Spring 2017

Satellite Sequencing Optimization and Observational Orbit Determination Using Genetic Algorithms

Andrew W. Verstraete
Western Michigan University, andrewverstraete@gmail.com

Recommended Citation

Verstraete, Andrew W., "Satellite Sequencing Optimization and Observational Orbit Determination Using Genetic Algorithms" (2017). *Master's Theses*. 906.
http://scholarworks.wmich.edu/masters_theses/906

SATELLITE SEQUENCING OPTIMIZATION AND OBSERVATIONAL ORBIT DETERMINATION USING GENETIC ALGORITHMS

by

Andrew W. Verstraete

A thesis submitted to the Graduate College
in partial fulfillment of the requirements
for the degree of Master of Science in Engineering
Mechanical and Aerospace Engineering
Western Michigan University
April 2017

Thesis Committee:

Jennifer Hudson, Ph.D., Chair
Tianshu Liu, Ph.D.
Brook Sullivan, Ph.D.
Bob White, Ph.D.

SATELLITE SEQUENCING OPTIMIZATION AND OBSERVATIONAL ORBIT DETERMINATION USING GENETIC ALGORITHMS

Andrew W. Verstraete, M.S.E.

Western Michigan University, 2017

The problem of mission design for a robotic servicing satellite in geosynchronous Earth orbit (GEO) was investigated. A representative set of potential client satellites was selected, and operational needs were randomly assigned based on the average number of GEO retirements, anomalies, and repositioning maneuvers that currently occur each year. An objective function was developed to represent the value of servicing mission sequences, including client fees, time penalties, and operational risk. A genetic algorithm was then used to find sequences of operations on the potential client set that maximized the objective function's value. Scenarios were analyzed with the database of satellites as well as with a dynamic client model. Sequences that begin with repair operations and later include refuel, observation, and retirement maneuvers were found to be the most valuable, with some differences in the optimal sequences depending on parameter values in the objective function. A second genetic algorithm was then developed to determine optimum firing patters for a satellite performing an observational relative orbit around another satellite. The associated objective function minimized the cost of metrics such as relative distance, illumination, observational coverage, and fuel consumption.

ACKNOWLEDGEMENTS

I would first like to acknowledge my mentor, Dr. Jennifer Hudson. Her availability and willingness to answer my questions truly enabled me to finish this thesis. I would also like to thank her and the other members of my committee, Dr. Tianshu Liu, Dr. Brook Sullivan, and Dr. Bob White for taking the time to review my work.

Secondly, I would like to thank my friend and colleague, Nicole St.Louis, whose assistance and encouragement was invaluable.

Finally, I would like to thank Dr. Scott Erwin and the AFRL for their guidance and support during my time there last summer.

Andrew W. Verstraete

TABLE OF CONTENTS

Table of Contents--Continued

LIST OF TABLES

LIST OF FIGURES

NOMECLATURE

β = Time penalty, lost revenue per day of customer non-operation

C = Total value of a chromosome

$c_{1,2}$ = Scaling factor for relative weight of terms

ΔV = Change in orbital velocity

$\delta[]$ = Change in relative [] position of the deputy to the chief

λ = Longitude

μ = Gravitational parameter

n = Mean motion of the chief

p = Probability of operation success

R = Risk factor associated with a satellite

r = Number of satellites in the satellite database

$\boldsymbol{r}$ = Orbital position vector

ρ = Probability that every satellite will show up at least once in the first generation

s = Number of satellites in each chromosome

T = Time from beginning of mission to completion of satellite operation

T_{allow} = Wait time allowed before daily time penalty deductions begin

θ = Angle between the sun and the deputy satellite

V = Revenue from a servicer operation

x = Number of chromosomes in each generation

I. INTRODUCTION TO THE SATELLITE SERVICING PROBLEM

Satellites in geostationary Earth orbit (GEO) are highly valuable assets with finite design lives. In many cases, these lives could be extended by on-orbit robotic servicing. However, the GEO belt is very expensive to reach and adding that cost to that of building a robotic servicer would make single-satellite servicing missions impractical limited profit potential. As such, GEO satellites are simply replaced when they reach the end of their operational lives. A robotic vehicle that could be launched only once and yet service multiple client satellites would be a more cost-effective alternative. Out of approximately 400 satellites in GEO, there are an estimated 30-40 satellites every year in need of some sort of service.[1] The selection and sequence of clients to visit in a single trip could vastly reduce the cost of a servicing mission.

Five servicing operations were considered: observation, repair, reposition, refuel, and retire to disposal orbit. Each operation would require the servicer to rendezvous with the customer satellite. Repairing and refueling would require docking and robotic operations, while repositioning and retiring would require the servicer to perform a maneuver while attached to the customer spacecraft. Each operation is associated with a different risk level and potential value to the customer. This thesis addresses the problem of finding a maximum-value trajectory to service a sequence of GEO satellites.

This analysis focuses on (1) methods for evaluating the economic value of a multi-client GEO servicing mission, (2) development of a genetic algorithm for finding optimal mission sequences given a large client satellite database, (3) identifying trends in high-value servicing missions to inform the design and operation of future robotic servicing vehicles.

Satellite sequencing for the mission design of a GEO servicer represents the highest level of this problem. The genetic algorithm developed for the servicer determines where the servicer should be and places it in the general vicinity of the client. There are not any nested programs to tell the servicer what to do during the operations. The way the servicer performs these proximity operations would vary drastically for each operation listed. In Section VI, the proximity operation of a satellite observation is explored in further detail.

A. Background

The Defense Advanced Research Projects Agency (DARPA) recently initiated the Robotic Servicing of Geosynchronous Satellites (RSGS) program. Within the next five years, DARPA intends for the RSGS program to have a servicer able to inspect and service both commercial and military GEO satellites on a fee-for-service basis.[2] Two other recent programs in on-orbit robotic servicing technology are the DARPA Phoenix program and NASA Goddard's Satellite Servicing Capability Office (SSCO). Phoenix originally sought to develop methods to collect hardware, such as antennas, from retired satellites to be reused, while NASA's SSCO has been working on technology for the refueling of GEO satellites.[3] In 2007, the DARPA Orbital Express program demonstrated the feasibility of robotic, autonomous, on-orbit refueling, and reconfiguration of satellites in low Earth orbit.[4] During the program's demonstrations, liquid propellant, a battery, and a flight computer were transferred from the Autonomous Space Transport Robotic Orbiter (ASTRO) to the NextSat satellite. NASA's Robotic Refueling Mission (RRF) also recently began demonstrating on-orbit robotic systems by performing a demonstration on the International Space Station.[5]

Aside from DARPA and NASA, commercial entities have been pursuing robotic servicer technology as well. Canada's MacDonald Dettwiler and Associates (MDA) has been working towards the development of a craft capable of refueling on-orbit satellites while other companies, ViviSat and Alliant Techsystems Inc. (ATK), have started developing on-orbit tug technology.[3]

Building a GEO robotic servicer would be costly, but the expense could be justified by an inspection of the financial state of the satellite industry. In 2014, the satellite industry revenue was $203 billion with an overall industry growth of 4% worldwide.[6] A satellite launched into GEO can cost upwards of $200 million.[3] If a satellite were to experience an anomaly early in its life that severely diminished its performance capabilities, the assistance of a robotic servicer that is already in orbit and able to rendezvous with the satellite would be of high value. For example, if a GEO satellite were to generate approximately $50 million per year[7], and it experienced a failure early in its life that decommissioned it, a revenue of over $750 million over its 15-year estimated lifespan would be gone. Considering that it is estimated that about half of on-orbit failures could be corrected by servicing,[3] this is an area of high interest from an economic standpoint.

Aside from early anomalies, servicing satellites that are near the end of their lives could be of high value as well. If a servicer were able to extend the life of a satellite by a tug or refuel, even if by a single year, the company owning the satellite could continue to profit from it. For example, if each transponder on a satellite can make, on average, $2 million a year in revenue, a small satellite containing 25 transponders would generate $50 million in revenue a year while a large satellite with 48 to over 100 transponders could generate more than $200 million.[3] One recent study roughly converted the revenue of a GEO satellite by the costs of fuel to determine the value of a refueling mission. It was found that fuel would generate about $468,000 per kilogram. The same study found that the cost of delivering fuel to the spacecraft would be about $40,000 per kilogram, yielding a high value-to-cost ratio.[7]

Satellite servicing, specifically autonomous servicing, is becoming more of an interest to satellite owners due to the frequency of anomalies and the cost of satellites. The technology for such a servicer has grown and continues to grow, which leads to a new area for research, the most efficient way to service GEO satellites. To be cost-effective, the servicer would need to rendezvous with multiple satellites per mission, therefore, the next issue encountered and the one addressed in this paper, is the issue of deciding the best sequence of satellites to service.

B. Problem description

The demand for a GEO robotic servicer leads to a unique problem. How can a servicer be efficiently implemented in a way that maximizes the monetary gain? It is extremely expensive to launch vehicles into space and this expense is exacerbated if a new servicer has to be launched to every single satellite that has any sort of servicing need. This means that a GEO servicer would be much more attractive if it were able to service multiple satellites in a single mission. The issue with multi-satellite servicing is now efficiency. How can the trajectory of this servicer be optimized to operate on the most profitable satellites in the most efficient order? There are multiple factors to be taken into account when developing a mission trajectory. Some of these factors include the location of the potential customer's satellite, the price the customer is willing to pay for the service, the urgency of the service, and the type of service performed. The type of service performed is particularly important because different services, such as satellite observation, repositioning, end-of-life transfer to disposal orbit, and deployment anomaly correction, have different risks associated with them and different needs in terms of fuel and time required for the service. The objective of this project is to find optimal paths to service satellites in the GEO belt with various operational needs while remaining within servicer fuel and time constraints.

A genetic algorithm (GA) was developed to search for the highest-value mission sequences among a set of GEO satellites with various longitudes and servicing needs. GAs have been widely used for multi-segment spacecraft trajectory design problems.[8,9,10] A database of potential customer satellites was used to test the GA. For initial testing, simulated orbital phasing maneuvers with impulsive burns were used to perform transfers between GEO customer locations. Preliminary results have been compiled for sequences of 10 customer satellites.

II. ECONOMIC VALUE OF GEO SATELLITE SERVICING

A. GEO satellite servicing opportunities

Commercial GEO satellites average about $50 million per year in revenue.[7] Several recent studies, notably References [1,7,11], have identified several opportunities for robotic servicing in GEO.

Each operating GEO satellite requires about 55.4 m/s ΔV per year for GEO station-keeping, primarily for inclination control. If an active satellite depletes its onboard propellant stores, it may continue operating with north- south drift (an estimated 24-30% of GEO communication satellites exceed their design lives[12]) or it may perform a retirement maneuver to reach a disposal orbit, typically about 300 km above GEO. Approximately 7.5 satellites begin inclined operations each year and another 10-15 are retired.[7] Satellites approaching the end of their design lives may be candidates for on-orbit refueling or transport to disposal orbit. In some cases, a satellite must burn large amounts of propellant to correct launch vehicle errors or propulsion system problems, in which case GEO refueling could restore the satellite's intended life expectancy.

An average of 10-15 commercial GEO satellites perform relocation maneuvers each year to change operating locations.[7] A versatile robotic servicer could perform "space tug" operations, allowing these commercial satellites to reserve their onboard propellant for station-keeping.

Additionally, on-orbit anomalies, such as failure to deploy a solar array, can be costly for GEO satellites. An average of 0.5 mechanism deployment anomalies and 3.4 other anomalies occur each year.[7] Satellites that experience deployment anomalies may be candidates for robotic operations, for example to release a stuck appendage, while other types of anomalies may warrant on-orbit inspection to determine the nature of a failure.

B. Satellite database and assumptions

This study made use of a database of all operational GEO satellites that has been compiled from public sources.[7] From the database, candidate satellites were identified that met three criteria: (1) owned and launched by U.S. operators, (2) currently active, and (3) operating with orbit inclination less than 1°. A total of 95 satellites met these criteria; these were classified as potential clients (if a GEO robotic servicing vehicle were available today).

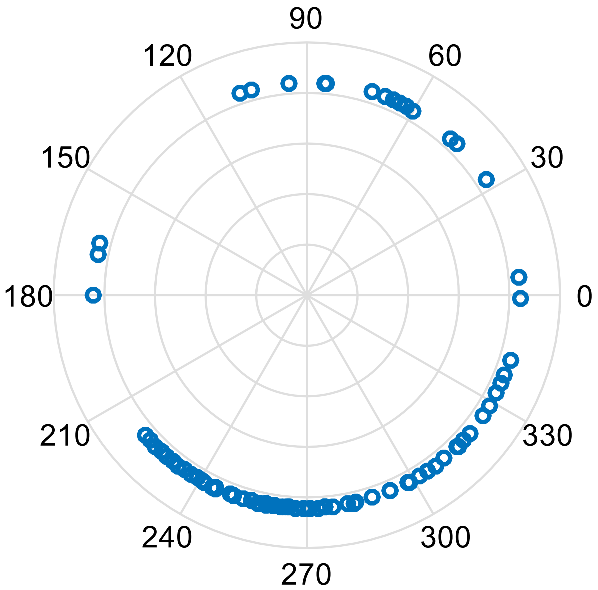

Figure 1. Locations in eastward longitude of 95 potential servicing clients.

Satellites in the potential client set were randomly assigned servicing needs, based on the approximate annual rates of serviceable incidents. The assigned needs included:

- 2 standoff observations
- 5 repairs
- 14 repositions
- 15 retirements
- 17 refuelings

yielding a total of 53 potential operations. The constraint was added that only 10 satellites would be selected for any single mission. For most of the cases, all of the satellites and the associated data were known before the servicer embarked on a mission. However, cases were also run with a dynamic client model, where a new satellite was added mid-mission.

III. METHODS

A. Genetic algorithm

A genetic algorithm (GA) is an optimization method that mimics natural selection and evolution. A GA is typically used to find an adequate solution when the number of possible solutions is so extremely large that it is not feasible to test each solution. The vernacular of a GA mimics that of evolutionary biology. Each possible solution is called a chromosome, while each iteration of the algorithm is called a generation, as seen in Figure 2.

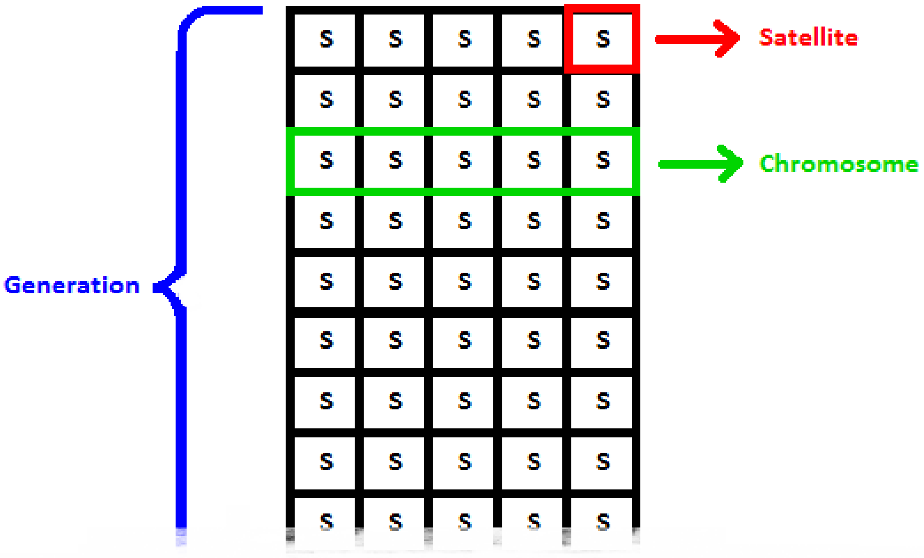

Figure 2. Example generation in the GA

For this GA, a chromosome represents the satellites selected in a particular sequence for a single servicing mission. The algorithm runs each chromosome through the objective function to determine its fitness, as in “survival of the fittest”, and uses this fitness to determine which chromosomes produce “offspring” for the next generation. These “offspring” are slightly altered copies of the chromosomes from the previous generation. The fitness function evaluates the value of a chromosome based on the potential monetary gains and losses from the perspective of the service operator. The chromosome determined to have the highest fitness in the generation is set as the leader of that generation and protected from genetic operations, or alterations, in the next generation. The remaining chromosomes are assigned a probability of producing offspring proportional to their fitness values. This means that it is more likely for the more fit chromosomes to produce offspring in the next generation. Due to this, each generation should produce an equal or better solution than the previous generation, until an optimal solution is found.

In this study, the desired solution is a specific sequence of satellites to visit on a multi-satellite servicing mission. Because the order does matter, the number of possible solutions is

$$N = \frac{r!}{(r-s)!} \tag{1}$$

where N is the number of possible outcomes, r is the number of satellites available to choose from, and s is the number of satellites to visit in a trip. Applying this equation to the full set of 53 GEO satellites, if 10 satellites are to be visited in a single mission, there would be over $7\text{x}10^{16}$ possible solutions. The magnitude of this number offers some insight as to why a genetic algorithm was used. With the available computing power, it takes approximately one minute for the GA to analyze 100,000 possible solutions. At this rate, if the solution were to be found instead by going through an analyzing every single solution, it could take more than *1,000,000 years*! Obviously, this is impractical and most of the potential solutions are no good. Using the fitness function with the GA weeds out bad solutions early on in order to hone in on the better solutions.

One strategy for choosing the number of chromosomes per generation is based on a user-assigned probability that each satellite in the database appears at least once in the first generation. In this strategy, the number of chromosomes in each generation is

$$x = \frac{\ln(1-\rho)-\ln(r)}{\ln(1-\frac{s}{r})} \tag{2}$$

where ρ is the chosen probability that every satellite in the database shows up at least once in the first generation. If $\rho = 0.99$ for a 10-satellite sequence with database size $r = 53$, each generation should have at least 41 chromosomes. This analysis was performed to affirm the decision to use 100 chromosomes in each generation in this study. This meant that the probability of each client appearing at least once was much greater than 99 percent.

The first generation is randomly populated with satellites from the database. For each satellite selected, the GA checks to make sure that the satellite has not already been selected in that chromosome. The chromosomes from the first generation act as seed for the rest of the algorithm. Once the first generation is set, the second generation is selected for reproduction using a roulette wheel GA methodology. All other generations are iteratively created and evaluated for fitness until the stopping criteria are achieved.

In order to vary the chromosomes and find a leader with a higher value, various genetic operations are performed. The following genetic operations are performed on each non-leader chromosome in a generation:

- Mutation: A random satellite within the chromosome is selected and replaced with another satellite from the database. Protections ensure that the replacement satellite is not the same as any of the satellites already in the sequence.
- Permutation: There are two types of permutations present in this GA. The first selects two satellites and swaps their locations within the chromosome. The second selects two satellites within a chromosome and reverses the order of all satellites between them.
- Crossover: One satellite from one chromosome and another satellite from the next chromosome are randomly selected and switched. There are protections in place to ensure that a crossover operation does not lead to redundancies in the selected satellite sequences.

To avoid local maxima within the GA, immigration is used to increase genetic diversity. If there has been no change in the leader for 5,000 generations, immigration introduces new chromosomes. All chromosomes whose fitness is less than the mean fitness of the generation have a 50% chance of being replaced with a new, randomly generated chromosome. Immigration tends decrease the average fitness of the generations; however, it often identifies a new leader.

Two stopping criteria were applied to the GA. The first limits the total number of GA iterations to a maximum number of generations. The second terminates the algorithm after 10,000 consistent leader results. Many cases were run with these two stopping criteria and results were compared after 1,000 generations, 10,000 generations, and the final generation. It was discovered that the value of the leader averaged 99.5% of the value of the final solution after only 1,000 generations and 99.97% after 10,000 generations. The sequencing results in Section IV, all of the cases presented were terminated after 10,000 consistent generations.

B. Trajectory maneuvers

A robotic servicing vehicle with a high-thrust chemical propulsion system was assumed. Trajectory maneuvers from one GEO client to another were modeled as impulsive phasing maneuvers. The phasing orbits were sized for 2° per day eastward or westward drift; each transfer is assumed to take $\Delta\lambda/2$ days, rounded to the nearest even integer. Upon reaching the next client's location, the servicer performs a second burn to re-circularize its orbit. The total ΔV required for each client-to-client transfer is 10.8 m/s.

A similar approach is used for repositioning and retirement operations. In this analysis, we assume each repositioning operation changes the client's position by 12° longitude, so an extra six days and 10.8 m/s ΔV are added to the mission. Retirements are modeled as Hohmann transfers to a circular graveyard orbit 300 km above GEO. We assume an extra 10.88 m/s ΔV for the servicer to transfer the client to the disposal orbit and return to GEO.

C. Objective function

An objective function was defined to evaluate, and maximize, the value, from the service operator's perspective, of a sequence of satellite servicing operations. The objective function includes revenue, risk, and time penalties:

$$Maximize\ C = \sum_{i=1}^{s} \left[c_1 \frac{V_i}{2-\prod_{k=1}^{i} p_k} - c_2 \beta_i max\left(0, T_i - T_{i,allow}\right) \right]. \tag{3}$$

The first term in Equation 3 is based on the method of discounted cash flow in financial investing. The discounted value of an investment with risk is

$$Value = \frac{cashflow}{1+risk\ discount\ rate}, \tag{4}$$

assuming the time value of money is constant. For a satellite operation with a risk of failure, the risk discount rate is $(1-p)$. In a sequence of robotic servicing operations, it was assumed that the risk compounds; the probability of successfully completing a given operation is the product of the probabilities of success of the operation itself and all preceding operations in the sequence. For a sequence of operations $A \rightarrow B \rightarrow C$, the risk-adjusted value is

$$Value_{ABC} = \frac{V_A}{1+(1-p_A)} + \frac{V_B}{1+(1-p_A p_B)} + \frac{V_C}{1+(1-p_A p_B p_C)}. \tag{5}$$

The sum of the risk-adjusted values of the operations in a mission sequence is the first item in Equation 3. The second term evaluates any time penalties. If operation i is completed before time frame $\mathrm{T_{i,allow}}$ ends (with time measured from the beginning of the mission sequence, including all previous transfer times and operation times), no penalty is applied. If the time to complete operation i exceeds $\mathrm{T_{i,allow}}$, the penalty β is deducted from the servicer's revenue for each extra day, $T_i - T_{i,allow}$.

IV. RESULTS

A. Strictly algorithmic results

This first results section will deal only with the results in the sense of displaying how well the algorithm works. The focus of these results is consistency and rate of solution convergence. This does mean the satellite sequences will be closely analyzed, but they will be analyzed in a comparison to each other in order to determine the functionality of the algorithm. They will not be analyzed individually for economic merit. This does not however lessen the optimality of the output sequences and they will be shown and analyzed for their own sake later on in Section IV.B.

In order to get results, the algorithm was run multiple times for 100,000 generations and results were saved after 1,000, 10,000, and 100,000 generations.

Table 1. GA results after 100,000 generations

Sequence No.	**Longitude (°)**	**Operation**
1	244.8	Repair
2	261	Repair
3	265	Repair
4	319.5	Repair
5	301.88	Observe
6	241	Observe
7	241.15	Retire
8	304.5	Retire
9	259.01	Retire
10	325.45	Retire

Table 1 is representative of the many trial results for the genetic algorithm after running 100,000 generations. For all cases run, the first six satellites were exactly the same. The last four satellites were always retirement operations, however they were not always the exact satellites shown in the table. The significance of these runs is that they all converged on exactly the same final value; the value was the same down to the cent.

Now, 100,000 may seem like a high number of generations, but recall that there are more than 7×10^{16} possible solutions to the problem, and 100,000 generations only covers an absolute maximum of 10^6. Being that this is on the order of one billionth of a percent of the possible solutions, having all of the test runs converge on exactly the same results from completely random origins is encouraging. It shows that the algorithm works, and works consistently. However, this consistency comes at the cost of time which is a similar GA test was also run for 1,000 generations.

Table 2. GA results after 1,000 generations

Sequence No.	**Longitude (°)**	**Operation**
1	261	Repair
2	265	Repair
3	244.8	Repair
4	301.88	Observe
5	241	Observe
6	241.15	Retire
7	266.91	Retire
8	273.04	Retire
9	341.99	Retire
10	283.03	Retire

After only 1,000 generations, the results became harder to represent because the satellite to satellite consistency was not as high. In all cases, the first three satellites chosen were all repairs, and they were all the same repairs, however they were not always in the same order. Also in all cases, the last three satellites were all retirements, but not always the same retirements. The middle four satellites were more of a jumble. Most cases had the fourth satellite as a repair and the next three were a mix of the two observations and a retirement. At first glance, this inconsistency seems discouraging until it becomes known that all of these solutions still converged to within 1% of the value seen after 100,000 generations. The average value, in fact, was 99.5% of the value from the 100,000th generation. This proves that not only does the genetic algorithm work, but it works quickly. This means that the genetic operations on chromosomes coupled with the survival-of-the-fittest routine are effective in altering chromosomes for the better.

For all intents and purposes in a real world application, running 1,000 generations would be effective. For the purpose of academic rigor, many of the remaining results to be presented were obtained over runs of 10,000 generations of the GA. After 10,000 generations, the satellite to satellite consistency was much higher than 1,000, with only 33% of the results diverting from the

100,000 results; note that these diversions were much smaller than at 1,000 generations, with only two satellites switching places in the sequence. The converged value at 10,000 was usually exactly the same as 100,000 and even those that were not the same converged at the same answer of 99.97% of the 100,000 value.

B. Evaluating effectiveness of cost function terms

Now with the algorithms operation established, the actual results for satellite sequencing can be presented. As stated in Section III.C the cost function, Equation 3, has essentially three metrics that it uses to maximize the value of the mission: revenue, risk and time. To illustrate the impact of these three terms, the GA was run with the cost function in ways to show how they work independently and how they interact with each other.

Table 3. Risk-only GA results, $\boldsymbol{V} = \$\mathbf{10}\boldsymbol{M}$ *and* $\boldsymbol{\beta} = \mathbf{0}$ *for all satellites*

Sequence No.	**Probability of Success (%)**
1	99
2	99
3	98
4	98
5	98
6	98
7	98
8	98
9	97
10	97

Table 3 represents a case where essentially only risk is considered. Turning the time penalty, β, to zero and equating all the revenues at 10 million dollars from every operation yielded a solution that only had one varying factor according to the cost function; that factor was the risk of each operation, or, as it is defined, the probability of success. The probability of success is defined as the probability that the servicer will not experience a catastrophic event that completely decommissions the servicer and eliminates the potential for any further servicing operations. For the purpose of illustrating the effects of this single term, all satellites in the database were assigned a random probability of success based on a bell curve with a mean of 0.95 and a standard deviation of 0.02; the results make sense. The highest probability of success in the database was 99% which appeared twice, both appearing in the front of the solution. In fact, the results from the GA exactly match the top ten sorted values from the whole database. This shows that the algorithm chooses to put the operations with higher probability of success first in the sequence with that probability declining as the servicer progresses through the sequence. Intuitively, this makes sense as it pushes the riskiest operations to the end, minimizing the overall risk to the servicer. Section III.C shows how the risk term compounds throughout the sequence. This means that having the risks arranged as they are in Table 3 minimizes the impact of this compounding.

Table 4. Revenue-only GA results, $\boldsymbol{p = .9}$ *and* $\boldsymbol{\beta = 0}$ *for all satellites*

Sequence No.	**Revenue (M)**
1	30.7
2	29.1
3	28.3
4	25.5
5	24.8
6	24.4
7	24.3
8	24.2
9	24.2
10	24.1

The results for the cases where only revenue was considered were easy to obtain and easy to display. The revenues were determined from a randomly defined normal distribution curve with a mean of 20 million and a standard deviation of 3 million. The highest 10 revenues in the database were all selected in the final sequence. One thing to note is the order of the revenue values in the sequence. The probability of success for all satellites in the database was set to 0.9 for these runs of the algorithm. If they were all set to 1, and there was no risk of failure, all of the revenues in the solution would have been the same values, but in jumbled orders. Introducing a risk into the system ensured that the higher revenue satellites were chosen to be serviced first just in case the servicer was rendered incapable of continuing at any point in the sequence. By this logic, any value $0<p<1$ would have worked as the introduced risk. As described in Section III.C in Equation 5, the probability of success in the operations decreases continually throughout which is why it is beneficial to see the highest revenue satellites all coming first in this sequence.

Table 5. Time-only GA results, $\boldsymbol{p} = \mathbf{1}$ *and* $\boldsymbol{V} = \$\mathbf{10}\boldsymbol{M}$ *for all satellites*

Sequence No.	**Longitude (°)**	**Time Penalty ($/day)**	**Operation Time (days)**
1	241	112,451	10.0
2	241.15	127,269	20.0
3	256.98	127,172	25.1
4	259.01	97,412	21.1
5	266.91	108,529	22.8
6	273.04	126,302	22.2
7	283.03	109,311	23.4
8	301.88	117,586	16.0
9	325.45	90,557	32.0
10	335.48	106,923	33.4

Table 5 illustrates the case where only time is considered. The revenue value and probability of success for all satellites was set to the same values and the time penalty, β, was randomized using a normal distribution around $120,000 with a standard deviation of $10,000.

The individual effect of time without revenue or risk was more difficult to analyze and display. There were trends that appeared between all of the runs for this case, but they are not as easily seen as the trends for the revenue-only and the risk-only cases. One challenge was that there was not as much consistency between solutions as there had been in previous trial cases. This is because the time based trials take many more things into effect. This includes the varying times required to complete each operation, the time required to transfer between satellites in the sequence, and the interaction of these times with the determined time penalty of the satellite being serviced. The operation time column encompasses both of the times that were just mentioned. The sequence presented in Table 5 most wholly represents the general trends that were seen. The first trend is that the operation time for the first satellite is the smallest of all the operation times. The second is that the second satellite is the one with the highest time penalty. The final trend worth note is that the final satellite is the one with the highest operation time. These three trends show that the time evaluating term in the cost function (III.C, Equation 3) is functioning as intended. The first satellite being a low time operation has the minimal effect on the rest of the satellites. The second satellite being the highest value means that the penalty in revenue would be much smaller than if it were later in the sequence. Again, the final satellite being the highest time operation means that this high time has no impact on the sequence because there are no more satellites to follow it. Finally, the longitudes of all satellites chosen are in order, minimizing the transfer times between them and thus minimizing the overall time of the mission.

Table 6. Interaction of Revenue and Risk

Sequence No.	**Revenue (M)**	**Probability of Success (%)**
1	21	1
2	25	0.96
3	22	0.96
4	22	0.96
5	26	0.95
6	23	0.95
7	23	0.95
8	23	0.95
9	25	0.93
10	26	0.92

Now all of the factors in the cost function have been evaluated for individual effectiveness. There is, however one more trial that was run before the full cost function was used. As seen in the cost function itself, the revenue and the probability of success are both in one single term in the equation. As such, in Table 6, both the revenue and the risk are varied while maintaining the constraint of $\beta=0$. The risk term was given a mean of 0.95 and the revenue a mean of 20 million. The GA still chose the higher revenue satellites, as it did when only revenue was being considered, because all of the satellites in the sequence have revenues higher than the mean of the random assignment. The order, however, appears to have been solely determined by the risk. The sequence always followed the pattern of the probability of success decreasing as the sequence number increased. The value of the satellites chosen, though it is favorable that were always higher values satellites, seemingly had very little or nothing to do with the sequence in any of the trials. This shows that the risk outweighs the revenue in Table 6. Now note that the 9th and 10th satellites are lower success but higher revenue. As a matter of fact, 26M was the highest revenue in the database after random assignment. This means that the higher revenue satellites were selected even though they had a worse probability of success. This illustrates that the revenue was still a factor in determining which satellites were selected, even if they did not determine the order of those selected. A way to change this would be to increase the distribution of revenue values and decrease the distribution of risk values.

C. Client data assumptions

Previous studies of the business case for on-orbit servicing have included estimates of client fees for different operation types. Reference [1] estimated a fee range of \$3-10 million for standoff inspection and 5-15 percent of remaining value at risk for deployment assistance (corresponding to fees of \$12.5-37.5 million for a five-year life- extension, given \$50 million per year client revenue).

For a preliminary analysis, it was assumed that repair operations would warrant the highest fees (as they include greater technical risk and greater potential to increase long-term client revenue), followed by refueling operations, then repositioning, retirement, and standoff observation. The assumptions for revenues resulting from each type of operation are shown in Table 7.

Table 7. Client satellite data assumptions

Operation	**Revenue (V)**	**Probability of Success (p)**	**Operation Time**	**Time Penalty (β)**
Observe	\$10M	99.9%	10 days	0
Repair	\$25M	99%	30 days	\$100,000
Reposition	\$10M	99.5%	20 days + transfer time	\$100,000
Retire	\$10M	99.5%	20 days + transfer time	0
Refuel	\$15M	99%	30 days	\$100,000

The probability of successfully completing each type of robotic operation was also considered. GEO robotic servicing has not yet been attempted, therefore no statistical data exists. The reliability of refueling operations in low Earth orbit, including rendezvous, docking, and propellant transfer, have been estimated to range from 95-99 percent.[13] Failure of a servicing operation can imply different results, including failure to grapple the client, failure to accomplish the intended operation, or damage to the client and/or servicer. Consequences of these failures range from time delays to financial penalties to total loss of the servicer. Here, we consider the most severe case: catastrophic damage to the servicer and inability to complete subsequent operations in the mission sequence. We assume the risk of this type of failure is very low; the assumptions for probabilities of success for the various operations are among those listed in Table 7.

Also note, in Table 7, that operation times are included as client data. The transfer times included in the Reposition and Retire operations are internal transfer times; these two operations in particular require orbital transfers. This is not encompassing of the transfer time of the servicer to clients that require these operations. All client satellites will have an associated time of transfer of the servicer to it from the previous client but only retirements and repositionings will have the additional transfer time during the operation.

Finally, it was assumed that the client and service operator agree to a pricing model in which the client must be serviced within a fixed time frame. If the operation is not completed on time, the contract specifies that the client's fee is reduced by a penalty to account for lost operating revenue.[14] A penalty of $100,000 per day was assumed for the more time-critical operations: repair, reposition, and refueling. This penalty comes from the yearly revenue of a satellite in GEO.

D. Sequencing results

Finally, the entire cost function could be used to evaluate the problem of multi-satellite servicing using the client data assumptions in Table 7. The first case run assumed that the time penalty of all satellites is taken into account from the launch of the servicer; T_{allow} was set to 0.

Table 8. GA results with full objective function and $\boldsymbol{T_{allow} = 0}$ *days*

Sequence No.	**Longitude (°)**	**Operation**	**Operation & Transfer Time (days)**	**Total Time (days)**
1	244.8	Repair	30	30
2	261	Repair	38	68
3	265	Repair	32	100
4	319.5	Repair	58	158
5	301.88	Observe	19	177
6	241	Observe	41	218
7	241.15	Retire	20	238
8	62.05	Retire	72.54	310.54
9	304.5	Retire	90.08	400.62
10	304.51	Retire	21.01	421.64

The solutions all had the first four operations as repairs. Now recall from Table 7 that repairs have the highest revenue but also high time penalty and lower probability of success. After the first four repairs, it was no longer beneficial to perform an operation with such a high time penalty. This is why there are not any refueling operations. Instead, the servicer jumped to the operation with the highest probability of success, observations. Because there is no time penalty associated with the rest of the sequence, the longitude of the satellites no longer had an effect on those chosen. Starting at the 5th satellite in the sequence, the observations were chosen first because they were the least risky. After that, the rest of the sequence was retirements.

The second case was that of $T_{allow} = 120$ days. This case assumes that the clients offered a grace period for servicing their satellites before the time penalty became a factor. The GA results are shown in Table 9.

Table 9. GA results $\boldsymbol{T_{allow} = 120}$ days and time penalty from client perspective

Sequence No.	**Longitude (°)**	**Time Penalty ($/day)**	**Operation**	**Operation & Transfer Time (days)**	**Total Time (days)**
1	244.8	100,000	Repair	30.0	30.0
2	261	100,000	Repair	38.0	68.0
3	265	100,000	Repair	32.0	100.0
4	319.5	100,000	Repair	58.0	158.0
5	95	100,000	Repair	98.0	256.0
6	301.88	0	Observe	87.0	343.0
7	241	0	Observe	41.0	384.0
8	241.15	0	Retire	20.0	404.0
9	341.99	0	Retire	49.5	453.5
10	169	0	Retire	74.8	528.3

Many trials were run and the general trends were that the sequence began with the high revenue, higher risk satellite operations and then moved on to low risk observations and finished with retirements, similar to the first cast without a grace period. The T_{allow} is intended to represent a client offering the servicer four months to operate on their satellite before lowering the payment due to tardiness. The common effect of the added grace period was that the sequence selected more high revenue satellites to service than in the cases without a grace period. This makes sense because delaying a monetary penalty allows for more of the higher time penalty satellites to be serviced. This higher allowable time for servicing also diversified the selections for the early slots to include lower revenue operations if it made sense for timing. This is in contrast to the T_{allow}=0 cases that were run and resulted in all repairs in the first four slots and then immediately to observations in the 5th slot. For all cases, after the time-critical operation satellites were serviced, the serviced moved on to service both observations with retirements pulling up the end of the sequence.

Both of the cases assumed values of the time penalty for each operation β_i based on the urgency of the client's servicing need. A time penalty of $\beta = 0$ was assumed for all observation and retirement operations, as these are less likely to be time-critical operations from the client's perspective. However, this assumption does not fully capture the time constraints of the servicer. Sequences that result from this assumption include a random selection of retirement clients at the end of the sequence. These clients can be widely distributed in longitude, which leaves the possibility for very long missions. Table 9 and Figure 3 illustrate a result with a no time penalty for the later satellites in the sequence.

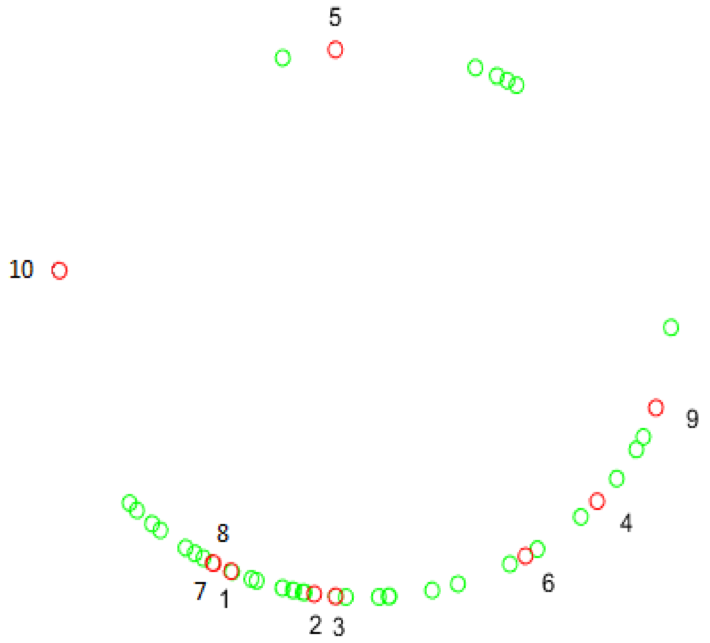

Figure 3. Visualization of results from Table 9

In Figure 3, the circles represent all of the satellites in the database. The numbered circles represent the satellites selected to be in the mission sequence, and the numbers represent the order in the sequence that these satellites appear.

Table 9 shows a common case of five repairs being selected at the beginning of the sequence due to their high revenue. This case also illustrates well the possibility of a long mission. The 5th satellite is a repair and it is thus worth it to make the long journey around the earth to reach it. However, making this journey adds a lot of time to the mission. This is exacerbated by the fact that the journey from the 5th to the 6th satellites is just as long plus jumping back and forth between the 6th and 10th satellites. This yields no monetary penalty for the mission because these last four satellites all have no time penalty, but it adds a lot of time. As shown, the mission would be 528 days long or approximately 17 months.

Approaching the time penalty from the perspective of the servicer is more realistic. Even though there might be no client operational requirement to perform observation or retirement operations within a limited time frame, there is a small time penalty for the operator to perform any operation, since they must employ ground support personnel and track the servicer throughout the mission. As such, when approaching the problem from the perspective of the servicer, a small time penalty of $\beta =$\$5,000 was added to the operations that previously were not assigned any time penalty. This slightly penalizes the time spent in transit to lower revenue satellites. Table 10 and Figure 4 show the sequence resulting from this assumption.

Table 10. GA results $\boldsymbol{T_{allow}} = \mathbf{120}$ days and time penalty from servicer perspective

Sequence No.	**Longitude (°)**	**Time Penalty ($/day)**	**Operation**	**Operation & Transfer Time (days)**	**Total Time (days)**
1	319.5	100,000	Repair	30.00	30.00
2	265	100,000	Repair	58.00	88.00
3	261	100,000	Repair	32.00	120.00
4	244.8	100,000	Repair	38.00	158.00
5	238.98	100,000	Refuel	33.00	191.00
6	241	5,000	Observe	11.00	202.00
7	241.15	5,000	Retire	20.00	222.00
8	301.88	5,000	Observe	28.05	250.05
9	304.51	5,000	Retire	22.00	272.05
10	304.5	5,000	Retire	21.01	293.06

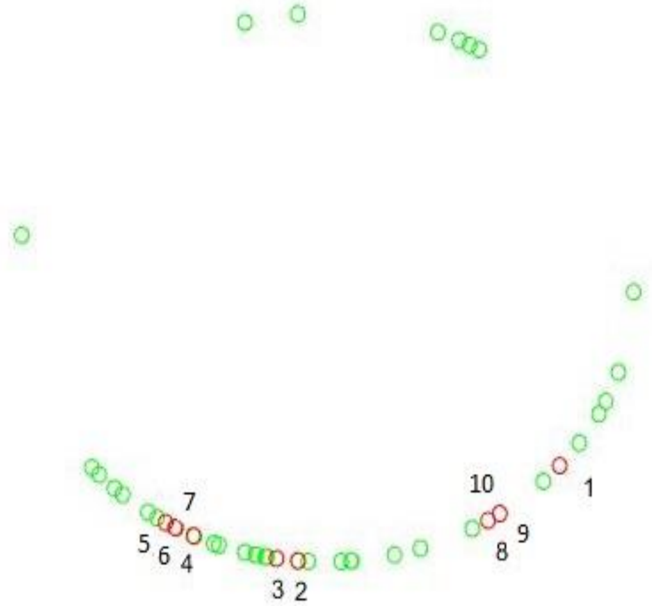

Figure 4. Visualization of results from Table 10

The difference between the two time penalty perspectives is most apparent in Figure 4. There are no outliers in that sequence and the numbers are grouped together much more than in Figure 3. Note, even, that the small time penalty yields a higher benefit to grouping satellites 4-7 and 8-

10 than in selecting both observations as soon as the higher revenue and more time-critical are finished being operated on. This drops the mission time by approximately five months compared to the client-perspective time penalty.

Table 10 represents perhaps the most realistic mission design for a robotic servicer under the given assumptions. The mission should begin with the highest-revenue operations, repair and refueling. When high-revenue operations can no longer be completed in time to satisfy client time limits, the servicer should proceed with the lowest-risk operations, standoff observation. Near the end of the servicer's mission, it should perform client retirement maneuvers (which has the additional benefit of allowing the servicer to remain in disposal orbit after the final operation). These should all be performed while minimizing transfer times be operating in order of longitudinal position.

E. Dynamic client model

Next, we consider the case where a satellite is added to the database when the servicer is already in the middle of a mission. While this could happen with any type of satellite operation, the trials were run considering the added satellite to be a higher revenue operation: repair or refuel. We assume the servicer is operating under a contract with the 10 satellites in sequence when a new client approaches the operator with an urgent servicing need.

Analysis of this case is straightforward. The new satellite is added to the set of remaining clients and the algorithm determines if it is economically beneficial to add this new satellite to the sequence. The added satellite's time penalty counter does not start with the rest of the satellites in the sequence, at the start of the sequence. Instead it starts at the time of being added to the database.

However, if it is not economically beneficial to add it to the sequence, the algorithm can ignore the new satellite and simply proceed with the sequence as originally planned.

To illustrate this case, consider that the servicer was on the mission outlined Table 10, in transit between the 3rd and 4th satellites, when the operator was approached with a new satellite. A higher-revenue satellite operation was chosen for the added satellite because it increases the likelihood it will be chosen, so its positioning in the sequence can be analyzed. The new satellite was a refuel operation at 295°E.

Table 11. Refuel operation added to database

Sequence No.	**Longitude (°)**	**Time Penalty ($/day)**	**Operation**	**Operation Time (days)**	**Total Time (days)**
i	319.5	100,000	Repair	30.00	30.00
ii	265	100,000	Repair	58.00	88.00
iii	261	100,000	Repair	32.00	120.00
ADD NEW SATELLITE	255	--	--	--	--
1	244.8	100,000	Repair	35.00	155.00
2	238.98	100,000	Refuel	33.00	188.00
3	241	5,000	Observe	11.00	199.00
4	241.15	5,000	Retire	20.00	219.00
5	295	100,000	Refuel	46.04	265.04
6	301.88	5,000	Observe	14.00	279.04
7	304.5	5,000	Retire	22.00	301.04
8	304.51	5,000	Retire	20.00	321.04

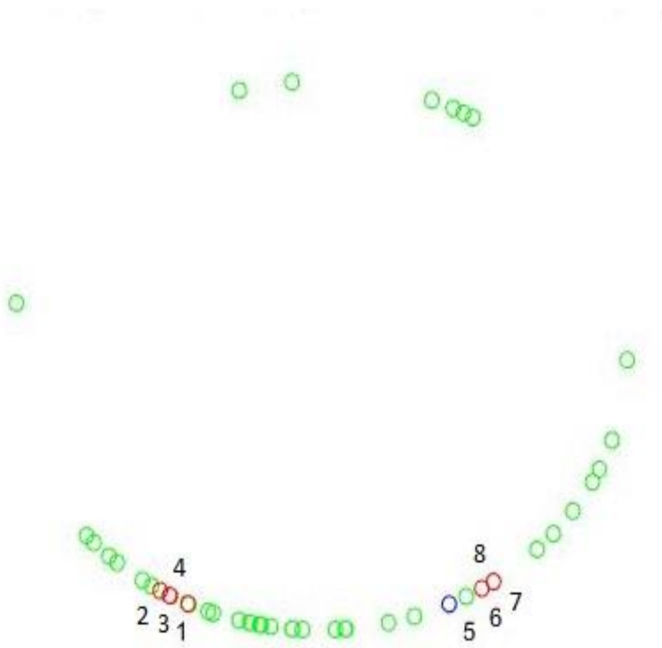

Figure 5. Visualization of results from Table 10

As shown in Table 11, the servicer was approached with a new potential client satellite when the servicer was a 255°E longitude. The GA determined that it was economically beneficial to add the satellite to the solution. Table 11 and Fig. 5 illustrate that the GA-determined best placement was in the 5th slot in the sequence after the satellite was added or the 8th slot in the sequence overall.

To address dynamic problems in which new clients arise throughout the servicer's lifetime, the GA could be re-run each time a new opportunity becomes available. It provides an assessment tool to determine if and how to re-route the servicer to maximize mission value, while accounting for any existing client contracts and time requirements.

V. SATELLITE SERVICING FUTURE WORK

There are a number of ways that facets of this problem could be expanded. Different types of operational failures could be considered, including failure to grapple the client satellite without damage, damage to the client, and damage to the servicer resulting in partial loss of functionality, as well as total servicer loss, as considered in this analysis. These different types of failure will have different probabilities of occurrence and different effects on the servicer's mission and profit, so further modifications of the cost function could be implemented. More complex trajectory dynamics could also be explored, including plane change maneuvers (to reach GEO satellites operating in inclined orbits). Propellant expenditures and constraints could also be considered. For instance, increasing the drift rate while in transit between client satellites would lower the transfer times and thus the time penalty. However, this would also increase the fuel usage and thus lower the number of possible clients that the servicer could operate on. A future GEO servicing robot may use low-thrust electric propulsion rather than the high-thrust chemical propulsion that was assumed for these preliminary results. To account for this, the trajectory maneuvers and propellant use could also be re-evaluated for low-thrust propulsion. Finally, including variation in client fees would more accurately represent real-world operations. For instance, it is unlikely that every repair would represent a $25M possible revenue; a repair in the first year of a satellite's life would likely be worth more than a repair in the 15th year. Similarly, delivering three additional years worth of fuel to a satellite would likely be worth more to the client than delivering a single year of fuel. Even within similar operations, stochastically varying client fees would also better represent the free market.

VI. INTRODUCTION TO THE OBSERVATIONAL ORBIT PROBLEM

The rest of this thesis will focus one level lower on a potential algorithm for determining how the servicer would perform an observation operation. An observation would consist of proximity operations where the servicer collects images of a client satellite. The analysis includes (1) methods of quantifying the effectiveness of an observational orbit, and (2) development of a genetic algorithm to optimize firing patterns for an observational satellite.

There are a number of methods for creating an orbit for one satellite (henceforth called the deputy) to observe another (henceforth called the chief). One method is the point-click method. This method treats the deputy like a tourist at an interesting landmark, the chief. The tourist knows exactly what they want to take pictures of and they position themselves accordingly. In other words, if there are specific features on the chief satellites that need imaging, the deputy can be programmed to travel to specified points around the chief and take pictures from those points. As shown in Figure 6, the deputy would execute very specific orbital maneuvers to travel from point to point to image the chief.

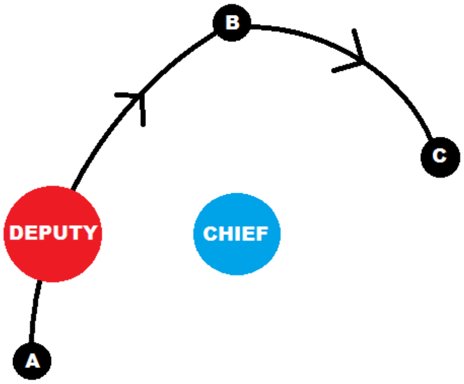

Figure 6. Visualization of the Point-Click Method of Satellite Imaging

Note that points A B, and C would not necessarily be coplanar as this is a problem in three dimensions.

Another method is the natural-orbit method. Recall that the chief itself would still be orbiting around the earth. Creating an observational "orbit" of the deputy around the chief is something like an illusion. The deputy would also be orbiting around the earth, but the relative motion of the deputy to the chief would give the appearance of the deputy orbiting the chief. There are natural orbits that would allow the two satellites to get in close proximity to one another. These orbits could be used by the deputy to image the chief. For instance, if the deputy were orbiting the earth in an orbit slightly lower than the chief, there would be a fly-by effect because the deputy would be orbiting slightly faster. A similar effect would be seen if the deputy were on a higher orbit. Another option would be to place the deputy on the exact same orbit as the chief, only slightly behind or ahead. This would allow the deputy to remain a constant and specified distance from the chief and observe constantly during orbit. These methods are practical if the chief is orbiting irrotationally, i.e. the sun is always in the same position relative to the chief. This would mean that the chief would be rotating in orbit relative to its ground position and the deputy would be able to image all sides from a stationary relative position to the chief. The problem with these natural orbits arises when the chief is an earth-pointing satellite. This would only allow the deputy, from a natural orbit, to image the chief from one position or with a limited viewing angle. A natural orbit solution to this problem would be to put the deputy on a slightly eccentric and inclined orbit such that it views all sides of the chief. The issue of time becomes prevalent here. At GEO, every orbit represents a full day. The deputy achieving a comprehensive view of the chief while relying on natural orbits could take a long time.

The method presented here represents a new method of determining relative orbits of a deputy to a chief.

VII. METHODS

A. Genetic algorithm

Another genetic algorithm was developed to determine the orbit of the deputy relative to the chief. The reason that a GA was used was that the approach to orbital maneuvering employed by the deputy for this method of satellite imaging has an astronomical number of possibilities. The observer GA developed could be used for any altitude orbit, but for the purpose of application to the satellite servicing problem, a GEO orbit was assumed. The problem that this observer GA solved was determining the timing and the direction of the thrusting burns on the deputy in order to optimize the deputy's path around the chief.

The orbit was approached in time steps of 15 minutes for a preliminary analysis. This time step could shrink to achieve a more accurate result or grow to decrease computing time. For each time step, three values have to be determined. The first value is a yes/no (1/0) value for firing the thrusters. The second two values are to determine the direction of thrust which is done by assigning right ascension and declination values relative to the frame fixed on the observational satellite. With four steps every hour, in a single day there are 96 time steps to determine yes or no for firing and then 360 possibilities of the degree measure of right ascension and 180 for declination for every time step that fires thrusters. The number of possibilities for solutions in just a single day is astronomical; for the just the binary choice of firing thrusters, without regard to direction, there are $2^{96} = 7.92 \times 10^{28}$ possibilities!

In the same way as with the satellite servicing problem, the GA randomly generates both good and bad chromosomes. For this GA, the first parameter determined is the amount of time for the deputy to observe the chief; this determines the size of a chromosome. A chromosome represents one possible burn pattern for the deputy's thrusters during the period of observation. For a single day with time steps of 15 minutes, a chromosome for a deputy/chief combination in GEO would have 96 rows and 3 columns; there would be one row for each time step and a column for yes/no firing, right ascension of thrust and declination of thrust as shown in Table 12.

Table 12. Content of a chromosome

Time Step	**Fire Thrusters?**	**Right Ascension [°]**	**Declination [°]**
1	1	253	48
2	1	245	44
3	0	--	--
4	1	176	32
5	0	--	--
. . .	. . .	. . .	. . .
96	0	--	--

These chromosomes are then evaluated for fitness and the natural selection method is employed to determine which should move onto the next generation. Again, this means that the chromosomes with higher fitness are more likely to produce "offspring" for future generations. Similar operations as in the satellite servicing GA are performed to slightly alter the chromosomes selected for the following generation and then that generation is reevaluated for fitness. This process is repeated and the chromosomes continue getting better until a stopping criterion is reached and the algorithm terminates. Also similar is the use of immigration to maintain genetic diversity. However, the complexity of each chromosome in the observer GA compared to the servicer GA merited an adaptation to be made to the immigration process. The nature of a GA holds that the immigrated chromosomes are unlikely to start with a high fitness. As such, the process of immigration for this GA does not occur every generation after a specified start. Instead, it occurs every 10 generations after a specified start. This allows the immigrated chromosomes to be operated and improved in their own right before they are emigrated out, if they are emigrated out. This gives the immigrated chromosomes a better chance at becoming a new leader.

B. Cost metrics

This observer GA follows perhaps a more conventional means of optimization than the satellite servicer GA because it goes about minimizing, instead of maximizing, an objective value. As such, for the observer GA, cost is an appropriate term for the metrics that quantify the fitness of a solution to the firing pattern problem because intuitively, cost is something that is desired to be minimized. Another difference between the two GAs was that the servicer GA had an objective function that was a single function with non-arbitrary values. Quantifying cost for observer GA was a more abstract process. As such, the cost metrics are arbitrary to the problem, but non-arbitrary relative to each other. Essentially, they are weighted costs that do not actually describe any monetary cost for the firing sequence. For the observer GA, there are four metrics of cost to be minimized. These metrics pertain to distance, illumination angle, observational coverage, and fuel consumption.

Before the distance could be assigned a cost, two parameters had to be set. The minimum distance desired from the chief to the deputy, to avoid potential collisions, and the maximum distance desired between the chief and the deputy, to maintain the highest quality images. The goal was to drive the deputy as close to the chief as possible without entering a region that merited concern for collision. This yielded the distance to be measured as a step function of three parts. The first part accounted for the case where the magnitude of the position vector, *r*, from the chief to the deputy had a magnitude smaller than the minimum desired distance, d_{min}, the second part accounted for the case where *r* is between the two desired distances, and the third part for when *r* was greater than the maximum desired distance, d_{max}. Being that the observer GA represents a first pass at the observational orbit problem, both bodies were treated as points rather than rigid bodies. This means that the position vector was essentially from the centers of the bodies. The cost metric for distance is shown in Equation 6.

$$C = \begin{cases} 1000(d_{\min} - r) & r < d_{\min} \\ r/2 & d_{\min} \le r \le d_{\max} \\ 2r & r > d_{\max} \end{cases} \tag{6}$$

The first part has the cost due to distance drastically increasing as the deputy gets closer to the chief. The second part has the cost drop significantly when *r* is in the desired range, but slightly increases as the deputy gets farther from the chief. The third part has a steeper increase as the deputy gets farther and farther outside of the desired range. Displaying this metric is difficult because the slope of the first step is so drastically steeper than the other two. Figure 7 illustrates conceptually the cost metric for distance.

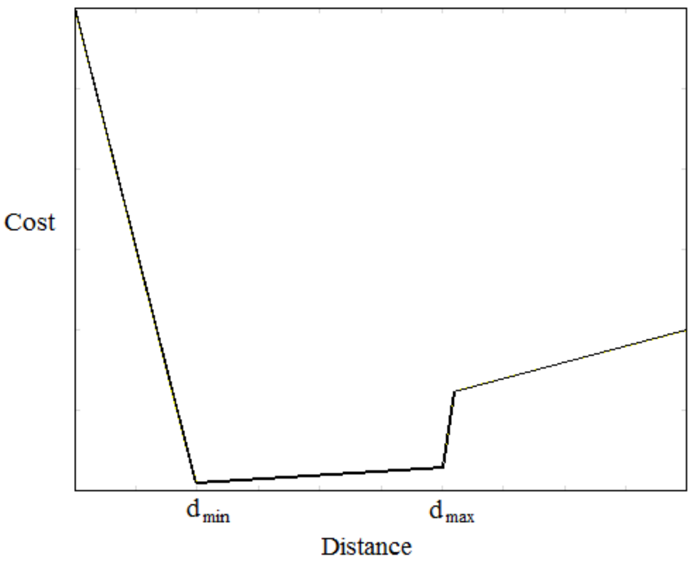

Figure 7. Distance-Based Cost Metric

Setting up the distance cost this way drives the deputy to d_{min} but still offers a range of distances with relatively low penalty. This gives the GA flexibility to sacrifice optimal distance for good-enough distance if it is required to significantly lower the cost of the other cost metrics.

The second metric was illumination angle. It is not desirable for the deputy to image a shadow. This metric drives the deputy to remain between the sun and the chief in order to be able to see the chief in its images and not just a water mark. This cost is described in Equation 7.

$$C = \begin{cases} \theta/10 & \theta < 90^\circ \\ \theta & \theta \geq 90^\circ \end{cases} \tag{7}$$

Where θ is the angle from 0-180° between the sun and the deputy measured from the chief. Figures 8 and 9 illustrate this angle.

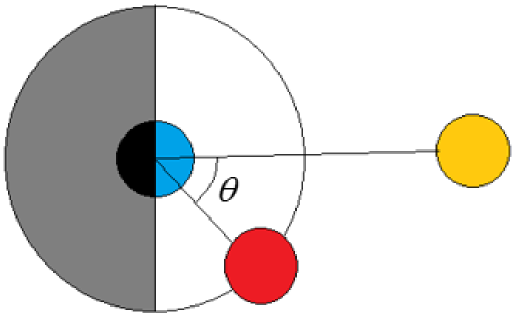

Figure 8. Visualization of Illumination Angle Cost Metric

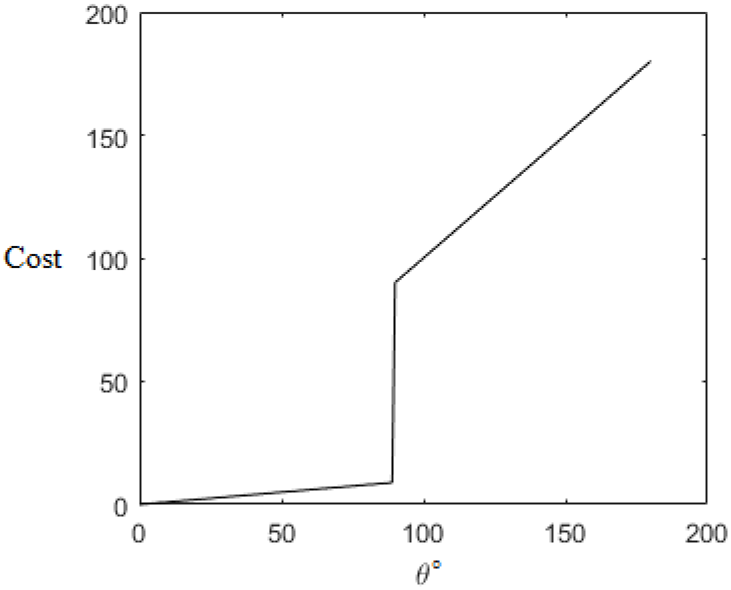

Figure 9. Cost vs Illumination Angle

Similar to the distance metric, there is an ideal scenario, when the sun is directly behind the deputy, and an area of leeway where the penalty is small, all points where θ < 90°, to allow the other cost metrics to minimize.

The third metric minimized was observational coverage. This is a binary cost metric. If the deputy was able to image the entire chief over the course of its observational "orbit", no cost penalty was enacted. If not, then a cost penalty was enacted to that specific chromosome. The criteria for full observational coverage has to do with the illumination angle and the allowable angle between useable images. Now due to time constraints during development, this metric was not fully transitioned with the others when the observer GA was taken from 2-dimensional to 3-dimensional. Each time step in a chromosome represented more than just a potential thruster burn, but also a potential slot for imaging the chief. For the entire orbital time, the algorithm tracked the angle of the deputy relative to the chief in 3-dimension. This angle was then projected onto the orbital plane and if the illumination of the chief was favorable for that time step, as shown in Figure 8, the angle of the deputy to the chief was stored. In Figure 8, θ will change for every time step throughout the orbit. All time steps with $\theta < 90°$ would be stored, as they represent good illumination images. After all of the useable images are stored by the algorithm, they are ordered from 0-360°. An absolute minimum of three images would be required to fully image a satellite; this is shown in Figure 10. As such, for the minimum requirement for observational coverage, it was determined that there not be a gap larger than 120° between any two useable images. This yields at least three images to image all 360° of the chief. This number could easily be changed based on the level of detail desired in the images of the chief. Note that at least a fourth image would be required to bring this cost metric into the third dimension.

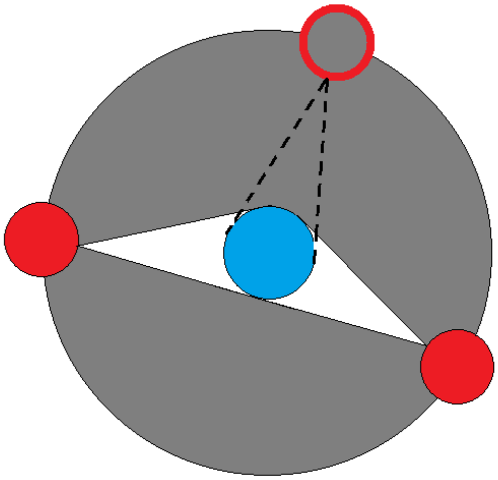

Figure 10. Visualization of Observational Coverage

The fourth and final metric to be minimized was fuel consumption. This metric is rather straightforward. For every time step that the deputy performs a burn, there is a small penalty. This drives the deputy to rely on natural orbits as much as possible but still gives the ability to change orbits when required, which would happen every time the thrusters are fired. This also lengthens the amount of time that the deputy can image the chief because it lowers the amount of fuel used per time.

The separate costs of all four metrics are added together and the total cost is assigned to that chromosome.

C. Orbit tools

For determining the actual orbit, two tools were used. The first was the governing equation for two-body orbital motion.

$$\ddot{\boldsymbol{r}}= \frac{\mu}{r^3} * \boldsymbol{r} \tag{8}$$

Where the bold ***r*** again represents the position vector of the satellite relative to the body it is orbiting, Earth, *r* is simply the magnitude of ***r***, and μ is the gravitational parameter of Earth. Equation 8 could then be numerically integrated, for every time step, from the initial orbit of the deputy and the chief. Once the position vectors for both satellites relative to the Earth were determined, the position of the chief needed only to be subtracted from that of the deputy to get the relative position of the deputy to the chief. The benefit of using Equation 8 was that it was absolutely correct with respect to the given information. The issue however was that this equation would have to go through the numerical integration process 100,000 or more times. This was extremely computationally cumbersome as the process of integration takes a long time.

One benefit to analyzing this problem in GEO is that the chief orbit can be assumed circular. As such, the Hill-Clohessy-Wiltshire (HCW) equations were employed. These are linear equations for evaluating the approximate position and velocity of one spacecraft relative to another shown in Equation 9[15].

$$\delta\ddot{x} - 3n^2\delta x - 2n\delta\dot{y} = 0 \tag{9a}$$

$$\delta\ddot{y} + 2n\delta\dot{x} = 0 \tag{9b}$$

$$\delta\ddot{z} + n^2\delta z = 0 \tag{9c}$$

An analytical solution can be found.

$$\delta x = 4\delta x_0 + \frac{2}{n}\delta \dot{y}_0 + \frac{\delta \dot{x}_0}{n}\sin(nt) - \left(3\delta x_0 + \frac{2}{n}\delta \dot{y}_0\right)\cos(nt) \tag{10a}$$

$$\delta y = \delta y_0 - \frac{2}{n}\delta \dot{x}_0 - 3(2n\delta x_0 + \delta \dot{y}_0) + \frac{2}{n}\delta \dot{x}_0\cos(nt) - 2\left(3\delta x_0 + \frac{2}{n}\delta \dot{y}_0\right)\sin(nt) \tag{10b}$$

$$\delta z = \frac{1}{n}\delta \dot{z}_0 \sin(nt) + \delta z_0\cos(nt) \tag{10c}$$

Where δx , δy, and δz are the components of the position vector from the chief to the deputy and n is the angular velocity or mean motion of the chief.

The relative position of the deputy to the chief can be found in the same way as it was when using Equation 8. The difference is that instead of integrating for every time step, the initial conditions can simply be plugged in to the analytical solution to the HCW equations and evaluated and used as the initial conditions for the next time step. Note that using the HCW equations meant that the two spacecraft needed to start relatively close to each other, within ~50 km, for the equations to be useful. This is because the HCW equations are a linearization of the relative motion of the two bodies, and they are only valid for close distances

The benefit of using the HCW equations as opposed to the fundamental orbit equation is computing time. The analytical solution to them could be used at every time step which drastically decreased the computing time because they did not require a process of integrating. However, the linearization also meant that the approximate solution got less and less accurate as the starting proximity of the spacecraft got larger. For an initial distance of more than about 50km, the HCW equations always put the deputy on a hopping orbit regardless of the initial conditions. This is non-ideal and thus the initial distance was typically kept closer to 30 km. Another consequence of the linearization is that the error in the HCW solution compared to the actual solution increased as the time of propagation increased.

VIII. RESULTS

The most important data to collect early on in the project was data to suggest how many generations of the GA needed to run to produce useable results. The observer GA was executed six times with the same initial conditions and set to run for 1,000 generations. The results were stored for the first, 100th, 500th, and final generation. The results stored were the chromosome with the lowest cost, or the best solution the algorithm had found at that point. The results showed that by the 100th generation, the cost of the best chromosome, or leader, of that generation had decreased to 95% of the final cost that the algorithm ended on. By the 500th generation, this number increased to 99.8%. For purposes of testing the GA, 95% was determined to be an acceptable figure and the algorithm was hence forth run for 100 generations at a time when using the governing orbital equation. The reason that this number is so much lower than for the GA used in the satellite servicing project comes back to computational time. The GA for satellite servicing can run through 1,000 generations in approximately 45 seconds. Using the HCW equations, this GA can run through 1,000 generations in approximately an hour. Using the governing orbital equation, 1,000 generations takes approximately six hours.

For this reason, the structure of the GA is to use both. The GA was structured in two levels. The first level runs the HCW equations for 1000 generations. Everything is initially randomly populated and then survival-of-the-fittest is employed to determine a leader. However, the determined leader is not the end of the algorithm. The solution for the firing pattern of the deputy obtained by HCW equations is used as a starting point. As specified before, the HCW equations come with inherent error, but are computationally easier than the governing equation. This GA captures the best of both worlds. The solution obtained by the HCW equations is then fed back into the second level of the GA. The first generation is not randomly populated for this second level of the GA; instead it is entirely populated by the solution obtained by the first level of the GA. The second level of the GA then can run for only 100 generations because it already has an excellent starting point.

The relative orbits of the deputy to the chaser are shown for three different scenarios, all of which have $d_{max} = 50$km and $d_{min} = 3$km. In order to present an analysis of the figures, the cardinal directions will be used to indicate positions on the figures.

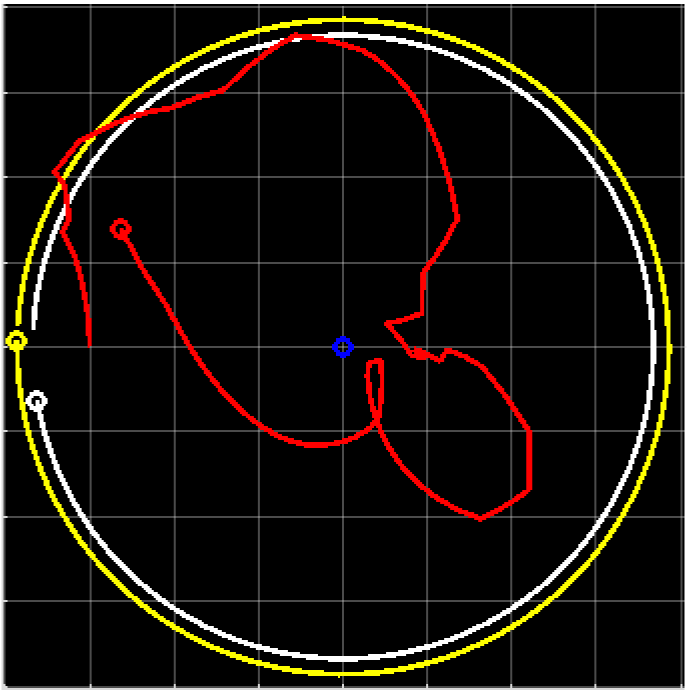

Figure 11. Deputy "orbit" around chief: ideal starting position

All of the figures image a reference frame fixed to the chief as it orbits the Earth. The circle in the center is therefore the chief satellite. The two circular "orbits" on the outside represent the relative “orbits” of the sun and the moon to the chief. The more jagged “orbit” is that of the deputy. Recall also that three out of the four cost metrics are three dimensional, so the path of the deputy does not stay in plane. However, this is the best way to display the path of the deputy while also showing the effect of the illumination angle metric, which is only two dimensional. Figure 11 represents the case where the deputy is inserted into a relative orbit beginning at an ideal position relative to the illumination of the sun and the moon. The small circles representing the satellites and celestial bodies are in their final locations. Every grid line represents 10km of distance between the deputy and the chief. Obviously the sun and moon are not only 40km away from the chief; those orbits are displayed as mere place-holders to illustrate the illumination. All three of the “orbiting” bodies began directly West of the chief. The “orbits” proceeded clockwise around the chief from that position. The starting distance of 30km for the deputy is within the bounds and the deputy maintains that distance for the first quarter of the “orbit” in the NW corner of Figure 11. In the NE corner, the deputy drives in close to the chief but in doing so sacrifices the ideal illumination it maintained in the beginning. This is the reason for the loop in the SE corner; the deputy waited for the sun and moon to catch up before proceeding back to a position closer to the chief.

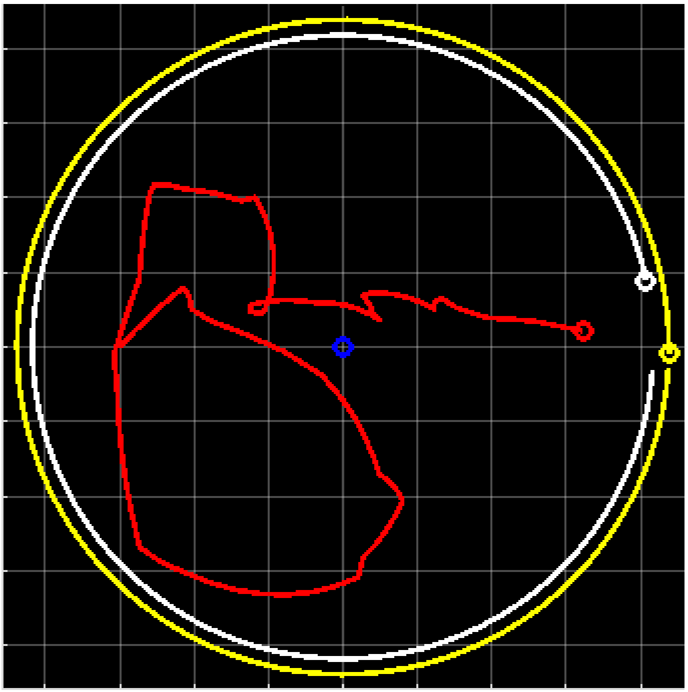

Figure 12. Deputy "orbit" around chief: non-ideal starting position

Figure 12 illustrates the case where the beginning of the observational orbit is non-ideal for the deputy with regards to illumination. The deputy begins directly West of the chief but both the sun and the moon begin directly East. The large loop in the SW corner of Figure 12 is the deputy waiting for the sun and the moon to offer a better illumination angle. That angle is achieved directly South of the chief and the deputy then follows the sun and moon for approximately a third of an orbit. Similar to Figure 11, the deputy then drives close to the chief.

One thing to note is that these figures represent a single orbit of the chief around the earth. This is why Figure 12 does not show an "orbit" with full observational coverage; the area in the SE of the figure where the deputy never reached. The deputy spent too much time reaching a point of good illumination in the beginning of its orbit to be able to get full converage in one orbit of the chief. Recall, again, that the HCW equations have error that get worse as time increases. For this reason, the algorithm is set to run for a single orbital period at a time. In order to get results for multiple periods, the GA would simply run single periods at a time and concatenate the results, each period starting from the ending point of the previous period. Note that the ending position of the "orbiting" bodies in Figure 12 is very similar to the starting position of the "orbiting" bodies in Figure 11. If another orbital period were to be run , it would turn out very similar to Figure 11 and the deputy would achieve full observational coverage

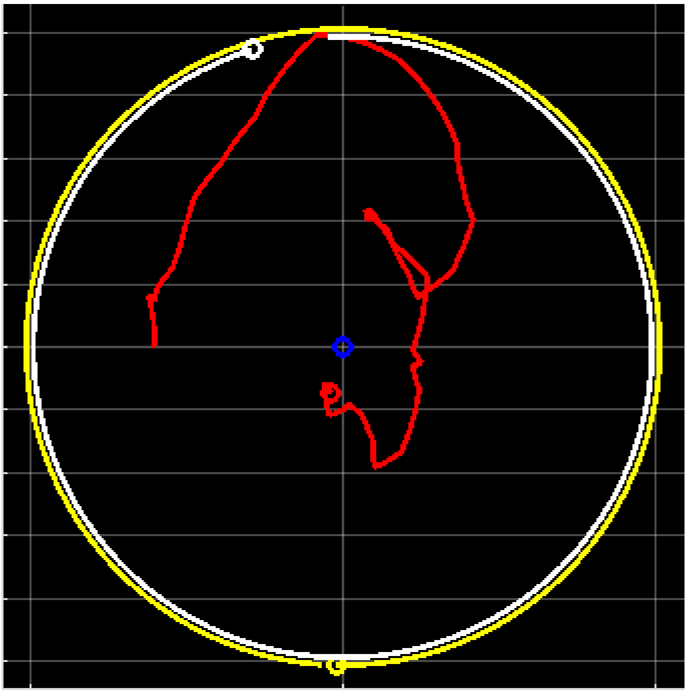

Figure 13. Deputy "orbit" around chief: worst-case-scenario starting position

Figure 13 illustrates what could be considered a worst-case-scenario for the starting position of the "orbital" bodies. The deputy begins directly West of the chief, the moon directly North and the sun directly South. The deputy tries to maintain an orbit in better position relative to the sun. The sharp loop in the NE corner shows where the deputy attempts to wait for the sun to "catch up" and then it continues around the chief. This is encouraging because the sun is weighted heavier than the moon in the cost function as it is significantly more illuminating.

IX. OBSERVATIONAL ORBIT FUTURE WORK

This problem was designed as an experiment in observational orbits. As such, there is no perfect solution, rather there are interesting results that could lead to future orbit planning. That being said, this project is still in its infancy. The results are promising and a step in the right direction, however there are many ways to improve the algorithm as well as the cost function.

For improving the cost function, finding a way of making the cost metrics non-arbitrary would be the most helpful change for presenting results. In making them non-arbitrary, they could likely also be weighted relative to each other with greater rigor. For instance, the multipliers for r and θ in the distance and illumination cost metrics could be varied depending on the capabilities of the observing satellite.

In improving the GA as a whole, a number of cost metrics would have to be added. One of more crucial metrics would be orientation of the deputy. If the deputy is going to be taking images of the chief, it would have to be pointing at the chief. Attitude control of the deputy is a large facet of future work on this project. Also, during observational missions where the deputy is driven close to the chief, collision of the deputy and chief becomes a very real concern. Protections are in place in the algorithm to prevent collisions, but those protections are useless if there is a malfunction in the thrusting system of the deputy. This breeds the need for passive safety. Passive safety protects against collision in the event of loss of thrust in the deputy. What this means is that the natural orbit of the deputy at any given time would have to be able to be propagated out two days without any additional thrust without coming within collision distance of the chief. With the

GA structured as it is now, this would likely add a third level, after the governing orbital equation is used, to determine the final orbit. Passive safety is the highest priority for future work. The enormous cost of a collision in orbit makes this a top priority for real world trajectory design for this sort of mission.

As it stands, the time of observation is not set by the GA but the user. An additional metric of the time required to fully observe a satellite based on the other metrics could be useful. This would hopefully decrease the cost as the deputy could perform observing maneuvers without having to sacrifice any of the cost metrics in order to make the time allotment. That being said, one of the advantages that this method has over the other methods mentioned is its ability to determine an orbit in a constrained time. Finally, the assumption was made that every burn of the thrusters was identical in time and magnitude. Adding the option to vary the amount of thrust during each time step would also open up the possibilities for the orbits that the deputy could attain.

X. CONCLUSIONS

Technical advances indicate that GEO robotic satellite servicing will become a reality in the near future. Economic analyses indicate that multi-satellite GEO servicing will be a financially lucrative market. However, the amount of profit to be made by a robotic servicer can vary greatly depending on the satellites serviced. The results achieved in this investigation demonstrate that operational sequencing is an important consideration in planning multi-client missions. The order of a servicing sequence can greatly impact the profit of a mission as well as the general safety of the robotic servicer. The servicer genetic algorithm and cost function proved to be effective in identifying high-value mission designs.

The highest value sequences will begin operations with the highest revenue satellites. Once all the highest revenue satellites are exhausted, or the time delay of service grows too large to make the operation profitable, the servicer should move on to the lowest risk operations. From there, the servicer should end its mission with retirement operations so that it can dispose of itself in the graveyard orbit along with its last client.

One level lower is where the observer genetic algorithm takes over in determining the trajectories for the observational orbits. The algorithm created trajectories that began by achieving favorable illumination before driving the deputy satellite into the ideal distance from the chief. This method was effective but would require further development if it were to be considered as useable in lieu of the methods already being used today.

REFERENCES

[1] Sullivan, B. R., Akin, D. L., and Roesler, G., "A Parametric Investigation of Satellite Servicing Requirements, Revenues, and Options in Geostationary Orbit," *AIAA SPACE 2015 Conference and Exposition*, Pasadena, CA, 2015.

[2] Defense Advanced Research Projects Agency, "Program Aims to Facilitate Robotic Servicing of Geosynchronous Satellites," March 25, 2016.

[3] Benedict, B. L., "Rationale for Need of In-Orbit Servicing Capabilities for GEO Spacecraft," *AIAA SPACE 2013 Conference and Exposition*, San Diego, CA, 2013.

[4] Walker, J., Defense Advanced Research Projects Agency, "Orbital Express," March 2007.

[5] Rhian, J., "NASA to Launch Robot Gas Station Experiment on Final Shuttle Flight," Space.com, June 13, 2011. http://www.space.com/11957-nasa-robot-space-refueling-experiment-final-shuttle.html

[6] Satellite Industry Association, "State of the Satellite Industry Report," September 2015.

[7] Sullivan, B.R. and Akin, D. L. "Satellite Servicing Opportunities In Geosynchronous Orbit," in *AIAA SPACE 2012 Conference and Exposition*, Pasadena, CA, 2012.

[8] Conway, B. A., Chilan,C. M., and Wall, B. J. "Evolutionary principles applied to mission planning problems,"

Celestial Mechanics and Dynamical Astronomy, Vol. 97, No. 2, pp. 73–86, 2007.

[9] Englander, J. A., Conway, B. A., and Williams, T. "Automated Mission Planning via Evolutionary Algorithms,"

Journal of Guidance, Control, and Dynamics, Vol. 35, No. 6, pp. 1878–1887, 2012.

[10] Morimoto, M., Yamakawa, H., Yoshikawa, M., Abe, M. and Yano, H., "Trajectory design

of multiple asteroid sample return missions," *Advances in Space Research,* Vol. 34, No. 11, pp. 2281–2285, 2004.

[11] Sullivan, B., Kelm, B., Roesler, G., and Henshaw, C. G., "DARPA Robotic Space Servicer: On-Demand Capabilities in GEO," in *AIAA SPACE 2015 Conference & Exposition,* Pasadena, CA, 2015.

[12] Sullivan, B. R., "Technical and Economic Feasibility of Telerobotic On-Orbit Servicing," PhD Dissertation, Department of Aerospace Engineering, University of Maryland, 2005.

[13] Cirillo, W. M., Stromgren, C., and Cates, G. R. "Risk Analysis of On-Orbit Spacecraft Refueling Concepts," in *AIAA SPACE 2010 Conference & Exposition,* Anaheim, CA, 2010.

[14] Dastidar, R. G., "On Orbit Satellite Servicing from a Vehicle Routing Viewpoint," in *AIAA SPACE 2010 Conference & Exposition,* Anaheim, CA, 2010.

[15] Curtis, H. D., *Orbital Mechanics for Engineering Students,* Butterworth-Heinemann, Oxford, UK, 2014