization method employed is a Particle-Swarm Optimizer (PSO) [7]. Its main advantages include ease of implementation, quicker convergence compared to other global search algorithms, robustness, and the ability to handle multi-modal functions. However, it can converge prematurely to a sub-optimal solution if the optimizer's parameters are not properly tuned.

The first step towards practical optimization is to define the fitness function and the design variables. Then, a set of solutions or "particles" is generated, and their fitness, associated with a combination of design variables, is computed. The optimizer will then seek the particle with the minimum fitness value. As mentioned earlier, tuning the parameters of the optimizer is crucial to ensure the best solution in the shortest possible time.

i. Fitness function

The fitness is determined by the combined magnitude of the two maneuvers required to transfer from an Earth-bounded orbit to a Moon-bounded orbit in km/s.

$$\text{Fitness} = |\Delta \vec{V}_1| + |\Delta \vec{V}_2| \quad (1)$$

Large values are assigned to the fitness if there are convergence issues or if the final trajectory brings the satellite too close to either Earth or the Moon (lower than 100 km above Earth's surface or 50 km above the lunar surface).

ii. Design variables

The selection of design variables impacts the size and complexity of the search space, as well as the time required to calculate the fitness for each particle. Hence, the chosen design variables are as follows: the departure state, the arrival state, and the ToF, the initial epoch (et0), which significantly influences the ΔV required to travel from Earth to the Moon.

The parameters defining the initial and final orbits are provided as inputs by the user in the form of Keplerian elements. Hence, the initial and final states can be defined by two parameters (τ_1 and τ_2 respectively). τ_i represents the time elapsed since a reference point in the orbit divided by the orbital period, ranging from 0 to 1. These variables are depicted in Figure 2.

iii. Dynamics system

The choice of the dynamics system for this problem is influenced by the required accuracy, the methods used, and the acceptable computational time. The PSO optimization requires numerous function evaluations, thus employing a simple model will significantly reduce the total CPU time and decrease the complexity of the problem. Therefore, only Earth's and Moon's point-mass gravity fields will be included. However, it was observed that in some cases, when later propagating the obtained trajectory with a complete dynamics model, the solution failed to reach the Moon and could not serve as an initial guess for the local optimization. To overcome this issue, the spherical harmonics of Earth were incorporated into the orbit propagation only before the first maneuver. This adjustment favored obtaining a solution that, when propagated with a full dynamic model, successfully reached an orbit around the Moon.

iv. Solution generation

Generating the solution involves determining the best method to compute the fitness value (the required ΔV) from a particle with design variables (τ_1, τ_2, ToF, et0). Subsequently, the initial and final states (position and velocity) in the EME2000 reference frame are calculated.

Once the initial and final states are computed, the required ΔV is determined using a combination of

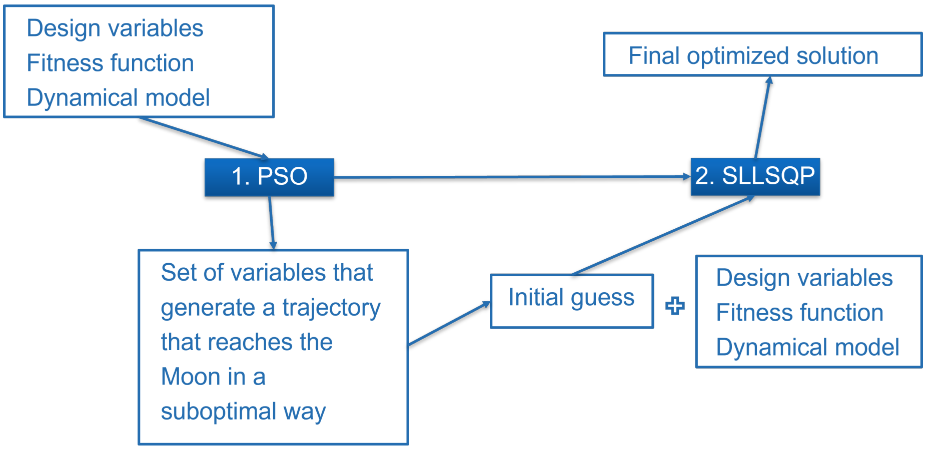

Fig. 1. General diagram of the optimization method.

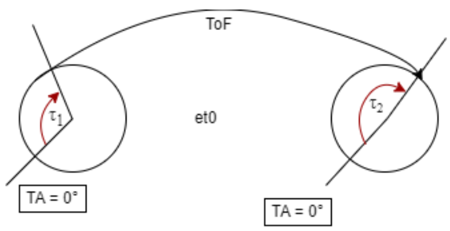

Fig. 2. Design variables diagram for global optimization.

Lambert's solver and a multiple-shooting method implemented with SEMpy. Lambert's problem is an analytical two-body method used to compute transfer orbits between two specified position vectors within a given transfer time [8]. However, the solver can only consider either Earth's or the Moon's point-mass gravity field and cannot directly account for both bodies simultaneously. Therefore, the solution serves as an initial guess for a multiple-shooting method, which subsequently corrects for the additional gravity field [9].

The multiple-shooting method discretizes the trajectory into $n-1$ legs with n patch points to enhance convergence. Differential corrections are then applied iteratively to determine the position and velocity at each patch point, ensuring a continuous trajectory. The optimal number of patch points varies depending on the specific problem, balancing convergence time and robustness. Here, a study has been conducted by analyzing the convergence and CPU time for 5 different cases.

It is important to note that this method struggled to converge when the initial guess, calculated without considering the influence of the Moon, resulted in a trajectory too close to the lunar center. This issue arises due to the significant computational resources required to propagate the State Transition Matrix (STM) in such cases. To prevent being trapped in an undesirable solution, a threshold was implemented to exclude cases where the trajectory of the initial guess approached within 1000 km of the lunar center of mass. This value was determined based on CPU time and convergence considerations.

v. Optimizer tuning

The primary criteria for analyzing the optimizer's performance is its robustness, including its ability to find the global optimum avoiding getting stuck in local optima, yielding consistent solutions with different initial conditions (seeds), and its convergence rate.

The parameters requiring tuning are the population size, number of generations, inertia weight (ω), and social and cognitive components (η_1, η_2) of the PSO. ω determines the influence of the particle's previous velocity, thereby regulating the extent to which the particle's velocity is influenced by its past behavior [10]. η_1 represents the shared experience of the swarm, updating the term associated with the influence of the swarm's global best position on the particle's behavior. η_2 enables each particle to update its position based on its own historical information, allowing it to explore the search space around its best-known solution so far [11].

ω, η_1 and η_2 are tuned independently, maintaining a constant population size and number of generations. Once their optimal values are determined, the best combination of population and number of generations is found. To achieve this, a problem with each combination of parameters was solved with 5 different seeds, and the solution was selected based on robustness, hence ensuring that solutions obtained for each seed provided very similar optimum particles and fitness values, and a reasonable CPU time.

B. Gradient-based optimization

As previously mentioned, a gradient-based optimization will be used to calculate the optimal solution, considering additional perturbations, while using the PSO-obtained solution as an initial guess. This method requires a new problem formulation due to the implementation of gradient-based optimization methods, specifically of Sequential Least Squares Programming.

The solution generation process involves propagating the satellite's trajectory from its initial state. This propagation comprises a first maneuver to start the transfer trajectory and a second maneuver to reach the lunar orbit. An important step for gradient-based optimization is to include constraints, both equality and inequality, to ensure that the solution found consistently reaches the desired orbit. Subsequently, the Keplerian elements of the resulting orbit are computed and compared to the objective ones. The dynamics model used for this propagation includes additional perturbations, such as the presence of the Sun's gravity field, a gravitational model with spherical harmonics for both Earth and the Moon, solar radiation pressure, and drag. The propagation with these perturbations within the software has been thoroughly tested.

The fitness function remains consistent with PSO (Equation 1). However, adjustments to the design variables are required.

i. Design variables

The solution will be generated through the propagation of the satellite's state. The design variables are:

- Time in the first orbit: t_1.
- The velocity components of the first maneuver in EME2000 reference frame: ΔV_{1x}, ΔV_{1y}, ΔV_{1z}.
- Time of flight: ToF.
- The velocity components of the second maneuver in an EME2000 reference frame: ΔV_{2x}, ΔV_{2y}, ΔV_{2z}.

The optimizer has proven to be capable in effectively optimizing this number of parameters.

ii. Constraints

Five inequality constraints are imposed for the semi-major axis, the eccentricity, the inclination, $\mathbf{\Omega}$ and ω. The general inequalities are as follows:

$$\begin{aligned} X &< A + \Delta \\ X &> A - \Delta \end{aligned} \tag{2}$$

where A is the objective value, Δ is the permitted variance, and X is the actual value obtained from the propagation.

iii. Optimizer tuning

To ensure that the found solution in this specific problem is optimal and the method is robust, certain parameters within the SLLSQP optimizer need to be fine-tuned. These parameters include the scaling factor for decision vector parameters (SP), the scaling factor for constraints (SC), the scaling factor for the fitness (cost) function (SCF), the accuracy (ACC), and the directional derivative of the perturbation parameters (DDPP).

Each parameter has been examined independently in order to comprehend its individual effect. This has been done by analyzing the results with different values, specially looking at the fitness. The main reason for errors in the SLLSQP optimizer is obtaining positive directional derivatives.

C. Software testing

Lastly, both software packages were tested to validate the premise that a transfer can be optimized under a full dynamics model, based on the idea that a PSO optimizer can identify a solution close to optimal using a simplified dynamics model, which can then be refined and optimized further by the SLLSQP optimizer to account for the perturbations. To achieve this, various test cases have been examined. All with a $\mathbf{\Omega}$ of $0°$.

E	h [km]	e [-]	i [deg]	ω [deg]
E1	250	0	28.4	0
E2	24389	0.7285	6	178

TABLE I
EARTH ORBITS.

M	a [km]	e [-]	i [deg]	ω [deg]
M1	1837.4	0	85	0
M2	6541.1	0.6	56.2	90

TABLE II
MOON ORBITS.

Test Case	Earth orbit	Lunar orbit
1	E1	M1
2	E1	M2
3	E2	M2

TABLE III
SUMMARY OF THE TEST CASES.

As found in Table III, test case 1 (TC1) explores a transfer from a LEO (E1) to a LLO (M1), and it has been established as the basic case for all the tuning procedures. Test case 2 and 3 (TC2, TC3) allow to estimate the potential propellant savings for a satellite departing from a LEO (E2) compared to a GTO (E3) to a lunar orbit (M2) that can be used for a polar coverage constallation.

The common input data provided for all the test cases for the optimization with the PSO is as follows:

- The initial epoch around which the optimization is conducted: 23$^{\text{rd}}$ July 2024.
- Spherical harmonics for the Earth up to degree and order 10.
- Boundaries for the ToF: 0.1 days and 6 days.
- Boundaries for the initial epoch: 15 days before and after the initial epoch given.
- Boundaries for τ_1 and τ_2: from 0 to 1.

The general input data provided for all test cases for the optimization with SLLSQP is as follows:

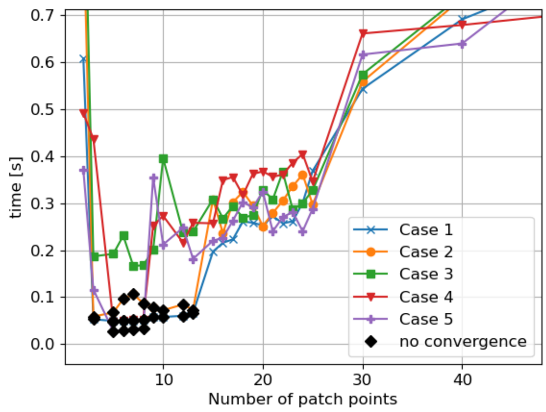

Fig. 3. Computational time it takes to obtain a Lambert + multiple-shooting solution for a range of patch points.

– Mass of the satellite: 1500 kg.
– Drag area: 3.5 m^2.
– C_D: 2.3.
– A_{CR}: 10 m^2.
– C_R: 1.3.
– Allowed deviations for a: 1 km.
– Allowed deviations for e: 0.001.
– Allowed deviations for i: 1°.
– Allowed deviations for Ω: 10°, only available if inclination is not 0°.
– Allowed deviations for ω: 10°, only available if eccentricity is not 0°.

III. Results

A. *Number of patch points in multiple-shooting method*

The PSO algorithm's solution was obtained using a hybrid method that combines a Lambert solver with a multiple-shooting approach. This method requires setting the number of patch points, as it affects both robustness and CPU time. As depicted in Figure 3, it was observed that using 2 patch points, corresponding to a single-shooting method, consistently converged but consumed significant time per iteration. Additionally, it was noted that CPU time increased linearly for more than 25 patch points. Between 3 and 15 patch points, convergence is not always achieved. Therefore, a compromise of 20 patch points was chosen.

B. *Tuning PSO parameters*

The default value for the inertia weight given by SciPy, which we found was the best for this problem, is 0.7298. It was observed that lower values resulted in longer computational times (up to 16 minutes longer), while larger values led to shorter times but more scattered solutions. Regarding the social and cognitive components, lower values were found to produce more robust results. As for CPU time, generally, larger values yielded faster results, but with a difference of only around 3 minutes, an acceptable value. A value of 1.5 was selected for the social and cognitive components. An important result to be noted is that the effect of the inertia weight value was larger both for robustness and computational efficiency.

The population size and the number of generations were tested together. Lower numbers resulted in lower CPU time but lacked robustness, while larger numbers ensured robustness at the cost of significantly slower performance, with differences of up to 100 minutes. Therefore, a balanced combination was sought. The optimal combination for the majority of the problems was found to be 50 for the population size and 75 for the number of generations. However, it is crucial to emphasize the importance of assessing whether the optimizer has indeed converged with these settings.

C. *Tuning SLLSQP parameters*

The SLLSQP parameters also required tuning. Firstly, it has found that SP and SC have no measurable impact for the solution on this problem.

Analysing the SCF demonstrates that the optimizer prioritizes decreasing the fitness value when its value is higher. However, setting SCF too high can lead to errors if the initial solution is not sufficiently close. To mitigate this, a loop that gradually increases the scaling factor of the fitness function has been implemented. This iterative approach uses the solution from the previous optimization as input, raising the scale value gradually, aiding the optimizer in reaching the solution with the lowest fitness value possible.

Using a low ACC yields the same result as using a value of 1, and a high ACC leads to the same solution as the initial guess without optimization, as the initial guess aligns closely with the desired orbit, already satisfying all the inequality constraints.

The DDPP represents each parameter's derivatives, hence a value is required for each parameter. If all DDPPs are too low, an error can occur due to a positive directional derivative. Furthermore, there is a dependence on this value individually, as well as in combination. Through various cases, it was found that the value of 10^{-5} for all parameters effectively works across different initial guesses and will be set as the default. However, users should adjust this parameter for each specific case to seek improved results.

D. *Test Case 1*

The first test case consists on the transfer from LEO to LLO. After running the PSO for 5 different seeds, solutions in Figure 4 are obtained. S30 (i.e. seed number 30) brings the best solution, very similar to that of S10, with a fitness of 4773.57 m/s. It needs to be highlighted that all seeds lead to very similar results except for S50, with a fitness 12 m/s larger and very different

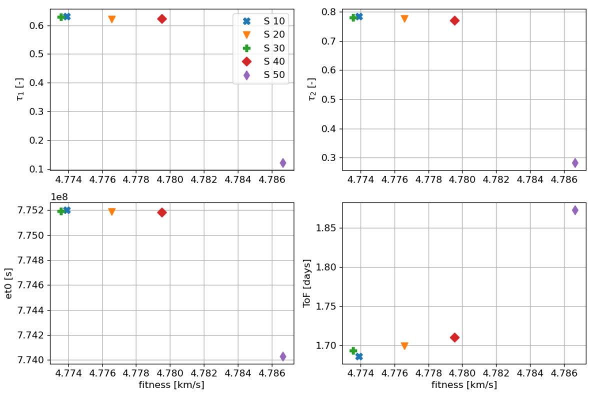

Fig. 4. Design parameters for the optimal particles found for each of the five seeds in test case 1.

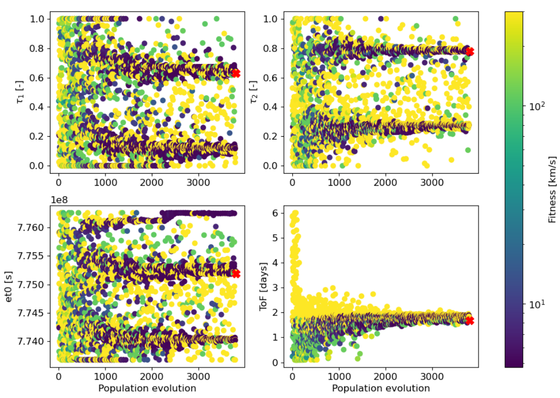

Fig. 5. Evolution of the whole population for test case 1 and seed 30.

design variables. The difference in τ_1 and τ_2 is 0.5, corresponding to a transfer from and to opposite sides of both orbits, and showcasing a suboptimum solution. Both the optimal and suboptimal solutions can be seen in the evolution of all particles throughout the optimization, found in Figure 5 for seed 30. The difference in epochs between seed 50 and the value for the other seeds is approximately 13.47 days, nearly half of the Moon's orbital period.

The final step in understanding the solutions is to plot the trajectories. Figure 6 shows the trajectories followed with the best result for each seed. It is apparent that the trajectories for the initial four seeds cluster around a similar epoch, whereas for S50, the trajectory veers entirely in the opposite direction. This observation is important for mission operations, indicating that despite not achieving the absolute optimal solution, an alternative solution half a lunar orbit later giving only around a 10 m/s loss in ΔV remains viable.

As previously mentioned, these solutions do not account for additional perturbations except for the spherical harmonics of Earth during the initial part of the transfer when the satellite is orbiting Earth. Therefore, this solution requires further analysis to determine the optimal parameters under a more complete dynamics model. To address this, EMTOS is used, not only for further optimization but also to specifically correct the trajectory to ensure it reaches the desired orbit under the influence of this new model.

The results obtained from the SLLSQP optimization are presented in Table IV. The initial lunar orbit reached with the result from seed 30 (First S30) is not the intended one due to the additional perturbations, particularly concerning the semi-major axis, which is nearly double. The optimization with EMTOS not only corrects the final orbit, but also reduces the required ΔV, in this case by approximately 50 m/s. This demonstrates that the second stage of the software can improve the solution while simultaneously correcting the trajectory to attain the desired orbit in the new dynamics system.

	Limits	First S30	Last S30
a [km]	1836.4 to 1838.4	3222.62	1836.40
e [-]	0.000 to 0.001	0.445	0.001
i [deg]	84 to 86	88.46	84.09
Ω [deg]	-10 to 10	-0.86	-1.28
Fit [m/s]		4773.57	4716.62

TABLE IV
VALUE OF THE KEPLERIAN ELEMENTS FOR THE FINAL LUNAR ORBIT BEFORE (FIRST) AND AFTER (LAST) OPTIMIZATION FOR THE BEST SOLUTION OF SEED 30 FOR TEST CASE 1.

Figure 7 depicts the achieved orbit before and after EMTOS. The orbits before are non-circular and have a significantly larger semi-major axis than the required one. However, after optimizing the solution for each seed, the orbits obtained for all cases are identical. Furthermore, it is noticeable that the initial orbit reached from the solution for S50 exhibits an opposite argument of perilune compared to the others.

E. *Test Case 2 and 3*

Test cases 2 and 3 (TC2 and TC3) analyze two transfers to a lunar orbit designed for polar coverage. The first one includes the transfer from a LEO and the second one from a GTO, both aiming at M2, an eccentric inclined lunar orbit. Comparing both test cases can give a good estimate of the propellant that can be saved by including the satellite as piggybacking to a mission to GEO.

For TC2, more generations and a higher population size were required to reach convergence, hence highlighting the significance of the optimizer's parameters and result analysis, as the tuning was done for general cases, with particular emphasis on minimizing computational time. The particle with lowest fitness value

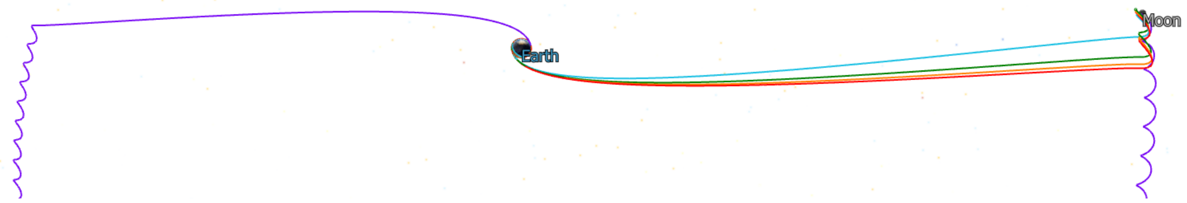

Fig. 6. Trajectories obtained from PSO optimization for test case 1. The seeds corresponding to each color are: blue for seed 10, orange for seed 20, green for seed 30, red for seed 40 and purple seed 50.

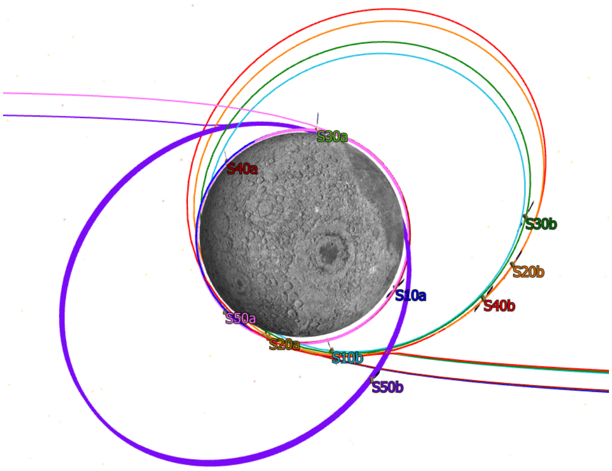

Fig. 7. Final orbits around the Moon before (b) and after (a) the SLLSQP optimization.

obtained after convergence requires a ΔV of 3895.38 m/s. TC3, on the other hand, converged smoothly with the predetermined parameters, and reached an optimum particle requiring a ΔV of 1523.16 m/s. Clearly, launching from a GTO instead of a LEO reduces the required ΔV by 2336.22 m/s, representing a decrease of approximately 60% in ΔV requirements.

Refining and optimizing these solutions using the SLLSQP optimizer yielded further improvements. In this test case, the SLLSQP optimizer effectively addresses additional perturbations and reduced the fitness, resulting in a total ΔV of 3859.81 m/s for a transfer from a LEO and 1512.95 m/s for a transfer from a GTO.

Another distinction between transfers from LEO compared to GTO is that, when propagating the solution obtained from the PSO with the full dynamics model, the lunar orbit obtained significantly deviates from the required one, especially in terms of the semi-major axis, which nearly doubles its intended value. This disparity is attributed to the higher influence of Earth's spherical harmonics near its surface compared to initial orbits with higher altitudes.

Another conclusion found in these test cases is that optimal trajectories aim to minimize inclination changes from the initial Earth orbit to the LTOs. However, reducing the inclination change for the lunar injection maneuver does not impact the total ΔV as significantly as the inclination change required to reach the LTO, due to the greater influence of Earth's gravity field.

Furthermore, analysis of the TC3 trajectory reveals that the optimal first maneuver occurs near the GTO perigee, aligning with the satellite's highest velocity point. For orbits with eccentricity and inclination, the initial orbit's geometry relative to the desired lunar orbit plays a crucial role in minimizing ΔV. Therefore, optimizing ω and Ω of the initial Earth orbit could yield further improvements in the results.

IV. Conclusions

The primary objective of this study was develop a hybrid optimization method and demonstrate that it can be used to obtain an optimal solution for a direct two-impulsive transfer under the influence of a full dynamics model.

The hybrid method involved using PSO within the SEMpy environment to conduct a global search with a simplified dynamics model. To determine the required maneuvers, Lambert's problem was solved, and a multiple-shooting method was used to consider the Moon's point-mass gravity field. Additionally, it was determined that including Earth's spherical harmonics during the satellite's trajectory propagation in the initial orbit is necessary to obtain a suitable initial estimate, enabling the gradient-based SLLSQP optimizer of the next step to rectify the trajectory under the newly chosen dynamics model. The SLLSQP algorithm not only refines the solutions obtained by PSO but also corrects for additional user-specified perturbations that could not be directly accounted for during the solution generation in PSO.

The key conclusion drawn is that this approach is feasible, and the SLLSQP optimizer not only enhances optimization further but also corrects for the newly introduced perturbations. Moreover, using a PSO algorithm to explore the defined solution space has proven to be the most effective and straightforward method of generating the initial guess. However, it was discovered that both optimizers required tuning to ensure robustness.

Throughout the process, additional conclusions were drawn. Firstly, it highlighted the importance of analyzing the data obtained for each method, as further optimization parameter tuning may be necessary, or suboptimal solutions may be found, thus showing the possibility of various interesting transfers. Secondly, it revealed that while the main parameters to optimize were the epoch of the transfer and the timing and positioning of the two maneuvers, as they have the greatest impact on the required ΔV, there are other parameters worth analyzing, such as the relative position of the initial orbit with respect to the lunar orbit, represented by ω and Ω. Thirdly, the process emphasized the significance of parameter tuning in optimization algorithms.

Lastly, both parts of the optimization procedure are independent, making it a versatile and valuable tool for primary optimization, both with and without perturbations.

References

[1] McMillan, G. L. Small satellite earth-to-moon direct transfer trajectories using the CR3BP. Missouri University of Science and Technology. 2019.

[2] Tatay, J. Optimal Trajectories in the Earth-Moon CR3BP system. MSc thesis. Delft University of Technology. 2019.

[3] Biesbroek, R., Janin, G. (2000). Ways to the Moon. *ESA bulletin, 103*, 92-99.

[4] Miller, J. K., Belbruno, E. A. (1991). A method for the construction of a lunar transfer trajectory using ballistic capture. *Spaceflight mechanics 1991*, 97-109.

[5] Blazquez, Emmanuel Thibault, Gateau Guardabasso, Paolo Fossà, Alberto Lizy-Destrez, Stéphanie. (2021). SEMpy: a Python Open-Source Toolbox for Non-Keplerian Astrodynamics. ISAE-SUPAERO.

[6] Pan, V., Reif, J. (1985). *Fast Efficient Algorithms for Linear Programming and for the Linear Least Squares Problem* (p. 0017). Center for Research in Computing Techn., Aiken Computation Laboratory.

[7] Zhang, X., Du, Y., Qin, Z., Qin, G., Lu, J. (2005). A modified particle swarm optimizer for tracking dynamic systems. In *Advances in Natural Computation: First International Conference, ICNC 2005, Changsha, China, August 27-29, 2005, Proceedings, Part III 1* (pp. 592-601). Springer Berlin Heidelberg

[8] Chobotov, V. A. (Ed.). (2002). Orbital mechanics. 3rd. Reston, VA: AIAA Education Series.

[9] Gupta, M. (2020). Finding order in chaos: Resonant orbits and Poincaré sections (Doctoral dissertation, Purdue University).

[10] Innocente, M. S., Sienz, J. (2011, June). Particle swarm optimization with inertia weight and constriction factor. *In Proceedings of the International conference on swarm intelligence (ICSI11)*, Chongqing China.

[11] Marini, F., Walczak, B. (2015). Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems, 149*, 153-165.