

Fuel Optimal, Finite Thrust Guidance Methods to Circumnavigate with Lighting Constraints*

Eric R. Prince, Ryan W. Carr, and Richard G. Cobb

Air Force Institute of Technology, Wright-Patterson AFB, Ohio

ABSTRACT

This paper details improvements made to the authors' most recent work to find fuel optimal, finite-thrust guidance to inject an inspector satellite into a prescribed natural motion circumnavigation (NMC) orbit about a resident space object (RSO) in geosynchronous orbit (GEO). Better initial guess methodologies are developed for the low-fidelity model nonlinear programming problem (NLP) solver to include using Clohessy-Wiltshire (CW) targeting, a modified particle swarm optimization (PSO), and MATLAB's genetic algorithm (GA). These initial guess solutions may then be fed into the NLP solver as an initial guess, where a different NLP solver, IPOPT, is used. Celestial lighting constraints are taken into account in addition to the sunlight constraint, ensuring that the resulting NMC also adheres to Moon and Earth lighting constraints. The guidance is initially calculated given a fixed final time, and then solutions are also calculated for fixed final times before and after the original fixed final time, allowing mission planners to choose the lowest-cost solution in the resulting range which satisfies all constraints. The developed algorithms provide computationally fast and highly reliable methods for determining fuel optimal guidance for NMC injections while also adhering to multiple lighting constraints.

1. INTRODUCTION

With the increasing number of objects in geosynchronous orbit (GEO), space situational awareness (SSA) of the GEO belt is a necessity. SSA efforts can be divided into ground-based and space-based techniques, where space-based techniques may be advantageous due to the close proximity to objects of interest and the ability to garner coverage of areas which may be denied to ground-based sensors [1]. Thus, this paper focuses on a space-based SSA technique where an inspector satellite operates in the proximity of a resident space object (RSO) in GEO. Due to the number of objects in GEO, one inspector satellite may have the capability to inspect many objects over one mission lifetime [2], especially if the guidance implemented is fuel optimal.

This paper is a continuation of the authors' most recent work in [3] where methods were developed to find fuel optimal guidance to inject an inspector satellite into a natural motion circumnavigation (NMC) orbit about an RSO in GEO. The methods developed in [3] accounted for finite thrust and also adhered to one of two sunlight constraints. It was also shown that the initial guesses supplied to the NLP were not as reliable as desired, and only led to convergence some of the time. The goals of the study herein are to advance the methods developed in [3] by accomplishing the following: 1) Improve the reliability of the developed algorithms; 2) Include additional types of lighting constraints; 3) Implement a different nonlinear programming problem (NLP) solver, and 4) provide mission planners with a range of optimal solutions.

Thus, the paper is structured as follows. First, a short summary of the methods developed in [3] is provided. Second, different initial guess generation methods are presented, with the goal of increasing the reliability of the developed optimization algorithms by using Clohessy-Wiltshire (CW) targeting [4], a modified particle swarm optimization (PSO) [5] algorithm, and MATLAB's genetic algorithm (GA). Then,

*The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the U.S. Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. This document has been cleared for public release.

the sunlight constraint previously developed in [3] is summarized, and new lighting constraints for the Moon and Earth are presented. The optimization problem is then solved with IPOPT [6], using an improved initial guess and the new lighting constraints. Finally, two methods for generating a range of fuel optimal solutions are presented, where the fixed final time for the injection maneuver is varied to provide mission planners with a range of optimal solutions to choose from.

2. BACKGROUND

The low-fidelity model developed in [3] used one method, MATLAB's PSO, to generate an initial guess for the NLP solver. The solution from the PSO was fed into MATLAB's *fmincon* as the initial guess, where the *interior-point* or *sqp* algorithm was then used to refine the initial guess and find a solution which satisfied the constraints to within some tolerance. The equations of motion used for this low-fidelity model were the Hill-Clohessy-Wiltshire (HCW) dynamics [4]. An analytical solution [7, 3] for the final six HCW states given six optimization variables for a burn-coast-burn sequence was used in the optimization formulation to determine the three-dimensional direction of the two burns and durations of the two burns to minimize the total time of the two burns and thus minimize fuel usage. Thus, the analytic propagation of the burn-coast-burn sequenced allowed the PSO to rapidly converge, performing 72 times faster than numerically propagating the burn-coast-burn sequence with MATLAB's *ode45* [7]. This analytic propagation also sped up the performance of the NLP solver, where analytic derivatives were also calculated and supplied to improve convergence [3].

In this study, three new methods are used to produce a better initial guess to improve the probability of convergence of the NLP solver. The first is CW targeting [4], which uses the HCW state transition matrix, Φ , a given maneuver time, t_f , and a desired or target position and velocity, p_t and v_t respectively, to find the ΔV magnitude and direction for the two burns in the burn-coast-burn sequence. If the 6x6 HCW state transition matrix Φ is split up into four 3x3 matrices, Φ_{11} , Φ_{12} , Φ_{21} , and Φ_{22} , then the first ΔV_1 is calculated by:

$$\Delta V_1 = \Phi_{12}^{-1} (p_t - \Phi_{11}p_0) - v_0^- = v_0^+ - v_0^- \quad (1)$$

where the subscript 0 represents the initial conditions at the beginning of the maneuver, and the superscripts $-$ and $+$ represent the values before and after the instantaneous ΔV . The inverse of Φ_{12} should exist for all cases except for when the following equations equal zero:

$$8 \cos(t_f\omega) + 3t_f\omega \sin(t_f\omega) - 8 = 0 \quad (2)$$

$$\sin(t_f\omega) = 0 \quad (3)$$

where ω is the constant mean motion of the RSO. Thus, at and near every $t_f\omega = n2\pi$ (where $n \in \mathcal{Z}_{\geq 0}$), and at or near the other zeros of the first equation (which only occur beyond 2π) will have singularities. The second ΔV is then calculated by:

$$\Delta V_2 = v_t - (\Phi_{21}p_0 + \Phi_{22}v_0^+). \quad (4)$$

The second initial guess method used in this study is a modified PSO [5]. In [3], MATLAB's PSO was used, which implemented an adaptive neighborhood size and inertia weight for the particles at each iteration. When a component of a particle's position vector violated an upper or lower bound, that optimization variable was moved back to the bound, and the corresponding component in the velocity vector was set to zero. This version of the PSO, used as an initial guess in [3], increased the chance of successful convergence of the NLP by approximately 40% over using an initial guess generated uniformly from a grid, although the exact results depended on the type of sunlight constraint enforced. The study herein uses this PSO but makes a slight modification to improve the performance, as described in the next section.

The third initial guess method used in this study is MATLAB's GA which contains built-in techniques to handle nonlinear constraints, unlike MATLAB's PSO where it can be difficult to enforce constraints. The two options for handling nonlinear constraints are the *Augmented Lagrangian Genetic Algorithm* and the *Penalty Algorithm* where the former is the default. According to MATLAB documentation pages, with the *Penalty Algorithm*, MATLAB's GA solves the problem by first attempting to create a feasible GA population with

respect to all constraints via its *fmincon* algorithm by starting from a variety of initial points from within the bounds. It automatically uses the tournament selection type, and then proceeds with the normal algorithm, using the penalty function as the fitness measure. This means that if an individual in the population is feasible, then the penalty function is the fitness function. If an individual is infeasible, then the penalty function is the maximum fitness function from the feasible individuals in the population plus the sum of the constraint violations of the current individual.

In [3], two types of sunlight constraints were presented, termed hard and soft lighting constraints. The hard sunlight constraint ensured that upon injection into the NMC about the RSO, the inspector satellite lined up exactly with the Sun vector projected into the orbital plane of the RSO such that the RSO remained lit with respect to the inspector satellite throughout the NMC, at least over the course of several days. The optimization variables for the hard sunlight constraint are:

$$X = [\alpha_1, \phi_1, t_{cf}, \alpha_2, \phi_2, t_{2f}] \quad (5)$$

where α_1 , ϕ_1 , α_2 , and ϕ_2 are the in-plane and out-of-plane directions for both burns, t_{2f} is the fraction of the total maneuver time, t_f , at which burn 2 starts, and t_{cf} is the fraction of that time at which coasting starts.

The soft sunlight constraint allowed an angular margin on both sides of the projected Sun vector, thus enabling the injection maneuver to save more fuel by allowing the inspector satellite to enter the NMC at such a location that satisfactory sunlight angles existed throughout the NMC. The optimization variables for this approach contain one extra variable, β , which is one of Lovell's relative orbital elements (LROEs)[8], and allows the angle at which the inspector satellite enters the NMC to shift in the given margin:

$$X = [\alpha_1, \phi_1, t_{cf}, \alpha_2, \phi_2, t_{2f}, \beta]. \quad (6)$$

According to the authors' literature search, optimization methods where sunlight constraints are incorporated to ensure proper lighting exists throughout the NMC have not been previously described, nor has the incorporation of Moon and Earth lighting constraints which will be developed in the following section.

3. METHODOLOGY

3.1. Initial Guess Improvements

The first method used to improve the initial guess for the NLP solver is CW Targeting. Once ΔV_1 and ΔV_2 are calculated, the initial guess is then produced by using the relationship

$$Ft_{dur} = m\|\Delta V\|_2 \quad (7)$$

where the force, F , imparted by the engine in the direction of the burn and the mass of the satellite, m , are assumed to be constant for the duration of the burn, t_{dur} . This may be a suitable assumption for a low-thrust, on/off type thruster. Thus, using an estimate for F and m , or $a_0 = \frac{F}{m}$, the duration of the burn is estimated to be

$$t_{dur} = \frac{\|\Delta V\|_2}{a_0} \quad (8)$$

for both burns 1 and 2 in the burn-coast-burn sequence. Then, the initial guess for the NLP solver can be generated, where the optimization variables in Equation 5 can be calculated from the vectors ΔV_1 , ΔV_2 , and from t_{dur_1} and t_{dur_2} for the hard sunlight constraint. For the soft sunlight constraint, the initial guess for β is calculated using the target states and the LROEs [8].

The second method used to improve the initial guess for the NLP solver is a modified PSO. The modified PSO written by the authors follows MATLAB's PSO, except for the logic used when an element of a particle's position vector reaches a bound. Instead of bringing all variables back to the bound they violated, the first four angle variables, i.e. α_1 , ϕ_1 , α_2 , and ϕ_2 , are transported to the other bound and the velocity remains as it was, since the angles are periodic in nature.

Finally, the third method used to improve the initial guess for the NLP solver is MATLAB's GA as described in the previous section. The default method used to handle nonlinear constraints, *Augmented*

Lagrangian Genetic Algorithm, did not perform well, and thus the non-default option, *Penalty Algorithm*, was chosen.

The performance of these methods as the initial guess for the NLP is analyzed in the same way as the original PSO was analyzed in [3] for both the hard and soft sunlight constraints with results presented in the next section.

3.2. Earth and Moon Lighting Constraints

An inspector satellite may be equipped with a sensor which needs to be pointed at the RSO throughout the NMC. It is thus desirable that the view of the RSO be unobscured by any bright objects behind it and in the field of view of the sensor. The previous study, [3], only accounted for light from the Sun. The study herein develops methods to ensure the Earth and Moon remain at a prescribed angle away from the sensor boresight vector, to ensure they do not appear in the background during the NMC.

The Earth light is excluded from the background of the RSO by the design of the NMC. Similar to the NMC design in [9], an exclusion cone with a half cone angle θ is created such that the sensor boresight vector avoids the Earth in addition to the prescribed exclusion angle for the Earth, θ_E :

$$\theta = \tan^{-1} \left(\frac{R_E}{2a} \right) + \theta_E \quad (9)$$

where R_E is the radius of the Earth and a is the semimajor axis of the RSO. With this half cone angle, the LROE z_{max} , which is the maximum out-of-plane amplification, can be calculated as

$$z_{max} = \frac{a_e}{2} \tan(\theta) \quad (10)$$

where a_e is the prescribed size of the NMC and is one of the LROEs. The LROE γ , which is the constant phase difference between the in-plane and out-of-plane motion, must be

$$\gamma = \pm 90^\circ. \quad (11)$$

These LROE values of z_{max} and γ ensure the Earth is excluded from the background throughout the NMC of the inspector satellite, and are incorporated into the terminal equality constraints.

In order to keep the Moon excluded from the background, the strategy developed in this study is to project the Moon vector into the orbital plane of the RSO (the radial, in-track plane of the RSO LVLH frame), as was done with the Sun vector, and check for cases where the in-plane angle to the projected Moon vector, α_m , is within the Moon exclusion angle, θ_M , from the expected in-plane angle to the inspector satellite (plus 180°), α_d , throughout the NMC. In order to do this, the first step is to use the Julian Date at the final time of the maneuver to calculate the Moon vector using Vallado's algorithm 31, *Moon* [10], following which the vector from the RSO to the Moon is calculated in the same fashion as was done in [3] for the Sun vector. The angle to this vector projected into the orbital frame of the RSO, $\alpha_m(t)$, is then calculated over the course of one period, P , (one trip around the NMC). Using the LROEs, the in-plane angle to the inspector satellite (plus 180°), $\alpha_d(t)$, is also calculated over the course of one period, and the difference between the two angles is calculated. The minimum difference between the two angles is extracted, α_{min} , and then used as follows: For the hard lighting constraint, if $\alpha_{min} < \theta_M$, then the desired NMC entry state, X_t is shifted in the smallest direction away from the projected Sun vector to a position on the NMC such that the new $\alpha_d(t)$ creates a new α_{min} which doesn't violate θ_M . For the soft lighting constraint, the bounds on β are changed from the original angular margin allowed from the projected Sun vector such that no β can be chosen by the optimizer which makes α_{min} violate θ_M . If this changes the bounds on β enough to where X_t is no longer within the bounds, then the CW Targeting target position and velocity are changed to the state on the new bound closest to the original X_t . The change in X_t for the hard lighting constraint and the changes on the bounds for β for the soft lighting constraint ensure that the Moon will not come within θ_M of the sensor boresight vector throughout the NMC, at least for the course of one period.

4. SIMULATION AND RESULTS

4.1. Initial Guess Improvements

To show how the new initial guess methods yield improved convergence results, the same type of analysis which was done in [3] has been performed here. Thus, the same simulation parameters are used as in [3],

Table 1: Simulation Parameters

Sun and Time	RSO Properties	Inspector Properties	NMC Parameters
$\theta_s = 45^\circ$	$a_c = 42,164.137$ km	$a_0 = 0.02$ N/kg	$a_e = 5$ km
Year ₀ = 2017	$\vec{r}_{e2c_0}^I = [a_c; 0; 0]$ km	$c = 3.33$ km/s	$y_{d_0} = 0$ km
Month ₀ = Aug	$\vec{v}_{e2c_0}^I = [0; 3.0747; 0]$ km/s	$x_0 = -20$ km	$z_{max} = 1$ km
Day ₀ = 31		$y_0 = 10$ km	$\gamma = 90^\circ$
Hr ₀ = 23		$z_0 = -5$ km	
Min ₀ = 0		$\dot{x}_0 = -1.5$ m/s	
Sec ₀ = 0		$\dot{y}_0 = 0.4$ m/s	
		$\dot{z}_0 = 1.1$ m/s	

shown in Table 1. With these parameters, the three new initial guess methodologies are used to generate an initial guess for MATLAB’s *fmincon*, which is solved five different times for each lighting constraint. It is solved three times with *interior-point*: once with no analytic derivatives supplied, once with the gradient of the objective function and Jacobian of constraints supplied, and once with the Hessian of the Lagrangian also supplied. It is solved twice with *sqp*: once with no analytic derivatives supplied, and then with the gradient of the objective function and Jacobian of the constraints supplied.

The CW Targeting method differs from the other methods in that it is deterministic. The modified PSO and the GA are stochastic in nature, and thus are analyzed in the same way as the original PSO, using the same initial conditions but running the simulation thousands of times to see how often the new methods produce a successful initial guess for the NLP solver. A t_f of 1.5 hours, as used in [3], allows comparison of the results.

The results for both the hard and soft lighting constraints where the solution from the modified PSO is given as the initial guess for the NLP solver is shown in Table 2, where the *fmincon* exit flags are defined as follows: -2, the step size is below the tolerance and constraints are not satisfied; 0, the maximum number of function evaluations or iterations has been reached; 1, a local minimum has been found; 2, the step size is below the tolerance but constraints have been satisfied.

Table 2: NLP Performance for Both Lighting Constraints Given Initial Guesses from Modified PSO

Algorithm	Supplied	Exit Flag Obtained (%)				Mean Comp Time (s) for Exit Flag			
Hard Lighting Constraint		-2	0	1	2	-2	0	1	2
int-pt	None	17.99	3.98	0.24	77.78	0.2551	3.8062	0.1550	0.0182
int-pt	Gradients	18.65	3.44	0.39	77.51	0.4198	11.1668	0.0429	0.0157
int-pt	Hessian Also	18.70	3.49	1.12	76.68	0.5099	13.2954	0.0580	0.0183
sqp	None	17.80	0.17	82.03	0	0.0303	0.6533	0.0078	NA
sqp	Gradients	17.63	0.15	82.23	0	0.0300	0.6596	0.0050	NA
Soft Lighting Constraint		-2	0	1	2	-2	0	1	2
int-pt	None	3.57	0.27	75.08	21.08	0.3547	3.4614	0.5597	0.6749
int-pt	Gradients	2.38	0.27	94.56	2.79	1.1508	11.5545	0.3468	0.1680
int-pt	Hessian Also	1.87	0.91	63.33	33.88	1.9514	13.9828	0.2294	0.1569
sqp	None	43.16	0.32	53.91	2.61	0.0195	0.9723	0.0097	0.1371
sqp	Gradients	6.08	36.12	55.24	2.56	0.0756	0.7111	0.0064	0.0988

The analysis of these results provides some valuable information. Given the initial guesses from the Modified PSO:

- For the hard lighting constraint, *sqp* performs best.
- For the soft lighting constraint, *interior-point* performs best.
- Supplying derivative information helps in all cases where a solution is found, except for when including the Hessian also for the soft lighting constraint when using the *interior-point* method. However, by

supplying first derivative information for the soft lighting constraint when using the *interior-point* method, the chance of successful convergence increases by almost 20%.

- Given the modified PSO as an initial guess, the chance of convergence for the hard lighting constraint increases from 49.38% (from the original PSO in [3]) to 82.23%.
- Given the modified PSO as an initial guess, the chance of convergence for the soft lighting constraint increases from 76.31% (from the original PSO in [3]) to 94.56%.

Table 3: NLP Performance for Both Lighting Constraints Given Initial Guesses from MATLAB’s GA

Algorithm	Supplied	Exit Flag Obtained (%)				Mean Comp Time (s) for Exit Flag			
Hard Lighting Constraint		-2	0	1	2	-2	0	1	2
int-pt	None	0	0	98.66	1.34	NA	NA	0.0060	0.0150
int-pt	Gradients	0	0	99.85	0.15	NA	NA	0.0053	0.0129
int-pt	Hessian Also	0	0	99.85	0.15	NA	NA	0.0070	0.0172
Soft Lighting Constraint		-2	0	1	2	-2	0	1	2
int-pt	None	0	0	98.45	1.56	NA	NA	0.0677	0.1203
int-pt	Gradients	0	0	100	0	NA	NA	0.0551	NA
int-pt	Hessian Also	0	0	100	0	NA	NA	0.0629	NA
sqp	None	3.66	0	0.46	95.88	0.0101	NA	0.0060	0.0049
sqp	Gradients	3.34	0.59	1.01	95.06	0.0885	0.6106	0.0042	0.0117

Given the initial guesses from MATLAB’s GA (see Table 3):

- For the hard lighting constraint, *sqp* performs best, with a 100% success rate. *interior-point* is close behind, with a 99.85% success rate.
- For the soft lighting constraint, *interior-point* performs best, with a 100% success rate, and *sqp* doesn’t perform well at all.
- Supplying derivative information either helped a small amount or had no effect on achieving successful convergence.

The average computation times for each guess generation method is shown in Table 4, emphasizing how computationally fast the metaheuristic methods are, due to the analytic propagation of the burn-coast-burn sequence.

Table 4: Average Computation Times (seconds) for Metaheuristic Initial Guess Methods

Lighting Constraint	PSO	Modified PSO	GA
Hard	5.23	2.80	6.01
Soft	8.36	1.86	8.32

The CW Targeting initial guess method is deterministic compared to the other two stochastic methods and thus cannot be analyzed in the exact same manner. For the hard lighting constraint and with $t_f = 1.5$ hours, the CW Targeting initial guess provides a successful guess only for the two *sqp* methods. For the soft lighting constraint, it does not provide a successful initial guess for any of the five *fmincon* methods. This is due to the fixed final time, t_f , being too small and thus the error due to the impulsive approximation is too large. For example, if t_f is changed to 3 hours, then all three *interior-point* methods converge to a successful solution when given the CW Targeting initial guess. This makes sense, as it’s obvious that as the fixed final time increases, the required engine-on time typically decreases, and the closer the CW Targeting solution resembles the actual finite-burn solution.

To summarize this section, the initial guess methods developed in this study have improved those used in [3]. If MATLAB’s Optimization Toolbox is available, then MATLAB’s GA performs extremely well, by

using the non-default *Penalty Algorithm* method to handle the nonlinear equality constraints. If MATLAB's Optimization Toolbox is unavailable or not desirable for use, then the Modified PSO has good performance for initial guess generation, and is relatively easy to code. And, if the fixed final times of the maneuvers of interest are large enough compared to the actual finite duration burns required to implement the calculated ΔV s, then CW Targeting is an extremely fast and simple method to produce a good initial guess.

4.2. Simulations with Multiple Lighting Constraints

This section presents simulation results where varying levels of lighting constraints are enforced for the motion in the resulting NMC. The first simulation only enforces the hard sunlight constraint, as was done in [3], with the same parameters as shown in Table 1, except for the starting Coordinated Universal Time (UTC) date is changed to September 5th. The solution is obtained by using CW Targeting as the initial guess and then IPOPT [6] as the NLP solver, where IPOPT has been chosen for implementation due to its good reputation as an NLP solver and the fact that it is open-source software. For this case, IPOPT successfully converges to a solution unlike *fmincon*, with the results seen in Figure 1, where the projected sunlight vectors from the Sun to the RSO, \vec{r}_{s2c} , are shown at the beginning of the maneuver, t_0 , the final time of the maneuver, t_f , and at one-fourth and one-half of the RSO period, P .

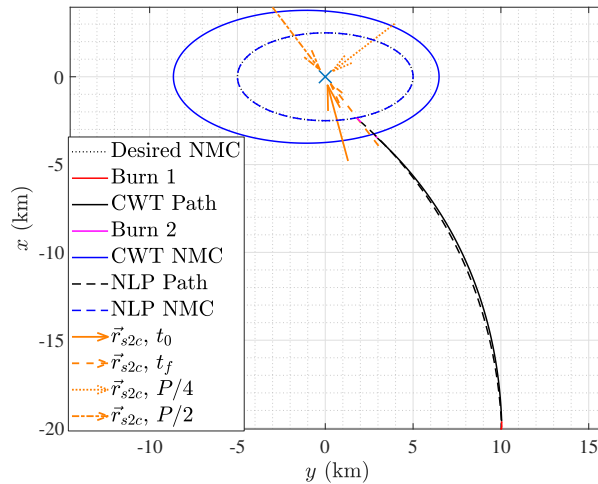


Figure 1: CWT and IPOPT Solutions, Hard Lighting Constraint, $t_f = 1.5$ hrs

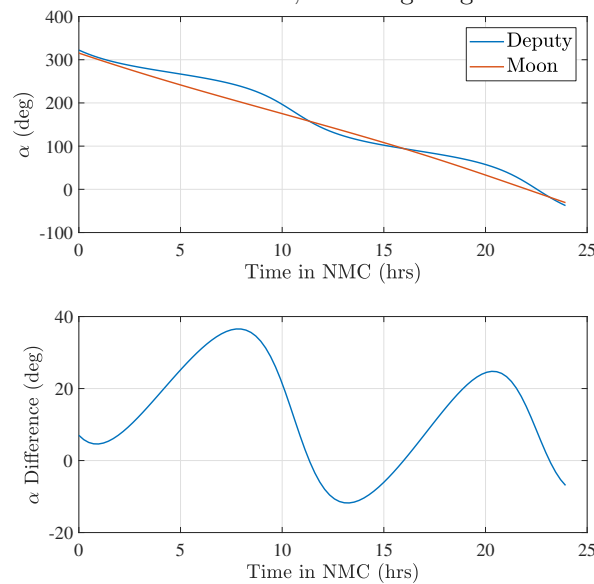


Figure 2: Moon Conflict Scenario, $t_f = 1.5$ hrs

For the new UTC date of September 5th, the angle to the projected Moon vector, α_m , comes within θ_M of α_d , and even crosses α_d , as seen in Figure 2. This means that if the inspector satellite enters the NMC at the original X_t (aligned with the Sun vector), then there is a good chance that at some point during the NMC, no matter the NMC cross-track properties, the Moon will appear in the background, within θ_M of the sensor boresight vector. Thus, for this scenario, the minimum shift for α_d must be 21.78° , which is the minimum angle α_m must be shifted to not cross α_d ($\alpha_{min} = 11.78^\circ$), plus θ_M (10°). This means that the inspector must enter the NMC at a shifted X_t which puts the inspector satellite behind the Sun vector at the time of entry, as seen in Figure 3 (a), as compared to Figure 1.

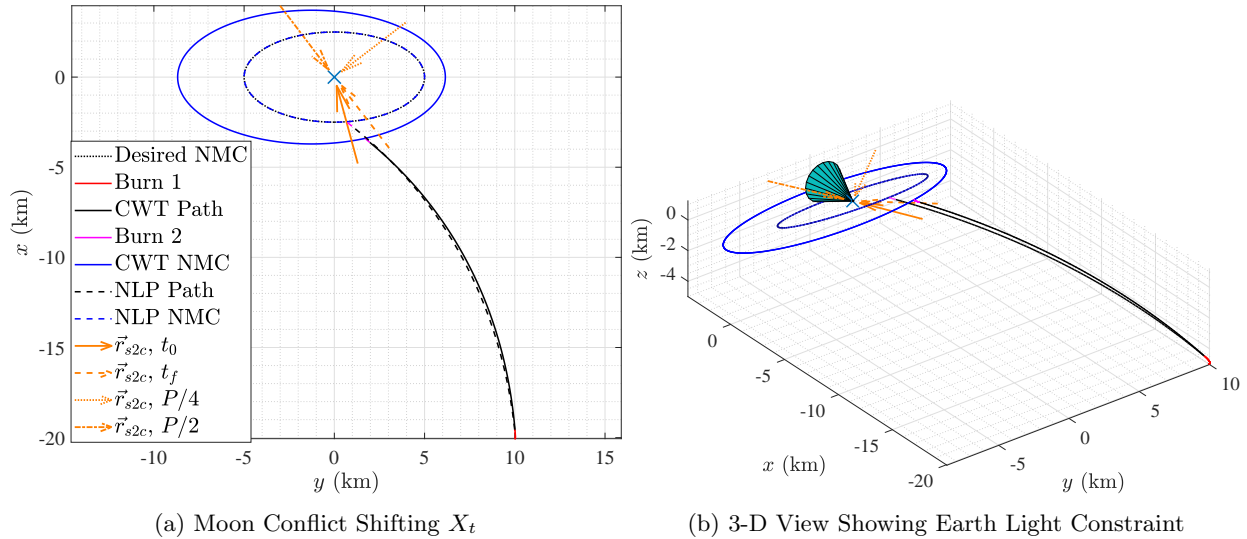


Figure 3: Moon and Earth Light Constraints Active, $t_f = 1.5$ hrs

The Earth lighting constraint is enforced now as well, by prescribing the appropriate LROEs z_{max} and γ as given by Equations 10-11. The result can be seen in Figure 3 (b), where the exclusion cone shows how the NMC must be designed such that the Earth does not appear in the background of the RSO during the NMC.

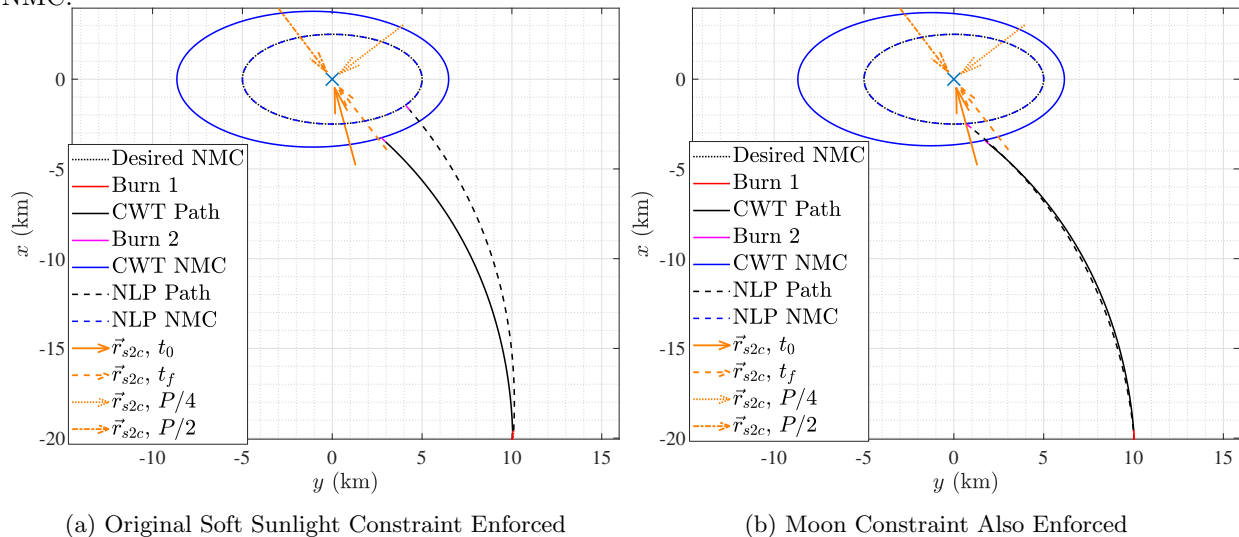


Figure 4: Soft Sunlight Constraint, Before and After Moon Constraint Enforced, $t_f = 1.5$ hrs

The same parameters are used again but this time the soft lighting constraint is enforced. An example solution from [3] is shown in Figure 4 (a), where the entry point has shifted from X_t within $\pm\theta_s$ to further minimize fuel usage. Now, the Moon constraint is enforced, where the new, shifted X_t from the hard sunlight constraint example now becomes one of the new bounds for β . Since in this example X_t was shifted behind

the original X_t , this puts a new upper bound on β . Thus, the original lower bound is unchanged, and the new upper bound ensures that the inspector satellite will not be in an NMC where the Moon may appear in the background. This can be seen in Figure 4 (b), where the entry point into the NMC has shifted to the optimal entry point in the new allowable range for β , which happens to be at the new X_t , due to the Moon constraint being enforced.

4.3. Range of Optimal Solutions

Mission planners may desire a range of optimal solutions based on varying fixed final times in order to see the trade-offs for how much fuel could be saved by changing the fixed final time of the maneuver. Typically, this is a monotonically decreasing curve, where the amount of fuel required decreases as the fixed final time increases. However, with the Sun, Moon, and Earth lighting constraints, this may not always be the case. Thus, two methods have been developed to find a range of solutions with varying fixed final times for the maneuver. The first is to use CW Targeting as the initial guess for the varying fixed final times, and use IPOPT to find the solution for every fixed final time. The second is to use a homotopic approach, where the IPOPT solution to the original fixed final time problem is used as the initial guess for a nearby, new fixed final time problem, and so on. An example range of solutions has been found for $t_f \in [0.25, 13]$ hours with the soft sunlight and Earth constraints active. Figure 5 (a) shows the range of optimal solutions where the first approach is used, and Figure 5 (b) for the second approach. It is interesting to note that the CW Targeting approach provides better initial guesses for IPOPT for decreasing fixed final times compared to the homotopy approach. It is also interesting to note that the CW Targeting approach has some convergence issues around $t_f = 12$ hours, but regains convergence afterwards, whereas the homotopic approach successfully converges, but follows a local minimum to very high engine-on times as the fixed final time increases. Figure 6 shows the region of interest where the detected minimum lies, where a fixed final time of about 8.5 hours produces the lowest engine-on time solution of about 255 seconds for this scenario.

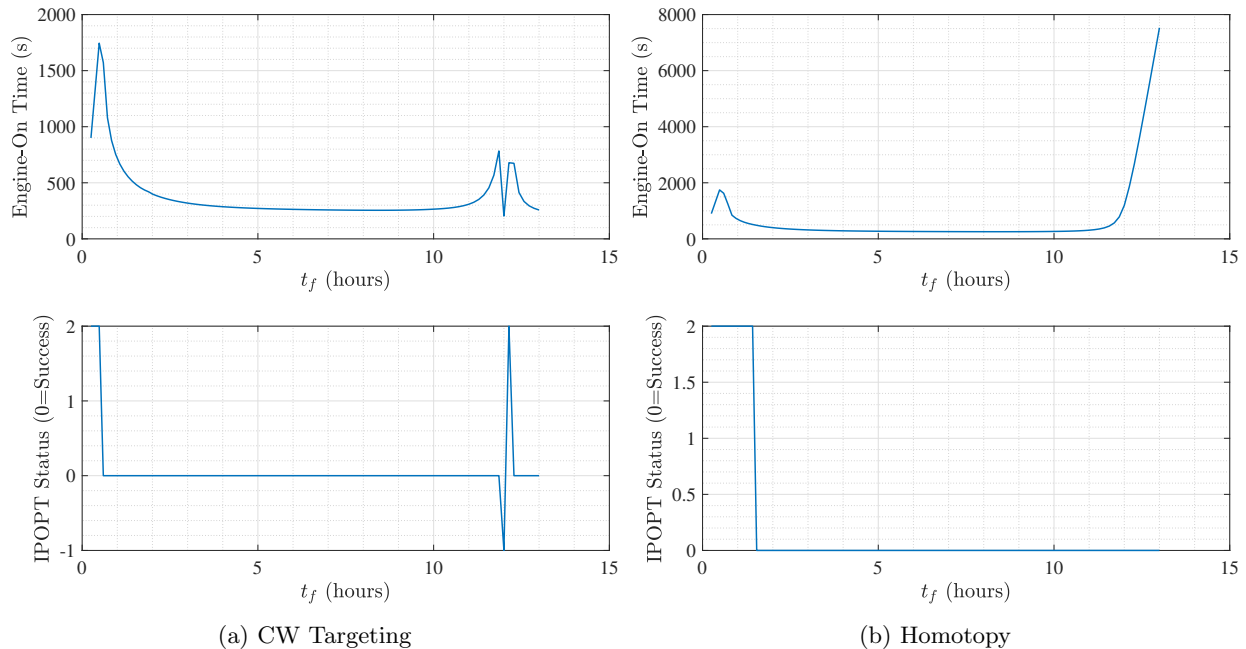


Figure 5: Soft Sunlight Constraint, Engine-On Time vs. IPOPT Status

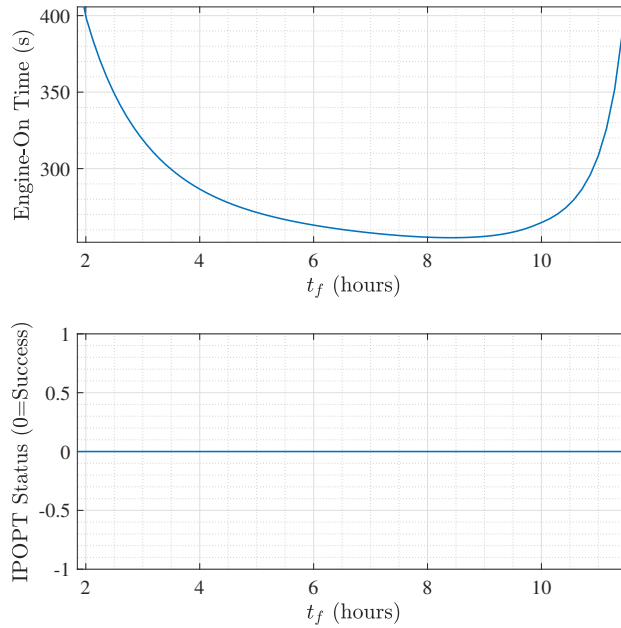


Figure 6: Soft Sunlight Constraint, Engine-On Time vs. IPOPT Status, Zoomed (2-11 hours)

CONCLUSION

Fuel optimal, finite thrust guidance methods have been further improved in this study, where better initial guess generation methods have been shown to increase the chance of an NLP to successfully converge for the burn-coast-burn nonlinear optimization problem. Specifically, one deterministic initial guess method, CW Targeting, works very well for fixed final times large enough compared to the finite duration of the burns required to produce the calculated ΔV s. Also, two metaheuristic methods, a modified PSO and MATLAB's GA, have been shown to be very reliable initial guess methods, with chance of convergence at 94.56% or better for the soft sunlight constraint. The computation times are also extremely fast, taking advantage of the analytic propagation of the finite burn-coast-burn sequence, where the modified PSO has a mean computation time of 1.86 seconds for the scenarios explored herein. It has been shown that an *interior-point* type method can be used for the NLP solver with high reliability for all cases except for when producing an initial guess with the modified PSO for the hard sunlight constraint scenario, where an *sqp* type solver should be used. Additional lighting constraints for an inspector satellite to circumnavigate an RSO in GEO have also been implemented, where the NMC can be designed and the entry point shifted such that neither the Moon nor the Earth enter the background of the RSO with respect to the inspector satellite. Two methods have also been developed to produce a range of optimal solutions with the Sun and Earth lighting constraints active, providing mission planners with options for the fixed final time.

REFERENCES

- [1] US Joint Chiefs of Staff, "Space Operations," 2013.
- [2] McCall, G. H. and Darrah, J. H., "Space Situational Awareness," *Air & Space Power Journal*, , No. November-December, 2014, pp. 6–16.
- [3] Prince, E. R., Carr, R. W., and Cobb, R. G., "Computationally Efficient Methods for Fuel Optimal Proximity Maneuvers with Constraints," *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, WA, 2017.
- [4] Wiltshire, R. S. and Clohessy, W. H., "Terminal Guidance System for Satellite Rendezvous," *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653–658,674.
- [5] Kennedy, J. and Eberhart, R., "Particle swarm optimization," *Proceedings of IEEE International Conference on Neural Networks*, Vol. 4, 1995, pp. 1942–1948.
- [6] Wächter, 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, No. 1, 2006, pp. 25–57.
- [7] Prince, E. R. and Cobb, R. G., "Optimal Inspector Satellite Guidance To Quasi-Hover Via Relative Teardrop Trajectories," *International Workshop on Spacecraft Constellations and Formation Flight*, Boulder, CO, 2017, pp. 1–20.
- [8] Lovell, T. A. and Tragesser, S., "Guidance for Relative Motion of Low Earth Orbit Spacecraft Based on Relative Orbit Elements," *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Providence, 2004, pp. 1–16.
- [9] Prince, E. R., Cobb, R. G., and Hess, J. A., "Optimal Slew-Rate-Limited Guidance for Combined Formation Establishment and Reconfiguration of Inspector Satellite With Exclusion Cone," *27th AAS/AIAA Space Flight Mechanics Meeting*, San Antonio, 2017.
- [10] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Microcosm Press, Hawthorne, 3rd ed., 2007.