Optimal Reconfigurations for Increasing Capacity of Communication Satellite Constellations

Afreen Siddiqi and Jason Mellein
16.888 MSDO Final Paper, Spring 2004
Massachusetts Institute of Technology, Cambridge, MA 02139, USA

System reconfiguration can be a means of fulfilling new requirements that may arise over time. In the case of communication satellite constellations, change in market conditions, mission requirements etc. may necessitate increasing the lifetime capacity of the deployed constellation. Expanding the capacity can deliver greater benefit and may extend the useful lifetime of the system. This paper explores optimal inter-satellite and intra-satellite reconfigurations that can expand the lifetime capacity. In Inter-satellite reconfiguration, changes in spatial relationships between satellites of a constellation through orbital changes and addition of new satellites were studied in order to increase capacity. In Intra-satellite reconfiguration for capacity expansion, the optimal reconfigurations of components onboard satellites were analyzed. The tools produced through this study can be used in designing and planning for reconfigurable constellations. Furthermore, the results obtained from studying intra-satellite reconfiguration provide direction for selecting and investing in key technologies for the future in order to enable commercially feasible realization of reconfiguration spacecraft.

Nomenclature

- \( A \) = Initial Constellation
- \( B \) = Reconfigured Constellation
- \( C \) = Lifetime Capacity [minutes]
- \( D_o \) = Transmit antenna diameter, [m]
- \( G_r \) = Receiver gain
- \( h \) = Altitude [km]
- \( J \) = Optimization objective
- \( N \) = Total number of satellites in constellation
- \( P_t \) = Transmitter Power, [W]
- \( R \) = Single user data rate, [kbps]
- \( T_{sat} \) = Satellite lifetime [yr]
- \( U_{sat} \) = Number of users per satellite
- \( x \) = Design vector
- \( \varepsilon \) = Minimum elevation angle [deg]
- \( \lambda \) = Objective function weighting factor

Introduction

Communication satellite constellations are normally designed for a fixed capacity/demand requirement. Recent economic troubles of Iridium and Globalstar however have shown it is often very difficult to predict the actual demand at the time when the constellation becomes operational. De Weck, de Neufville, and Chaize [1] have shown that evolving a large system in stages can mitigate the risks associated with such uncertain markets for capital-intensive projects. A case study of a Low Earth Orbit (LEO) communication satellite system shows staged deployment provides an “as-needed, as-afforded” approach which may allow system managers to delay decisions until there is greater certainty of market requirements [1].

One staged deployment strategy is to reconfigure the deployed constellation as the capacity demand evolves over time. Constellation evolution is considered for meeting a higher capacity requirement, although in a strict sense the evolution should ideally be such that the capacity may increase or decrease to match the market/mission needs. However, it only makes sense to reconfigure constellations for ‘growth’, meeting higher capacity demands. The ability to reconfigure for higher capacity would allow the exploitation of unforeseen market opportunities (or mission needs) providing greater economic (or other) benefits to stakeholders.

To determine the optimal evolution of a satellite constellation, de Weck et al [1] have developed a framework through which paths of architectures can be computed showing how a constellation can be optimally grown in stages. Figure 1 illustrates such a path for a particular case.

The various architectures in the “evolution path” are compared to the Pareto front of life cycle cost (LCC) and market capacity, and system optimality may decrease as it grows, i.e. its distance from the Pareto front increases. However, it may be more relevant to consider the Pareto optimality of reconfiguration cost (instead of LCC) vs capacity for a deployed constellation. Reconfiguration cost
is the cost incurred in reconfiguring an initial constellation to a different constellation with higher capacity. This paper presents a study for determining a new optimal constellation for an existing constellation such that the reconfiguration cost is minimized and the desired higher capacity requirement is met.

**Inter-Satellite Reconfiguration**

Inter-satellite reconfiguration means changing the spatial relationships between individual satellites of a constellation via orbit modification. This section studies the problem of determining the optimal target constellation for a specific on-orbit constellation given some higher target capacity.

**Case Study-I**

It is assumed that a communication constellation, $A$, is deployed at an altitude, $h_A$, of 2000 (km) with the minimum elevation angle, $\epsilon_A$, of the satellites at 5°. Furthermore, it is a polar constellation with single fold, global coverage, which corresponds to 21 satellites in 3 orbital planes, with 7 satellites per plane. All the satellites are assumed identical. The communication subsystem provides a total system lifetime capacity, $C_A$, of 1.5 x 10^{10} (minutes).

System lifetime capacity, $C$, is computed as:

$$C = N U_{sat} T_{sat} 365 \times 24 \times 60$$  \hspace{1cm} (1)

Where $N$ is total number of satellites in the constellation, $U_{sat}$ is the number of users per satellite, and $T_{sat}$ is the satellite lifetime (in years). Note the capacity is the number of user minutes the constellation provides over its lifetime.

1. **Problem Formulation**

Suppose market/mission changes necessitate an increase in the capacity of the deployed constellation to $1 \times 10^{11}$ minutes. The goal is to find a new constellation $B$ for constellation $A$ to reconfigure to while minimizing reconfiguration cost and meeting the target capacity. This optimization problem is stated more formally as:

$$\begin{align*}
\text{min} & \quad J = \text{reconfig cost} \\
\text{s.t.} & \quad 400 \leq h \leq 2000 \text{ km} \quad (2) \\
& \quad 5 \leq \epsilon \leq 20 \text{ deg} \\
& \text{and} \\
& C_B = C_{\text{desired}}
\end{align*}$$

The side constraints are imposed on the altitude since it is assumed the constellation is LEO. Since inclination changes are fuel intensive, the elevation angle is bounded between reasonable values.

Altitude and elevation angle drive the polar constellation parameters (with the assumption of single fold global coverage), therefore $h$ and $\epsilon$ were chosen as the constellation $B$ design variables to be optimized. The design vector is thus, $\mathbf{x} = [h, \epsilon]$. 

---

**Constellation Reconfiguration**

The transformation of an initial constellation to a new constellation of higher capacity involves three steps:

- Transfer of on-orbit satellites from locations in an existing constellation to those in the new constellation.
- Launch (addition) of new satellites from the ground.
- Reconfiguration of satellite components (such as adjustment of beam patterns as required by the new constellation architecture).

The first two parts are categorized as inter-satellite reconfiguration involving orbital transfers of on-orbit satellites and addition of new satellites to the new constellation. The third part, on the other hand, is intra-satellite reconfiguration meaning components of individual satellites are reconfigured to deliver new or modified functionality. It is clear a constellation would require some degree of both inter- and intra-satellite reconfigurability to evolve from one stage to the next. In this paper however, the two categories are analyzed separately in two specific case studies. The results presented in the two cases provide upper bound estimates of the reconfiguration requirements for the two types and tools for determining the best or optimal constellation reconfiguration to meet changes in capacity demand.
2. Simulation Models and Benchmarks

The simulation model is shown in Figure 2. Some of the modules are modified forms of modules developed by Darren Chang and Olivier de Weck [2].

Scialom [3] has studied the technical feasibility and optimality of orbit reconfigurations. He addressed the problem of determining the optimal orbital maneuvers to move satellites from their initial positions to final positions in the new constellation minimizing the total \( \Delta V \) requirement for the reconfiguration. In his study the orbital parameters (altitude, elevation angle, and constellation type) were known for both the initial and final constellations, and the aim was to determine how to best assign satellites from constellation \( A \) to positions in constellation \( B \). The Auction Algorithm [4] was used to determine the optimal assignments. Since additional satellites would launch from the ground to grow the constellation, the final ‘optimal assignments’ were determined by considering launch issues such as launch vehicle capacity and the orbital planes to which the ground satellites were assigned.

The assignment optimizer (depicted within the Function Evaluator block in Figure 2) is based on Scialom’s work [3] and is used in evaluating the reconfiguration cost of each candidate design vector (proposed by the top-level optimizer). The optimization is thus bi-level.

It is important to note that in this particular study, the launch issues were not considered due to some problems with the launch module code. To simplify, a one-vehicle launch capacity was assumed. Therefore, no re-assignment was done once the auction algorithm had determined the optimal assignments based on \( \Delta V \) considerations alone.

Constellation Module

The constellation module computed the number of satellites and orbital planes for an optimally phased, polar constellation with co-rotating planes given its altitude, elevation angle etc. Table 1 shows the inputs and outputs of the module. Calculations were based on equations developed by Adams and Rider [5]. The module was implemented by de Weck.

Table 1: Inputs and outputs of constellation module

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( h ) [km]</td>
<td>Circular orbital altitude</td>
</tr>
<tr>
<td>( \alpha ) [deg]</td>
<td>Minimum elevation angle</td>
</tr>
<tr>
<td>( n )</td>
<td>n-fold multiple coverage</td>
</tr>
<tr>
<td>( \varphi )</td>
<td>Min latitude above which coverage is guaranteed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outputs</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( T )</td>
<td>Total # of satellites in constellation</td>
</tr>
<tr>
<td>( s )</td>
<td># of satellites per orbital plane</td>
</tr>
<tr>
<td>( p )</td>
<td># of orbital, polar planes</td>
</tr>
<tr>
<td>( \alpha ) [deg]</td>
<td>Vector of angular separation of right ascending nodes (RAAN) of neighboring planes</td>
</tr>
</tbody>
</table>

Table 2: Inputs and outputs of Astrodynamics module

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( h_a ) [km]</td>
<td>altitude of satellite in constellation A</td>
</tr>
<tr>
<td>( e_a ) [deg]</td>
<td>min elevation of satellites in constellation A</td>
</tr>
<tr>
<td>( h_b ) [km]</td>
<td>altitude of satellite in constellation B</td>
</tr>
<tr>
<td>( e_b ) [deg]</td>
<td>min elevation of satellites in constellation B</td>
</tr>
<tr>
<td>( \text{prop} )</td>
<td>propulsion system type</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outputs</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( d ) [km/s]</td>
<td>matrix of ( \Delta V )s needed to move each satellite from A to a position in B</td>
</tr>
<tr>
<td>( t ) [s]</td>
<td>matrix of transfer time needed to move each satellite in A to a position in B</td>
</tr>
</tbody>
</table>

Launch Module

This module (implemented by Darren Chang) uses mass and volume of satellite, satellite altitude, and minimum elevation angle as inputs, and produces a recommendation of types of launch vehicles to use, launch sites, cost, availability, etc. The data source is International Reference Guide to Space Launch Systems, 1999, published by AIAA.

Fuel Module

This module computes the total propellant mass needed for transferring the satellites given a particular assignment matrix and the type of propulsion system. The assignment matrix treats the assignment of each satellite of \( A \) to a slot in constellation \( B \). It has two inputs: the assignment matrix and propulsion system type; and one output: the mass of propellant needed.
**Cost Module**

The cost module computes the reconfiguration cost for transforming constellation $A$ into $B$. The reconfiguration cost is a simple sum of the fuel cost, launch cost, and additional satellite production cost. It does not include costs from the use of special hardware, or lost revenue due to communication outage during the reconfiguration process. The fuel cost is determined from the mass of fuel needed for moving satellites $A$ into slots of $B$. The cost of manufacturing additional satellites is calculated while accounting for learning curve factors. Once the launch module is updated and the added capacity of the launch vehicle is considered launch costs will decrease.

**Link Budget Module**

This module determines the system lifetime capacity and evaluates the capacity constraint. A number of inputs are treated as constants (such as bandwidth, receiver gain, frequency spectrum etc). Two inputs to the module that change during the course of optimization are altitude and slant range. Slant range is a function of the elevation angle. Capacity is thus affected as altitude and elevation are varied during the course of optimization, and only those combinations of altitude and elevation are selected by the optimizer that correspond to the desired capacity. The following values for the link constants (inputs) were:

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Actual</th>
<th>Simulated</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISL</td>
<td>1 (yes)</td>
<td>inter-satellite links</td>
</tr>
<tr>
<td>$D_a$</td>
<td>1.5</td>
<td>antenna diameter [m]</td>
</tr>
<tr>
<td>$P_t$</td>
<td>400</td>
<td>transmit power [W]</td>
</tr>
<tr>
<td>Linkmargin</td>
<td>16 dB</td>
<td></td>
</tr>
<tr>
<td>$G_r$</td>
<td>1</td>
<td>receiver gain</td>
</tr>
<tr>
<td>$R$</td>
<td>4.8</td>
<td>single user data rate [kbps]</td>
</tr>
<tr>
<td>$T_{sat}$</td>
<td>5</td>
<td>satellite life time [yrs]</td>
</tr>
<tr>
<td>MS</td>
<td>QPSK</td>
<td>modulation scheme</td>
</tr>
<tr>
<td>$P_e$</td>
<td>0.5</td>
<td>satellite telephone power [W]</td>
</tr>
<tr>
<td>MAM</td>
<td>CDMA</td>
<td>multiple access method</td>
</tr>
<tr>
<td>incli</td>
<td>90</td>
<td>maximum inclination (latitude)</td>
</tr>
<tr>
<td>$\eta_s$</td>
<td>.55</td>
<td>satellite transmitter efficiency</td>
</tr>
<tr>
<td>$\eta_r$</td>
<td>.55</td>
<td>satellite transmitter efficiency</td>
</tr>
<tr>
<td>$D_k$</td>
<td>0</td>
<td>discount rate (%)</td>
</tr>
<tr>
<td>$G_b$</td>
<td>2.08</td>
<td>width of typical guard band</td>
</tr>
<tr>
<td>$\Delta f_c$</td>
<td>40</td>
<td>individual channel bandwidth [kHz]</td>
</tr>
<tr>
<td>$F_{m_link _ub}$</td>
<td>1.62650</td>
<td>upper bound mobile/satellite uplink frequency [GHz]</td>
</tr>
<tr>
<td>$F_{m_link _ub}$</td>
<td>1.61600</td>
<td>lower bound mobile/satellite uplink frequency [GHz]</td>
</tr>
<tr>
<td>BER</td>
<td>$10^{-3}$</td>
<td>bit-error rate</td>
</tr>
</tbody>
</table>

### Table 3: Constants used in link design

**Benchmarks**

The modules in the framework were validated by using data from a known communication satellite constellation, Iridium. Table 4 shows the model results alongside the actual values of the Iridium constellation which show good agreement.

### Table 4: Benchmark of simulated results with Iridium data

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Actual</th>
<th>Simulated</th>
</tr>
</thead>
<tbody>
<tr>
<td># of satellites</td>
<td>66</td>
<td>66</td>
</tr>
<tr>
<td>Frequency [GHz]</td>
<td>1.6212</td>
<td>1.6213</td>
</tr>
<tr>
<td># of orbital planes</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>Altitude [km]</td>
<td>780</td>
<td>780</td>
</tr>
<tr>
<td>Power flux [Jy]</td>
<td>$10^{10}$</td>
<td>$4.95 \times 10^{-11}$</td>
</tr>
<tr>
<td>Transmit gain [dB]</td>
<td>24.3</td>
<td>25.57</td>
</tr>
<tr>
<td>Transmit power [W]</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>Bandwidth [MHz]</td>
<td>10.5</td>
<td>10.5</td>
</tr>
<tr>
<td>FDMA channels</td>
<td>240</td>
<td>250</td>
</tr>
<tr>
<td>Voice circuits/sat</td>
<td>1110</td>
<td>1792</td>
</tr>
<tr>
<td>System lifetime capacity [mn]</td>
<td>$4.5 \times 10^{11}$</td>
<td>$3.1 \times 10^{11}$</td>
</tr>
</tbody>
</table>

### 3. Design Space Analysis

Since there were only two design variables and the evaluation of the objective was not very expensive, a full factorial computation of the design space was carried out. The altitude varied in 50 km increments from 400 to 2000 km, and the elevation angle was incremented by 1° from 5° to 20°. Reconfiguration cost for each design was computed (Figure 3).

![Figure 3: Orbit Reconfiguration Design Space](image)

Note that the plot does not show the ‘feasible’ design space, rather it shows how the objective behaves in the altitude and elevation angle space within the bounds of the problem. The feasible design space is a subset of this space.
It is clear the design space is highly non-linear. The sharp jumps or steps in reconfiguration cost are due to the discrete changes in the number of planes and satellites occurring as altitude and elevation angle values are varied (Figure 4).

The SQP method was chosen because all the design variables in the problem are continuous, and the constraint is non-linear. Simulated Annealing was chosen because it is less computationally intensive compared to Genetic Algorithms and is better established and understood than other heuristic techniques such as Tabu Search and Particle Swarm Optimization. Both algorithms were implemented in MATLAB.

**Gradient Method - SQP**

**Results**
The following parameters were set:

**Constellation A:**
- altitude, \( h_a = 2000 \) (km)
- elevation angle, \( \epsilon_a = 5 \) (deg)
- 21 satellites
- 3 planes
- 7 satellites per plane
- Target Capacity: \( 1 \times 10^{11} \) minutes
- tolerance = \( 10^9 \) minutes

The best solution found was a constellation **B** with altitude, \( h_b \), of 1066 (km) and minimum elevation angle, \( \epsilon_b \), of 5° (\( x^* = [1066 5] \)) which corresponds to 40 satellites and 5 orbital planes for single fold, global coverage. The reconfiguration cost, \( J^* \), is \( 587 \) million. Figure 6 and Figure 7 and show the schematics of the initial and optimal constellations.

**4. Single Objective Optimization Results**
The optimization was done with two different methods: Sequential Quadratic Programming (SQP), a gradient based method, and Simulated Annealing which is a heuristic technique.
space surface above and a series of optimization runs from different initial conditions in different sectors of the design space it was determined that the problem was greatly susceptible to this weakness. To overcome this issue the optimization was run many times using different initial conditions spread over the design space. The result of each optimization was saved and the best (lowest reconfiguration cost) was chosen as the global maximum.

The implementation uses the MATLAB function `fmincon` from the Optimization Toolbox. The initial conditions were set to vary from 400 – 2000 km altitude at 400 km intervals and from 5 – 20 degrees elevation angle at 5 degree intervals. All combinations of levels of the two variables were used as initial conditions.

**Sensitivity**

A sensitivity analysis was conducted at the optimal point using central differencing to determine the gradient. A perturbation size of 10% (after some experimentation) was used for the variable variations to obtain the gradients.

\[
\nabla J = \begin{bmatrix} \partial J / \partial h \\ \partial J / \partial e \end{bmatrix} = \begin{bmatrix} -0.00005 \\ 0.0899 \end{bmatrix} \quad (3)
\]

Normalized Sensitivities:

\[
\nabla J = \nabla J \frac{x^*}{J^*} = \begin{bmatrix} -0.8569 \\ 0.7653 \end{bmatrix} \quad (4)
\]

Both variables drive the reconfiguration problem with altitude being the slightly dominant variable. The minus sign on the altitude gradient demonstrates that lesser altitude in the target constellation degrades the objective (increases cost). Increases in elevation angle also increase cost.

**Scaling**

Because one design variable (altitude) is 3 orders of magnitude larger than the other (elevation angle) and the equality constraint is \(10^{11}\) larger than the objective (calculated in billions of dollars) a scaling analysis was performed on both the design variables and the equality constraint.

To determine whether to scale altitude, the diagonal entries of the Hessian at the optimal point were calculated using the central differencing method with a step size of 10% (the same step size used in the sensitivity analysis for the same reasons). The values for the second derivatives of \(J\) with respect to the design variables were:

\[
\text{diag}(\nabla^2 J(x^*)) = \begin{bmatrix} \partial^2 J / \partial h^2 \\ \partial^2 J / \partial e^2 \end{bmatrix} = \begin{bmatrix} -8.98 \times 10^{-6} \\ -0.3597 \end{bmatrix} \quad (5)
\]

The results indicate scaling was needed for the altitude variable only. Since its entry on the diagonal of the Hessian is on the order of \(10^6\) the appropriate scaling factor is \(10^3\). However, when this scaling factor was applied in the simulation there was negligible change in the results. This suggested that the degree of the magnitude imbalance between the design variables was ineffectual within this particular design space.

The one constraint in this system is an equality constraint on the order of \(10^{11}\). The design variables \(J\) are of the order \(10^1, 10^3, \) and \(10^5\) respectively. However, since the constraint is implemented by calculating the lifetime capacity of the system and comparing it to a target capacity of the same order of magnitude near \(x^*\) there is no need to scale the capacity constraint.

**Heuristic Method - Simulated Annealing**

**Results**

The parameters chosen for the Simulated Annealing algorithm are the same as those for the SQP method above.

The SA algorithm was executed 30 times, and the best solution that was obtained was \(x^*=[1085, 5]\) and \(J^*=$587 million (Figure 8). The \(J^*\) found equals the \(J^*\) from the SQP method and \(x_{sol}^*=x_{sol}^*\).

**Technique**

Simulated Annealing was implemented in MATLAB with the SA Toolbox. An initial guess of \(x_0=[800, 8]\) was used. Since SA is heuristic and does not guarantee convergence on an optimal solution, it was executed 30 times with the same initial condition and algorithm parameters in order to obtain statistically meaningful results.

Table 5 shows the values used for some of the parameters of the SA algorithm.

<table>
<thead>
<tr>
<th>SA algorithm parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Temperature ((T_0))</td>
<td>100</td>
</tr>
<tr>
<td>Cooling schedule</td>
<td>Exponential</td>
</tr>
<tr>
<td>Temperature increment ((dT))</td>
<td>0.8</td>
</tr>
<tr>
<td># of rearrangements attempted to reach equilibrium</td>
<td>5</td>
</tr>
<tr>
<td>Frozen condition ((# of successive temperatures for which the defined # of desired acceptances is not achieved))</td>
<td>15</td>
</tr>
</tbody>
</table>

The thirty ‘optimal’ solutions were statistically analyzed to obtain mean values and standard deviations. The mean \(J^*\) was $592 million with a standard deviation of 0.56%. The mean and standard deviations of the design variables in the optimal design vectors are shown in Table 6.
Table 6: Mean and Standard Devs of $x^*$ and $J^*$ from SA

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Standard Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h^*$ (km)</td>
<td>1106</td>
<td>101.72</td>
</tr>
<tr>
<td>$\varepsilon^*$ (deg)</td>
<td>7.7</td>
<td>2.89</td>
</tr>
<tr>
<td>$J^*$ ($\text{million}$)</td>
<td>592</td>
<td>0.0056</td>
</tr>
</tbody>
</table>

Figure 8: Orbit Reconfiguration SA Convergence history

Post Optimality Analysis

Figure 9 is the contour plot of the objective function in the design space. The plot shows the reconfiguration cost of transforming the initial constellation $A$ of 2000 km and 5 deg to another constellation $B$ (given by coordinates of altitude and elevation angle) in the figure below. The constellations that correspond to a capacity of $10^{11}$ minutes are marked by circles in the figure. It can be seen that the lowest cost associated with a constellation that satisfies the capacity of $10^{11}$ minutes is indeed $587$ million and its coordinates are 5 deg and 1066 km. Note that this figure was not generated through any optimization routine. It was created by doing a full factorial computation of the design space, and then marking the points that correspond to constellations of $10^{11}$ lifetime capacity. This figure thus validates the results that were obtained through SQP and simulated annealing algorithms, and it can be seen that indeed the global optimum was found for the solution.

5. Multi-Objective Optimization Results

Formulation

Adding the objective of maximizing system capacity changes the Single Objective Optimization (SOO) into a multi-objective optimization problem. The new problem was then to find optimal new constellations while minimizing reconfiguration cost and maximizing system capacity, $C_B$ greater than or equal to $C_{desired}$.

$$\min J_{mo} = \lambda J_1 + (1-\lambda)J_2$$

$$J_1 = Cost_B$$

$$J_2 = -C_B$$

s.t.

$$400 \leq h \leq 2000 \text{ (km)}$$

$$5 \leq \varepsilon < 20 \text{ (deg)}$$

and

$$C_B \geq C_{desired}$$

Scaling

The desired capacity was again set to $10^{11}$ minutes. However because this number is now being evaluated alongside the results of $J_1$, which are on the order of $10^1$, it became necessary to scale $J_2$. Due to the variance of $J_2$ over several orders of magnitude across the design space $J_2$ was scaled by $\log_{10}$. The new objective function was therefore:

$$J_{mo} = \lambda J_1 + (1-\lambda)\cdot(\log_{10}(J_2))$$

Direction of Objectives

The first objective is to minimize reconfiguration cost while the second objective is to maximize capacity. Reconfiguration cost decreases with smaller changes in altitude and elevation angle and capacity increases with decreasing altitude and increasing elevation angle. Since our
initial constellation, \( A \), is at the maximum altitude and minimum elevation angle, our objectives are competing. A negligible change in the design vector from constellation \( A \) would produce the lowest cost but the lowest capacity. Maximum change from \( A \) would produce the highest cost but highest capacity.

**Pareto Front**
The Pareto Front was determined with the Weighted Sum approach. The weight \( \lambda \) was varied from 0 to 1 in increments of 1/20.

Although 20 points were generated, there is overlap between some of the solutions so not all 20 points are visible. The following Figure 11 and Figure 12 show the efficient designs (values of altitude and elevation) that correspond to the Pareto front.

At each extreme of \( \lambda \), constraints are active. When \( \lambda \) is 0 (optimizing capacity) the lower bound of altitude and upper bound of elevation angle are active. When \( \lambda \) is 1 (optimizing cost) the capacity constraint \((1 \times 10^{11})\) and the lower bound of elevation angle are active. The capacity constraint keeps altitude from reaching its upper bound.

**Sensitivity to Weights**
The Pareto Front does not change shape but becomes smoother as the number of divisions increases. This indicates changing the weights on the individual objective functions does not change the final result.

**Intra-Satellite Reconfiguration**
Intra-satellite reconfiguration means reconfiguring internal components in individual satellites of a constellation to deliver a modified or new functional characteristic. Characteristics of the satellite components are constant in present-day constellations. However, the purpose of this analysis is to determine the effects on system capacity if some particular hardware characteristics were changeable or reconfigurable. In this study, the satellites are assumed to have some reconfigurable components on board, for instance, a transmitter antenna with variable diameter. This section studies the problem of determining the optimal reconfigurations a set of satellite components should undergo to increase the constellation capacity to a specified higher level. The objective is to minimize the magnitude of component reconfigurations needed to deliver the higher system capacity.

**Case Study-II**
In this case study, the parameters for the problem were set to those of the Iridium communication satellite constellation. It was assumed a single-fold, global coverage
The communication constellation is deployed at an altitude, $h_{at}$, of 780 (km) with the minimum elevation angle, $\epsilon_{av}$, of 8.2°. The constellation has 66 satellites in 6 orbital planes, with 11 satellites per plane. All satellites are identical. Assuming certain values for the link design elements (given in Table 3) the communications module computed the total system lifetime capacity to be $3.1 \times 10^{11}$ minutes.

1. Problem Formulation

The optimization problem addressed in this case was formulated as:

$$\begin{align*}
\min & \quad J = \|x_A - x_B\| \\
\text{s.t.} & \quad 0.1 \leq D_a \leq 10 \quad (m) \\
& \quad 50 \leq P_t \leq 800 \quad (W) \\
& \quad 2.4 \leq R \leq 10 \quad (kbps) \\
& \quad 5 \leq T_{sat} \leq 15 \quad (years) \\
\text{and} & \quad C_B = C_{desired}
\end{align*}$$

The variables $x_A$ and $x_B$ denote the design of satellites in constellation $A$ and $B$ respectively. The system lifetime capacity of the reconfigured constellation is $C_B$. The new capacity that constellation $B$ should meet is $C_{desired}$ where $C_{desired} > C_A$. The bounds on the design variables were chosen after experimenting with the optimization routine and finding physically meaningless designs. To produce only meaningful results, the bounds were applied.

Note, the only difference between constellations $A$ and $B$ is the satellite components in $B$ are configured to have different attributes than $A$. Orbital parameters, number of satellites etc. are the same.

The objective function $J$ was chosen to minimize the change between the component characteristics in $A$ and $B$. Therefore the difference between $x_A$ and $x_B$ should be as small as possible given the constraints.

The target capacity, $C_{desired}$, was set to $10^{12}$ minutes.

2. Design of Experiments (DOE) Analysis

It was not clear which variables should be used in the design vector for this problem. Among the several parameters affecting system capacity, some of the key variables with the greatest affects were identified. A full factorial DOE analysis was therefore conducted to determine the important factors. Each factor was analyzed at different levels, and the effects of each level along with the over all effects for each factor were computed. For instance, the effect of the satellite lifetime, $T_{sat}$, was evaluated at three levels of five, ten and fifteen years. Figure 13 shows the results.

It can be seen that the antenna diameter (m), $D_a$, transmit power (W), $P_t$, single user data rate (kbps), $R$, and satellite lifetime (years), $T_{sat}$, affected the capacity in reasonable amounts and are suitable candidates for the design vector. Some other variables (for instance the receive gain, $G_r$, shown in the figure) had negligible effects and were not considered for variables in the design vector.

From the results of the DOE analysis the final design vector to study the intra-satellite reconfiguration problem was $x^T = [D_a, P_t, R, T_{sat}]$.

3. Simulation Model and Benchmark

Figure 14 shows the fairly simple simulation framework structure that was employed for this case. It consisted of the constellation module and the communication module (the details of both were given earlier).

4. Single Objective Optimization

As with the Orbit Reconfiguration problem, SQP and Simulated Annealing were used to solve this optimization problem. The reasons for choosing these methods were the same as those described in the previous sections. The design
vector of constellation $A$ was set to equal Iridium’s characteristics and was $x_o = [1.5, 400, 4.8, 5]$.  

**Gradient Method - SQP**

**Results**
The optimal design $x^*$ was determined to be $[2.5, 400, 4.52, 5.24]$, with the optimal objective $(J)$ of 1.07. This solution indicates the satellites need to have their antenna diameters extended by one meter (to change from the initial size of 1.5 to 2.5), data rate per user reduced to 4.52 kbps from 4.8 kbps, and lifetime increased by 0.24 years (three months).

**Technique**
The SQP algorithm was implemented using MATLAB’s `fmincon` function. In most cases (except for some particular initial conditions) the algorithm converged easily. The solution was obtained in eight out of twelve optimization runs, each of which was executed with a different initial condition. The remaining four runs had objective function values higher than 1.07. It could therefore be assumed with fair amount of confidence that the solution was not from some local minimum.

**Sensitivity**
The gradient at $x^*$ was calculated with the central differencing method.

$$\nabla J = \begin{bmatrix} \frac{\partial J}{\partial D_a} \\ \frac{\partial J}{\partial D_T} \\ \frac{\partial J}{\partial R} \\ \frac{\partial J}{\partial T_{au}} \end{bmatrix} = \begin{bmatrix} 0.94 \\ 0 \\ -0.26 \\ 0.22 \end{bmatrix} \quad (9)$$

Normalized Sensitivities:

$$\overline{\nabla J} = \nabla J \cdot \frac{x^*}{J^*} = \begin{bmatrix} 2.19 \\ 0 \\ -1.09 \\ 1.09 \end{bmatrix} \quad (10)$$

The most important driver appears to be the transmit antenna diameter followed by the satellite life time. The data rate per user also affects the optimal point but in a negative way, meaning that reducing the data rate improves the objective. This trend in fact was observed in the optimal solution in which the antenna diameter and lifetime were increased, while data rate per user was reduced.

**Parameter Sensitivity**
The constellation altitude and elevation angle were treated as parameters in this case, and were set to 780km, and 8.2° respectively (to correspond with the Iridium constellation values). A sensitivity analysis demonstrated the effects of these two parameters. Altitude sensitivity was found to be 0.3% while elevation angle sensitivity was 45.5%. The elevation angle has a significant influence on the objective function.

$$\frac{\partial J}{\partial h} = 0.003 \quad (11)$$

$$\frac{\partial J}{\partial \epsilon} = 0.455$$

**Scaling**
Since the values of the $P_t$ variable in the design vector were one order of magnitude apart from that of the other design variables, $P_t$ was scaled by a factor of 10.

**Heuristic Method - Simulated Annealing**

**Results**
Out of the thirty iterations, the best $J^*$ was 1.48 and corresponding $x^*$ was $[2.5, 400.9, 4.4, 5]$ (Figure 15). This value of $J^*$ is close to the optimal value found by SQP (1.07).

The optimal solution is such that there isn’t any large change in any one variable (the antenna diameter is increased by 1 meter, negligible change in transmit power, a reduction of ~0.4 kbps in data rate, and no change in the satellite lifetime). This is a reasonable solution for reconfiguring satellites components to meet higher capacity demand, and is very similar to the SQP method solution.

**Technique**
Simulated Annealing was also implemented in MATLAB with the SA Toolbox.

An initial guess of $x_o = [1.8, 450, 4.5, 5]$ was used. Since there were four design variables, two variables at a time were perturbed because two degrees of freedom (DOF) had been determined to yield the best behavior [6]. The candidate design vectors were perturbed to produce feasible
designs before being evaluated for their “energy”. Table 7 presents values used for some algorithm parameters.

Table 7: Satellite Reconfiguration SA Parameters

<table>
<thead>
<tr>
<th>SA algorithm parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Temperature (T0)</td>
<td>5000</td>
</tr>
<tr>
<td>Cooling schedule</td>
<td>Exponential</td>
</tr>
<tr>
<td>Temperature increment (dT)</td>
<td>0.8</td>
</tr>
<tr>
<td># of rearrangements attempted to reach equilibrium</td>
<td>5</td>
</tr>
<tr>
<td>Frozen condition (# of successive temperatures for which the defined # of desired acceptances is not achieved)</td>
<td>8</td>
</tr>
</tbody>
</table>

The SA algorithm was executed thirty times (with the same initial condition and parameter settings). The thirty ‘optimal’ solutions were statistically analyzed to obtain mean values and standard deviations. The mean $J^*$ was 5.26 with a standard deviation of 2.42. The mean and standard deviations of the design variables in the optimal design vectors are shown in Table 8.

Table 8: Mean and Standard Devs of $x^*$ and $J^*$ from SA

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Standard Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D_a$ (m)</td>
<td>2.21</td>
<td>0.59</td>
</tr>
<tr>
<td>$P_t$ (W)</td>
<td>400.28</td>
<td>3.39</td>
</tr>
<tr>
<td>$R$ (kbps)</td>
<td>5.15</td>
<td>1.99</td>
</tr>
<tr>
<td>$T_{sat}$ (yr)</td>
<td>8.16</td>
<td>2.56</td>
</tr>
<tr>
<td>$J^*$</td>
<td>5.26</td>
<td>2.42</td>
</tr>
</tbody>
</table>

5. Multi-Objective Optimization Results

Formulation

Adding the objective of maximizing system capacity changed the SOO into a multi-objective optimization problem. The new problem is to determine optimal reconfigurations minimizing the difference between initial and final component configurations and maximizing system capacity greater than or equal to $C_{desired} = 10^{12}$ min. Formally:

$$\min J_{wo} = \lambda J_1 - (1 - \lambda) J_2$$

$$J_1 = |x_A - x_B|$$

$$J_2 = C_B$$

s.t.

$$0.5 \leq D_a \leq 10 \text{ (m)}$$

$$200 \leq P_t \leq 1200 \text{ (W)}$$

$$4 \leq R \leq 10 \text{ (kbps)}$$

$$5 \leq T_{sat} \leq 15 \text{ (years)}$$

and

$$C_B \geq C_{desired} = 10^{12}$$

Scaling

In the actual implementation, the transmit power, $P_t$, was scaled by a factor of 10 (its values were an order of magnitude different from all the other design variables). Furthermore, $\log_{10}$ of the capacity was used for $J_2$ so that $J_1$ and $J_2$ would be comparable in magnitudes.

Direction of Objectives

The objectives are competing because the first objective, $\min J_1 = ||x_A - x_B||$, minimizes the reconfiguration of the satellite components, while the second objective, $\max J_2 = C_B$, is achieved by changing the component configurations as much as possible.

No additional satellites may be added to the constellation, nor can the orbital parameters of the constellation be changed. The only way to increase capacity is to change the values of the satellite component characteristics.

The first objective keeps the reconfiguration small, whereas the second objective makes it large. System lifetime capacity in relation to the design variables is:

$$C = \frac{f(P_t, D_a, T_{sat})}{R}$$

where $f$ is a nonlinear function that directly relates $P_t, D_a,$ and $T_{sat}$ to $C$ (i.e. increasing these variables increases $C$). Equation 13 also shows $R$ is inversely proportional to $C$. The second objective will therefore try to maximize $P_t, D_a,$ and $T_{sat}$ and minimize $R$ as much as possible subject to the constraints. The first objective however, will try to minimize these changes. The two objectives are therefore opposing.

Figure 16: Satellite Reconfiguration Pareto Front
Pareto Front
The weighted sum (WS) approach was used to perform the multi-objective optimization and generate the Pareto Front (Figure 16). The weight \( \lambda \) was varied from 0 to 1. The negative sign (instead of the usual positive sign) was used since \( J_2 \) was being maximized.

Figure 17 shows the efficient solutions corresponding to the Pareto front. When \( \lambda \) is small, some of the variables are set to their limiting values to get as much capacity as possible. \( D_a, P_t, \) and \( T_{\text{sat}} \) are set to their maximum allowable value (up to their upper bound limits), while \( R \) is reduced as much as possible (up to its lower bound limit).

Increasing \( \lambda \) decreases the design variables from \( x_a \) values. As discussed earlier, the initial component configuration values were set to \( x_A = [1.5, 400, 4.8, 5]^T \). The plots show \( D_a, P_t, R \) and \( T_{\text{sat}} \) approach these values with increasing \( \lambda \). This is because when \( \lambda \) is zero, emphasis is on maximizing capacity (\( J_2 \)); and as \( \lambda \) increases emphasis shifts to minimization of \( \| x_A - x_B \| \) (\( J_1 \)). This is because when \( \lambda \) is zero, the optimization is effectively for only \( J_2 \).

Conclusions and Future Work
This study shows the capacity of communication satellite systems can be increased through orbital or internal component reconfigurations. The ability to reconfigure for higher capacity can add value to such systems and allow for increased economic benefit.

In the analysis for intra-satellite reconfiguration, the single user data rate, \( R \), was treated as a variable. For fixed quality of service (QOS), \( R \) cannot be modified and should thus be treated as a parameter rather than a design variable in future studies. Furthermore, the objective function used in the second case study was the norm of the difference of two vectors \( x_A \) and \( x_B \) that described characteristics of certain communication subsystem components. A more realistic objective function may be one in which the change in individual components is penalized differently. In this method, components that can be assumed to be more difficult to realistically ‘reconfigure’ will have higher penalties. For instance, \( J \) can be:

\[
J = \alpha(Da_A - Da_B) + \beta(Pt_A - Pt_B) + \gamma(R_A - R_B) + \eta(T_{\text{sat}A} - T_{\text{sat}B})
\]

With this objective function, values of \( \alpha, \beta, \gamma, \eta \) may be altered to more realistically capture the ‘cost’ of having changeable satellite life time, antenna diameter etc.

References


