Capacity-based Cislunar Space Domain Awareness Architecture Optimization

Naomi Owens Fahrner
Ball Aerospace
Jeremy Correa
Ball Aerospace

Joshua Wysack
Ball Aerospace

ABSTRACT

Cislunar space is generally defined as the volume of space between the earth and the moon. This encompasses a radius of nearly 385,000 km. There are an enumerable number of possible sensor architectures that might be constructed to search this space. When constructing architectures, one is typically interested in maximizing observability. The task of scheduling sensors to optimize capacity has previously been considered a separate problem from architecture design. While these problems may be tackled separately, looking at these problems together provides a powerful tool for cislunar space domain awareness (SDA). The solutions of yesterday are not sufficient to tackle the problems of today. It is no longer enough to only consider observability in architecture design. In this paper, we examine optimized architectures first, then combine this with optimized schedules to concurrently maximize capacity and observability. We will use variations of greedy and genetic algorithms to accomplish this.

1. INTRODUCTION

As traffic in cislunar space increases, the need for Space Domain Awareness (SDA) in this area greatly increases. This search space is vast, and the resources in which to observe and track objects in this space are limited. Thus, the allocation of these resources, as well as the architecture of these resources, must be optimized. We will present capacity and observability analysis that offers optimized architectures to provide meaningful SDA in cislunar space.

Architecture assessments for cislunar SDA often stop at observation analysis. These analyses do not include the sensors agility (FOV, sweep rate, etc.). These studies aim to find the 'best' sensor performance required to observe a cislunar volume for SDA or Space Traffic Management (STM) [11]. The sensor models used in this study account for various higher-order effects such as jitter, straylight, celestial background, and other optical losses. The inclusion of such effects is necessary to provide realistic performance assessments of architectures.

Additionally, most current architecture studies do not consider the sensor scheduling component [5] [15]. The authors in [5] compare various architectures with a Monte Carlo Tree Search scheduling algorithm with the goal of tracking cislunar objects. The goal in [7] is custody tracking of trajectories in cislunar space. However, we are primarily interested in cislunar volume search in this paper. To robustly accomplish this, we must both optimize observability and capacity.

We will analyze various architectures to first maximize observability. Observer orbits we consider will span Earth orbiting, lunar orbits, and the combined system. Candidate observer orbits will be drawn from cislunar orbit families near Lagrange points L1 and L2, XGEO orbits, and Earth-Moon cycler orbits. After this, we optimize the schedule of sensors in the best architecture to maximize capacity.

Capacity analysis includes the collaboration of multiple sensors to search a volume, subject to constraints. This improvement upon observability analysis considers sensors' sensitivities and agilities. Optimized architectures will be found by analyzing a variety of observer orbits, spanning the three-body Earth-Moon system. We will use greedy and genetic algorithm variations to optimize our scheduling goals for optimized architectures.

To optimize capacity, we will optimize sensor schedules. To create an intentional schedule of sensors, these sensors must coordinate to determine the best looking position and dwell time to reach the overall goal. Capacity analysis considers sensor sensitivity, field of view, agility, and quantity to fulfill an objective in a set amount of time.

To accomplish the capacity analysis, we will use various sensor scheduling algorithms to optimize the interoperability of multiple sensors in various orbits. Due to the high dimensionality of this problem, we present an approaches that use various optimization techniques to determine the best sensor tasking schedule. The entity that creates schedules, called a 'scheduler', will optimize the pointing directions and dwell times of all sensors in an architecture to maximize the volume observed.

Variations of genetic and greedy optimization algorithms will be considered for the scheduling algorithms. Our scheduling algorithms will be flexible to allow multiple performance metrics such as revisit time, coverage percentage, etc. An advantage to the way we use our scheduling algorithms is the coordination of multiple, heterogenous sensors. This requires balancing the tasking of sensors based on solar phase angle, obscuration due to Earth or Moon, and target priority. This sensor coordination provides an efficient and cost-effective tasking scenario.

The scheduling algorithms will be used to perform a comprehensive capacity analysis of cislunar SDA architectures. We provide a comparison of different architectures for cislunar SDA to include most favorable orbits and sensor characteristics. Ultimately, we conduct an assessment that evaluates not only the optimized allocation of sensors across orbit families, but also the best period and phasing of the chosen orbits. Our techniques allow for the creation of optimized capacity-based cislunar SDA architectures.

2. CONSIDERATIONS FOR CISLUNAR SDA ARCHITECTURES

2.1 Overview

This section will provide an overview of some of the key considerations of designing a space-based cislunar SDA architecture. The foremost consideration is the mission objective. As traffic in cislunar space increases over the next decade, the need to observe objects in this space grows. Various tasks included in this objective are maintaining position knowledge of known Resident Space Objects (RSO), identifying new RSOs, and identifying threats. These objectives highlight the dichotomy between Space Traffic Management (STM) and Space Domain Awareness (SDA).

Space Traffic Management for Earth-orbiting objects has historically been responsible for space-object catalog maintenance, collision avoidance (CA), and new object detection. The primary goal of STM is to detect, track, and retain custody of RSO. The data timeliness for STM depends on the orbit regime and its associated object density. However, this timeliness is typically on the order of hours, with routine CA updates occurring 2 to 3 times a day. Currently, the volume of space monitored spans from LEO to GEO altitudes. This paradigm changes for cislunar space. The volume suggested for cislunar STM can be defined as a geocentric sphere with a radius out to the Earth-Moon L2 location ($\sim$ 10 XGEO) [2]. Because of the relatively low density of space objects and slower relative motion, the position of objects in stable cislunar orbits can be maintained with tracking on the order of days, as opposed to hours.

Space Domain Awareness, the focus of this paper, includes many of the aspects of STM, with an additional emphasis on threat detection and responsiveness [4]. This requires more-timely monitoring of objects, with data timeliness on the order of tens of minutes for the near Earth regime. For cislunar space data timeliness can be expanded to hours. The primary consideration of SDA vs STM is to include potential object maneuverability – an object's ability to alter its projected/predicted motion from natural motion. For example, how fast could an object in cislunar space interfere with another object? The answer is ultimately dependent on distances to neighboring objects or protected orbital regimes (such as GEO). It is also limited by objects' assumed maneuverability.

This paper is focused on the SDA mission. Therefore, choices on parameters such as target volume definition will be directed by solutions for the cislunar SDA mission. Other key factors in the design of a SDA architecture are choice of sensor, metrics used for coverage analysis, definition of the coverage volume, and choice of orbits.

2.2 Cislunar Volume Definition

Given the immensity of cislunar space, we will use a threat-based definition of the volume that addresses the SDA mission. This volume is composed of two subvolumes that are fixed in the Earth-Moon rotating frame. The first volume is an Earth-centered hollow sphere with 3.4 XGEO outer radius and GEO inner radius. Here, 3.4 XGEO refers to 3.4 multiples of GEO radius [9]. This radius is the distance at which an object on a one-day Moon to Earth

trajectory is six hours from GEO altitude. Six hours was chosen as a minimum warning time for a satellite in GEO that is under possible threat.

The second subvolume considered is in the Earth-Moon corridor. The boundary of this is defined along a five-day Earth-Moon transit orbit that extends 10,000 km beyond L2. The total volume considered is shown in Figure 1. This is designed to capture objects in the Earth-Moon system that could pose a threat to Earth-orbiting spacecraft.

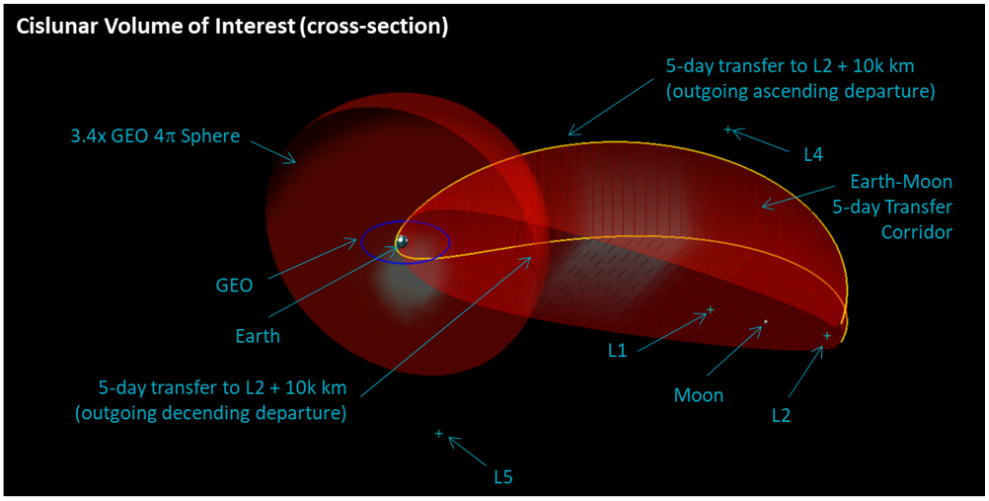


Fig. 1: Cislunar volume definition for Space Domain Awareness surveillance.

2.3 Sensor Design and Coverage Modeling

A challenge of performing SDA in cislunar space is routinely searching the volume given the great distances between objects. It is advantageous for optical space-based sensors in this regime to have high sensitivities so that fewer satellites are required to construct a meaningful architecture. An overlay of the cardioids for optical sensors of various visual magnitudes (mv) is shown in Figure 2. Clearly, the mv 20 sensor allows for coverage of a larger volume. This figure does not take into account constraints for limitations due to field of view (FOV) or sensor scheduling.

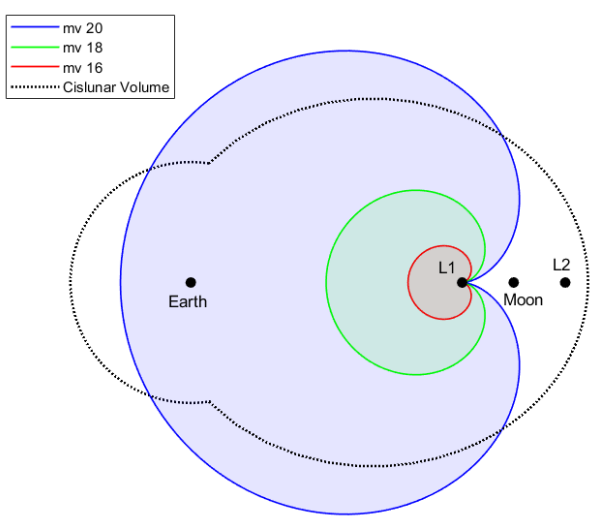


Fig. 2: Volume coverage (2D view) for various sensor sensitivities.

In this work, two methods of coverage analysis are used in architecture development for cislunar SDA. Both provide the coverage percentage of a volume of space, but differ in the level of fidelity of the result.

Observability Analysis provides the coverage percentage of a volume for one or more sensors subject to constraints like solar phase angle. Sensor FOV and agility are not considered.

Capacity Analysis expands upon observability by including sensor FOV, agility, and the collaboration of multiple sensors to provide the coverage percentage of a volume.

	Observability	**Capacity**
Key Features	• Fast runtime allows for 100k's of architecture evaluations • Provides insights into sensor performance and favorable orbits	• Combined with scheduling optimization provides realistic architecture sizing • Sensors scheduling optimized for collaborative volume search
Coverage Constraints	• Earth, Moon, Sun exclusion regions • Earth and Moon eclipses • Line of sign access	• Observability constraints + the following: • Sensor FOV • Sensor agility (slew and settle) • Dynamic dwell time • Sensor collaboration (no coverage overlap) • Range (SNR threshold using radiometric model)
Radiometric Modeling	• Visual magnitude (radiometric modeling performed with external tool)	• Visual magnitude or SNR-based using radiometric software first - third order effects (straylight, target motion, jitter, ...)

Table 1: Distinction between observability and capacity.

2.4 Orbit Families

Various three-body periodic orbits were considered for this work, including Earth-centered, Moon-centered, Lagrange point orbits, and Earth-Moon cyclers. Our intent was to model various orbit families and use coverage analysis to determine which of these orbits are best for cislunar SDA. Representative members of each orbit family are shown in Figure 3.

Orbits were modeled using the dynamics of the Circular Restricted Three Body Problem (CRTBP). In this frame, the Earth and Moon positions are fixed. Ephemeris for the various orbit families was obtained from JPL website [10]. Further orbits were developed using a single-shooting method to form orbits that are periodic under CRTBP dynamics. Using these ephemeris sources, we were able to assess the viability of many orbits from each family for cislunar SDA.

Now that we have described the general considerations for constructing cislunar SDA architectures, we will describe our process for accomplishing cislunar coverage analysis. We will start with identifying candidate architectures for cislunar SDA through the use of observability analysis. Favorable orbit families will be found. We will then use a capacity analysis to determine the best number of sensors to use. After that, we will optimize sensor pointing schedules to maximize capacity. This will provide a realistic architecture size that takes into account sensor sensitivity and agility. We will test variations of greedy and genetic algorithms to determine the best approach to further optimize coverage using these tools.

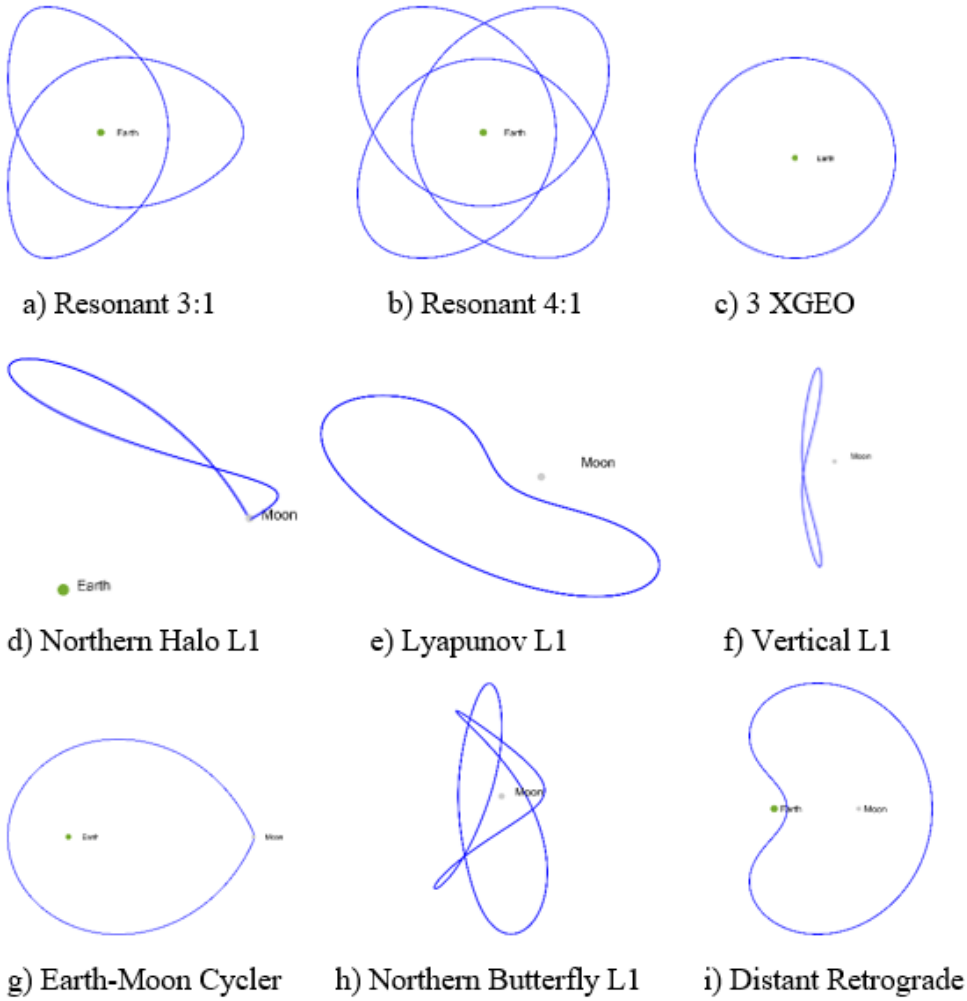


Fig. 3: Representative periodic orbits shown in the fixed Earth-Moon frame.

3. OBSERVABILITY-BASED ARCHITECTURE OPTIMIZATION

3.1 Best in Family Analysis

To find candidate architectures, we initially performed observability analysis to find favorable orbit families for cislunar SDA. This observability metric uses the range capability of the sensor based on the visual magnitude. Solar phase angle and eclipses are taken into account. Targets are modeled as 1 meter diameter Lambertian spheres with 20% reflectivity. In this work, we modeled a sensor with mv 20 sensitivity.

For each orbit family shown in Section 2, 10 to 20 orbits were equally sampled over the range of periods available from the JPL website. Observability analysis was performed for all orbits in separate runs against the two subvolumes (3.4 XGEO and cislunar corridor). This result gives insight in how a sensor in a particular orbit contributes to a full architecture.

An example of this best-in-family analysis is shown in Figure 4 for the Northern L1 Halo family. The cislunar corridor volume observability is higher for orbits that remain near the Earth-Moon plane for the majority of their period. As the out of plane component increases, the coverage of the corridor decreases.

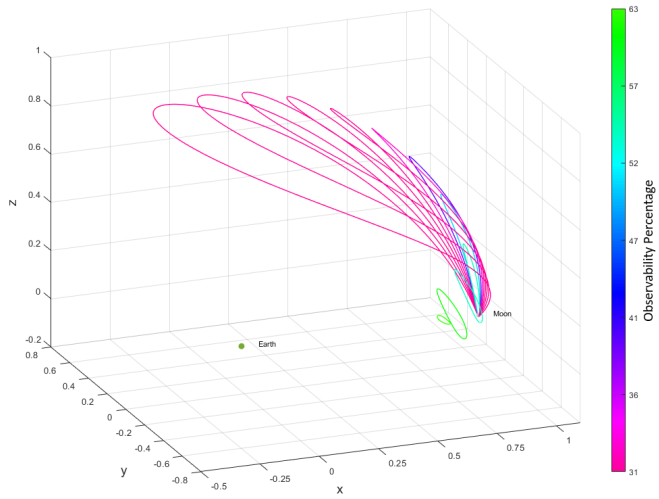
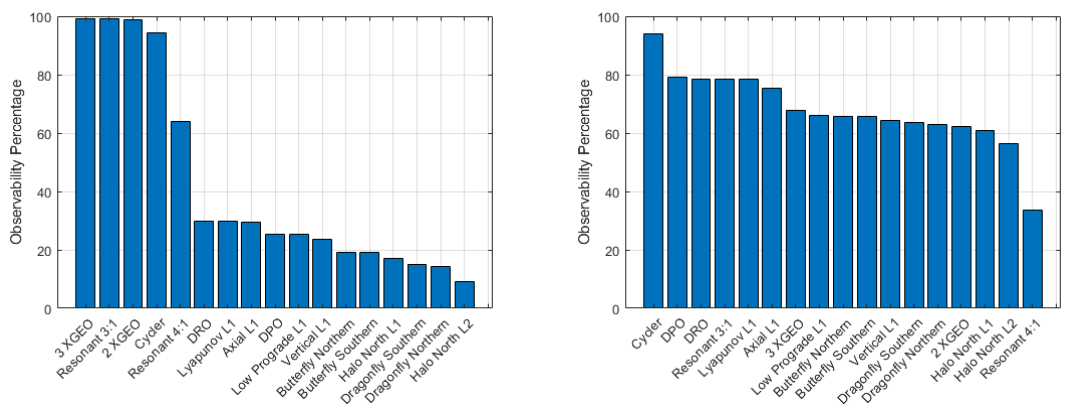


Fig. 4: Cislunar corridor observability for various Northern L1 Halo orbits.

For each family, the orbit with the highest observability results were chosen as the representative of that family. Figure 5 provides those results for both subvolumes. A key takeaway is that the resonant 3:1 orbit is amongst the top performers for each of the subvolumes.



(a) 3.4 XGEO Volume

(b) Cislunar corridor Volume

Fig. 5: Best in family observability of cislunar subvolumes for various orbits.

3.2 Candidate Cislunar SDA Architectures

Candidate architectures were constructed from the best orbits from each family. Orbits from each family were permuted to build candidate architectures. The number of sensors in each family ranged from one to six, where multiple satellites/sensors were equally distributed along the orbital path. These permutations resulted in 10k's of architectures. Observability analysis was performed against the entire cislunar volume. Results are shown in Figure 6.

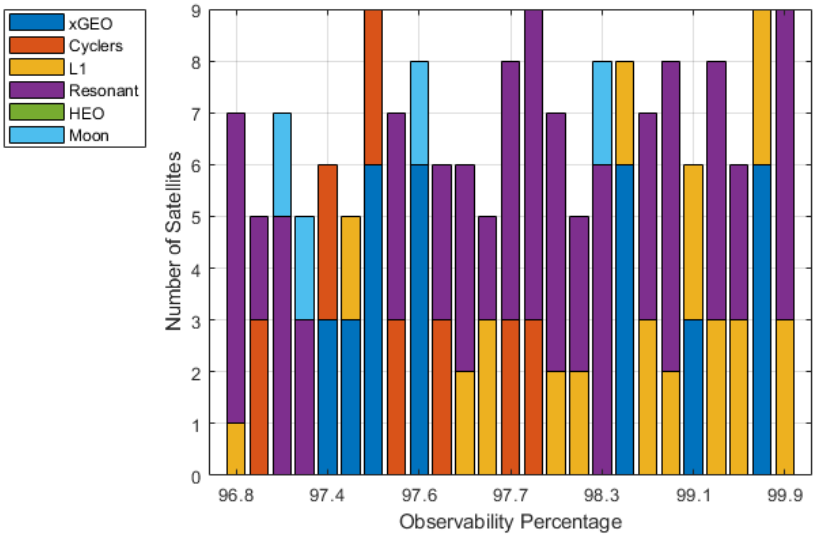


Fig. 6: Top performing architectures based on observability of cislunar volume.

Many architectures provide observability approaching 100%. Thus, results in Figure 6 are limited to architectures with a maximum of nine sensors and two orbit families. Top results have sensors in XGEO or planar resonant orbits combined with sensors in orbit about L1. In particular, the combination of resonant and L1 orbits are favorable architectures, with six of the top ten observability percentages. The resonant orbits are in a 3:1 resonance. The 3:1 resonant orbit is a good candidate due to its long-term stability. For example, that is the current orbit for the IBEX spacecraft that is able to maintain that orbit without the need of stationkeeping maneuvers [3].

The goal here was to find architectures with high capacity percentages for as few sensors as possible. In the next section we will determine how the architectures perform using capacity percentage as the coverage metric. Optimization algorithms will be presented to optimize the scheduling of the sensors to maximize coverage of the cislunar volume.

4. CAPACITY-BASED ARCHITECTURE OPTIMIZATION

4.1 Overview

Satellite task planning is the process by which a series of image collection opportunities are chosen for each satellite in a defined architecture with the intention of maximizing a mission specific capacity-based objective. A collection opportunity is typically associated with an individual satellites' observability of a region of interest. The decision making process required to command a group of satellites to realize collection opportunities is fundamental to the satellite scheduling problem. In a system where satellite motion is governed by complex non-linear dynamics and mission objectives require complex cooperative behavior, it becomes increasingly more difficult to formulate a set of behavioral rules that can result in optimal collective behavior within an architecture.

4.2 Problem Set-up

We explore the feasibility of rule-based and evolutionary optimization techniques to provide a multi-satellite solution for a space-based SDA mission in cislunar space. We have built a framework where satellite tasking is planned using cooperative behavior to maximize coverage.

4.2.1 Sensor Task Planning

A complete sensor schedule consists of a set of time-continuous actions for each satellite. In order to improve the tractability of this problem, we have chosen to focus primarily on slewing and dwelling as the two actions that satellites can execute to observe their environment. During the scheduling process, satellites are faced with a decision making process where,

1. A set of actions available to each satellite are explicitly enumerated
2. This set of actions is constrained according to feasibility
3. The anticipated effect on the environment is evaluated based on previous knowledge and a cost function
4. Feasible actions are ranked based on their anticipated effect on the environment
5. The actions resulting in the most desirable effect on the environment are assigned to a schedule

In our problem, we have multiple satellites cooperating to search a cislunar volume. In order to promote cooperative behavior between satellites, we implemented an information sharing heuristic where each satellite is given a set of shared information that may or may not have an impact on the immediate priorities of a satellite. If this information does prove to be useful, the scheduling algorithm can utilize it to make decisions that would improve the group's overall performance.

4.3 Mathematical Formulation

4.3.1 Scheduling Problem

Scheduling problems occur in many applications [12], they exist in scenarios where there are limited resources to accomplish a high dimensional goal. Recall, in this cislunar SDA sensor tasking problem, the goal is to search 3.4 XGEO volume and cislunar corridor over six hours with as few resources as possible. This results in a high dimensional solution space with a limited number of resources to observe this cislunar volume.

First, we will mathematically define our problem following the notation in [13]. We will define a schedule, X, as the following:

$$X = \{\vec{x}_{st}(d_{st}) : \forall s \in [0, S-1],\ \forall t \in [T-1]\}, \tag{1}$$

where S is is the total number of observers, T is the total number of time windows in our time interval, and $\vec{x}_{st}(d_{st})$ is the slew direction with dwell duration d_{st} of observer s at time t. Then the set of feasible solutions, namely the set of all possible schedules, is defined by the following:

$$F = \{X_1, X_2, ..., X_M\}, \tag{2}$$

where M represents the number of total possible schedule combinations.

4.3.2 Cost Function and Constraints

To quantify the quality of a schedule, we will use a cost function. The components we will consider in this function are the following: target priority of visitation, target probability of detection, cost to slew, and a direction weighting factor. We will describe these components similar to [13][14].

In this scenario, the **priority** of visiting target n is μ_n. Once a target is observed, its priority is reset for the current evaluation time span. This is one of the main ways in which coordination among observers is handled. When one observer has seen a target, that target priority is reset so that another observer will not attempt to observe that target.

The **probability** of a target being detected by each observer, p_{ns}, may be modeled by the cumulative distribution function (cdf) of a normal distribution. We will also consider the **cost of slewing** to various positions, p_s. Our implementation allows for use of a **slew bias** for each looking direction for the observers. We will represent this as

a weighting factor per observer, $\vec{w}_s$. This will include a specific weight for each looking direction the observer can accomplish.

With the components outlined above, we have the following objective function to minimize:

$$C = \sum_{s=0}^{S-1} \sum_{t=0}^{T-1} \left[\sum_{n=0}^{N_{ts}-1} -p_{ns}\mu_n \right] - p_s w_s. \tag{3}$$

Thus, we want to solve the following problem, where X^* is the optimal schedule and X_k represents some kth schedule:

$$X^* = min_{X_k \in F}\ C. \tag{4}$$

The feasibility of actions are determined via a constraint function which takes into account Solar, Lunar, and Earth exclusion zones, umbral/penumbral lighting conditions, maximum allowable time allocation for slewing between pointing configurations, and minimum/maximum allowable dwell times for predefined sensor collection modes.

4.4 Algorithms

As mentioned previously, this is a high dimensional problem. Brute force or predefined scanning plans are not practical or efficient methods for developing collaborative sensor scan schedules for cislunar volume search. Thus, we have implemented two optimization algorithms, greedy and genetic, to solve this sensor scheduling problem. We will compare these algorithms and show that the genetic algorithm outperforms the greedy algorithm.

4.4.1 Greedy

A standard greedy algorithm chooses the best locally available option at the current step [1]. At each increment of time and for every observer, a pointing configuration and dwell time is chosen such that the objective function is satisfied. The benefit to leverage rule-based greedy algorithms is that they tend to be easy to build and computationally efficient. One disadvantage to the rule-based greedy algorith is that it only guarantees a local optimal architecture performance over planning durations. Since the greedy algorithm has no concept of future possibilities, it may not maximize the number of pointing positions over the entire planning duration. Hence, in the next section, we employ a genetic algorithm to solve this problem.

4.4.2 Genetic

Genetic algorithms (GA) are heuristic-based evolutionary optimization methods that are inspired by biological processes such as natural selection and crossover/mutation. GAs have been used successfully in past SDA satellite tasking problems [13, 6, 8].

The typical genetic algorithm follows the processes of (1) initializing a population, (2) crossing over population pairs that are “fit enough", and (3) mutating new population members. Broadly defined, a population member is a feasible solution to the problem. Thus, step (1) consists of initializing a population of feasible solutions of a given size.

These members in a population undergo the same process as in standard genetic algorithms. Namely, an initial population is generated that consists of candidate solutions. Population member pairs are chosen that are “fit enough". Fitness in this case is determined by number of unique collections generated from an individual population member which consists of a feasible schedule.

After choosing “fit enough" population member pairs, these are crossed over to form a new population member. This crossover process involves combining the two member pairs to create a new “child" population member. The last step is to mutate this new population member. This is done by randomly changing elements of this member. This whole process is repeated for a set number of generations. You can see this process in Figure 7. For more detail, see [13].

4.5 Capacity-Based Architecture Optimization

In this section, an approach to tackle the capacity-based architecture optimization will be presented. First, the method for determining the number of sensors in our architecture will be described. This method will rely a greedy algorithm

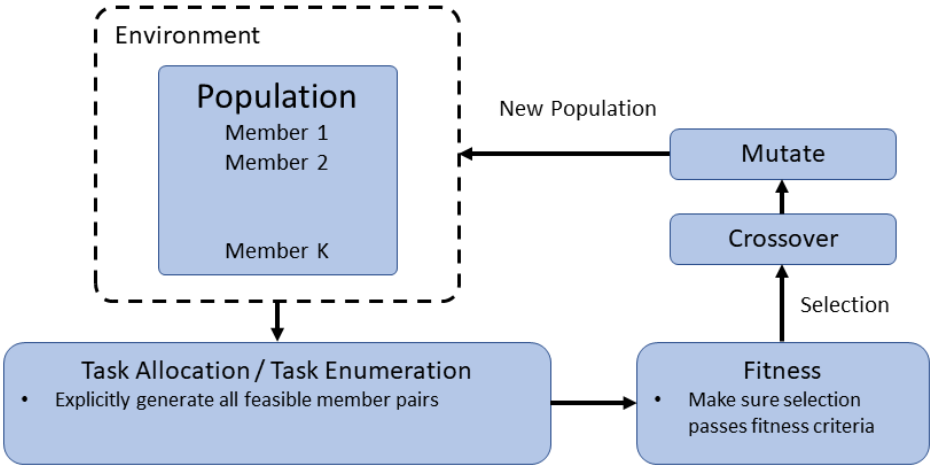


Fig. 7: A single generation of genetic algorithm flow chart. This process continues for a set number of generations.

to understand the potential capacity percent of each architecture. After this, we will use a genetic algorithm to further maximize capacity by optimizing sensor tasking. Results will be presented for sensors with mv 20 sensitivity and 4x4 deg FOVs.

4.5.1 Candidate Architecture based on Capacity Analysis

Due to the simplifications of observability analysis, further work from Section 3 was required to find architectures that provided adequate coverage subject to capacity analysis. It was not clear exactly how many sensors would be needed to accomplish our coverage goal once scheduling was considered. Therefore, architectures were formed with varying number of satellites using the combinations of orbit families shown in Figure 6 in Section 3. We then analyzed the capacity performance of each architecture to find the minimal number of sensors to accomplish our coverage goal for our given cislunar volume.

We used a greedy algorithm to evaluate the capacity performance for each of these architectures. A greedy algorithm was chosen due to fast run times, allowing for capacity results to be assessed over a lunar cycle. The top performing architecture consisted of seven satellites, with six satellites in a 3:1 resonant orbit and one satellite in a northern L1 halo orbit. Results from this architecture are shown in Figure 8. This plot shows the average observability and capacity calculated every six hours. The capacity over a specific six-hour coverage period will increase once further optimized with the steps shown below. Thus, we are not targeting a specific coverage metric at this point.

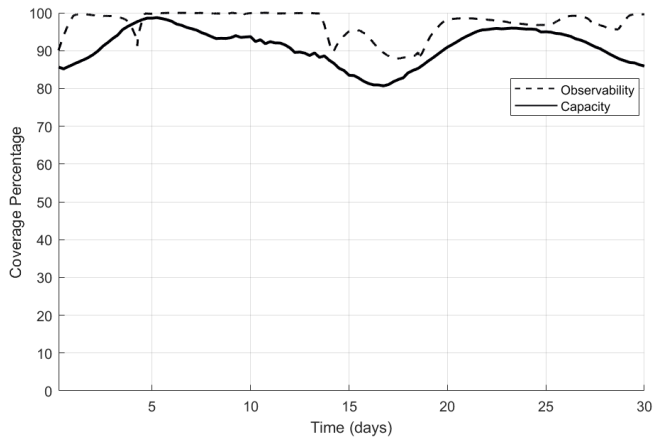


Fig. 8: Average coverage calculated as average every six hours.

4.5.2 Orbit Optimization

The flexibility of our GA implementation allows for application to problems other than the sensor scheduling problem. In this work, we also used the GA to find favorable orbits for a chosen architecture. Specifically, we varied the orbital period of the orbits and the phasing between satellites to maximize the capacity coverage of the architecture.

This approach was applied to the seven sensor architecture described in Section 4.5.1. Within the algorithm, the period of the orbits and the phasing were parameters with which to optimize capacity.

Figure 9 shows the cislunar architecture following this process. It consists of seven satellites: three in long-period 3:1 resonant orbits (orange), three in short-period 3:1 resonant orbits (green), and one in an Earth-Moon northern L1 halo orbit (blue). The orbital elements of the architecture orbits is listed in Table 2 in the CRTBP frame.

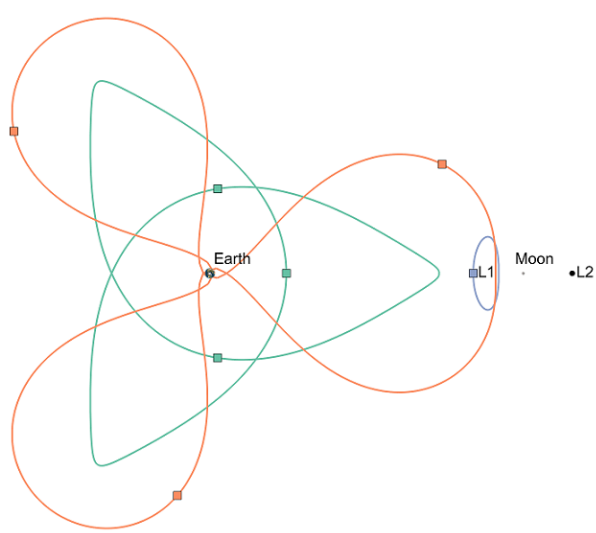


Fig. 9: Candidate seven satellite cislunar architecture.

To support further analysis, a wide variety of additional parameters can be varied within the GA implementation. These include orbital parameters, initial epoch, Walker constellation parameters (Earth-orbiting constellations). This allows for additional optimization objectives such as architecture orbit stability, minimal eclipse time, revisit time, etc.

Orbit Family	Period (days)	Position (LU)			Velocity (LU/TU)		
		X	Y	Z	VX	VY	VZ
Resonant 3:1 (long period)	28.96	2.32514797E-01	0	0	7.40948200E-13	2.20686855	0
Resonant 3:1 (short period)	27.77	2.16514784E-03	0	0	5.84399469E-11	1.16421184E+01	0
Northern L1 Halo	12.35	8.31192933E-01	0	1.21301233E-01	2.17827000E-15	2.36017313E-01	8.39036000E-15

Table 2: Orbital parameters for the candidate cislunar architecture. Elements are in the CRTBP frame where LU (389703 km) is the length unit and TU (382981 sec) is the time unit used for normalization.

4.5.3 Multi-Satellite Capacity-based Optimization

Finally, we present results for the chosen architecture with sensor scheduling. Figure 10 shows a comparison of the coverage calculated through the use of the two scheduling optimization algorithms. The maximum cumulative coverage of the greedy algorithm was 90.5% capacity while the genetic algorithm was able to achieve 94% capacity. Although the rule-based greedy algorithm is computationally more efficient, this implementation of a genetic algorithm improved upon greedy results and provided a reliable and systematic way to create cooperative schedules for multi-satellite cislunar architectures for SDA missions.

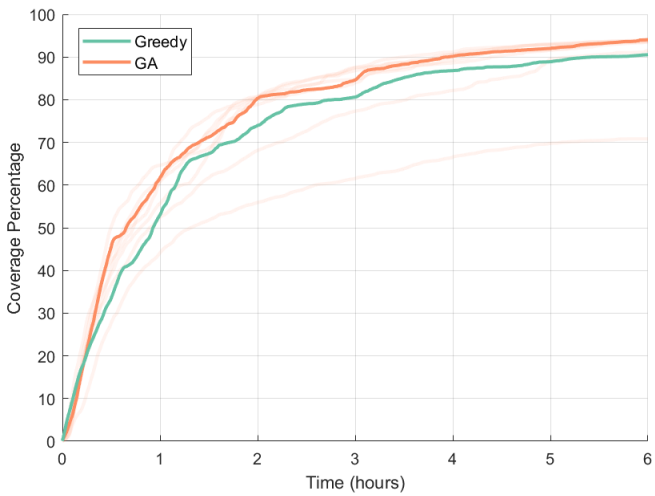


Fig. 10: The cumulative coverage achieved via the greedy and genetic algorithms.

Results shown here represent optimization of a schedule over a single six-hour time interval. In our work, we found that the genetic algorithm consistently outperformed the greedy algorithm for different architectures and cost function weighting. The genetic algorithm will test solutions that the greedy algorithm would not find. For example, the greedy algorithm would never search areas of space that have a low priority when a better option is available for the current time. A sensor that is exhibiting greedy behavior does not have the ability to foresee where it will be looking in the future, nor does it have the ability to plan based on the actions of other sensors. The genetic algorithm can attempt actions with lower immediate reward that may lead to better coverage over the entire time span.

In Figure 11, we can see the individual orbit family performance, as well as the total capacity achieved. These results were generated from our genetic algorithm. The long-period 3:1 resonant orbits have the largest contribution to the total coverage. The contribution of the short period 3:1 resonant orbit and the northern L1 halo orbit are primarily for regional coverage near GEO and the Moon, respectively.

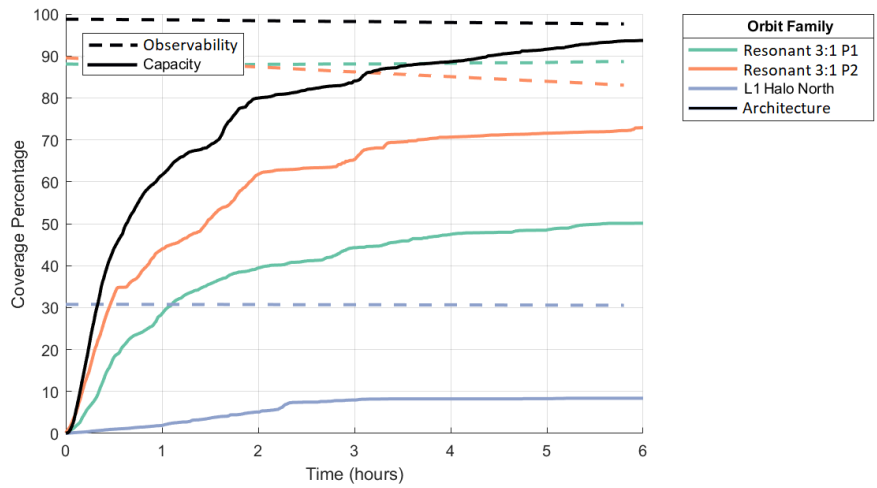


Fig. 11: Observability and capacity of candidate architecture after optimized with a genetic algorithm. Colored lines show the relative contributions of each orbit family to the mission objective.

Figure 12 is a heat map of the number of collects achieved in the 3.4 XGEO and cislunar corridor projected onto the Earth-Moon rotating plane. The lighter colors represent regions where the collections are less frequent. Darker colors represent regions where points were observed more often. Though not shown here, our implementation allows for the priority of some regions to be set higher so those regions are visited more frequently.

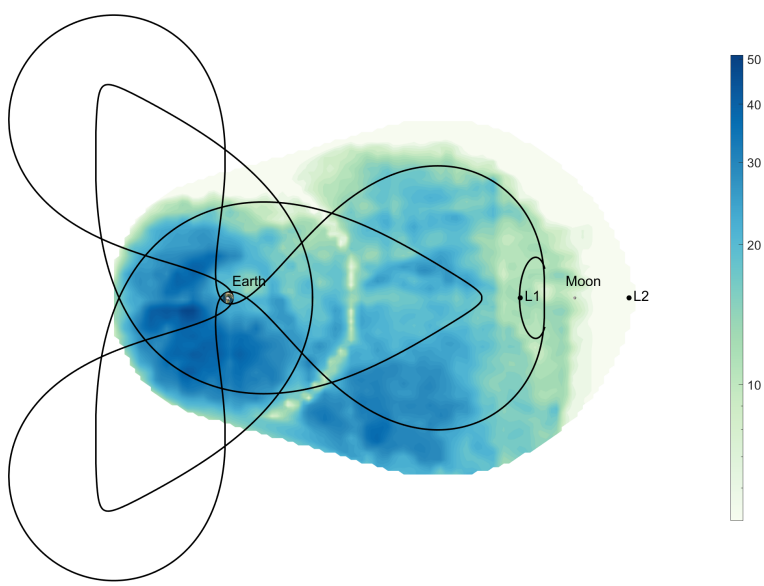


Fig. 12: A heat map of the number of collects achieved over six hours in the 3.4 XGEO + cislunar corridor.

In this section, we showed that the greedy algorithm was a useful step in our capacity-based cislunar architecture design. However, the genetic algorithm outperformed the greedy algorithm and allowed for even better results to be generated as our final capacity optimization step. The genetic algorithm generated the best coverage when used as the final step in optimizing sensor schedules. Additionally, we showed the flexibility of our genetic algorithm implementation by using it to optimize orbital parameters and phasing between satellites.

5. CONCLUSION

In conclusion, we have extended current techniques in sensor architecture design by not only considering sensor observability, but also capacity. We first considered the observability of many different architectures, through the evaluation of many different types of cislunar orbit families. After this, we used capacity analysis to size the architecture appropriately for realistic volume search scenario. Lastly, we optimized the schedule of sensor tasks to maximize coverage of the cislunar volume. We have shown that optimizing observability alone is not sufficient for creating realistic cislunar SDA architectures. Our results show a significant improvement when also considering capacity in creating these architectures. We have offered a solution for capacity-based cislunar SDA that can be extended to different types of cislunar volumes than this specific volume.

We shave shown results for a specific architecture, though this is not the only solution for providing cislunar SDA. The targeting of a specific metric like revisit rate or a change to the volume could lead to different architecture designs. Our sensor scheduling optimization implementation could provide those solutions. This work offers an approach to solving future capacity-based architecture design problems.

6. REFERENCES

[1] Paul E Black et al. Dictionary of algorithms and data structures. 1998.

[2] James Bridenstine. Memorandum of understanding between the national aeronaurtics and space administration and the united states space force. 2020.

[3] John Carrico, Lisa Policastri, and Stephen Stephen Lutz. Ibex revisited: Operational results for long-term cislunar orbit propagation, 2018.

[4] Sandra Erwin. Air force: Ssa is no more; it's' space domain awareness'. *SpaceNews, November*, 14, 2019.

[5] Samuel Fedeler, Marcus Holzinger, and William Whitacre. Sensor tasking in the cislunar regime using monte carlo tree search. *Advances in Space Research*, 70(3):792–811, 2022.

[6] Michael S Felten. Optimization of geosynchronous space situational awareness architectures using parallel computation. 2018.

[7] Roberto Furfaro, Vishnu Reddy, Tanner Campbell, and Bill Gray. Tracking objects in cislunar space: The chang'e 5 case. 2021.

[8] Andreas Hinze, Hauke Fiedler, and Thomas Schildknecht. Optimal scheduling for geosynchronous space object follow-up observations using a genetic algorithm. In *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*. Maui Economic Development Board Maui, HW, 2016.

[9] MJ Holzinger, CC Chow, and P Garretson. A primer on cislunar space. 2021.

[10] JPL. Three-body periodic orbits *https//ssd.jpl.nasa.gov/tools/periodic_orbits.html*.

[11] Simon R Knister. Evaluation framework for cislunar space domain awareness (sda) systems. 2020.

[12] S Mitrovic-Minic. Collection planning management: Multi-satellite collection scheduling. task 1 report: Problem statement and literature review. *Defence Res. Develop., New Delhi, India, Tech. Rep. DRDCRDDC-2016-C228*, 2016.

[13] Naomi Owens Fahrner. A regional greedy algorithm for space domain awareness resource allocation, 2021.

[14] Naomi Owens Fahrner. *Resource Allocation for Space Domain Awareness and Synthetic Aperture Radar*. PhD thesis, Colorado State University, 2022.

[15] Ming Xu, Jinlong Wang, Shengli Liu, and Shijie Xu. A new constellation configuration scheme for communicating architecture in cislunar space. *Mathematical Problems in Engineering*, 2013, 2013.