AE 6310 Design Project: Optimization of an Earth Satellite Constellation
Spring 2017

Overall Instructions

In this team project, you will perform an optimization analysis for the design of an Earth orbiting satellite constellation. The algorithms must be written in Matlab by your team; use of existing optimization functions within Matlab or written by others is not allowed. You will need to distribute the different tasks amongst the members of your team to share the workload fairly and effectively. Teams will be assigned as a result of the CATME surveys, or in the case of non-response to the survey, randomly.

To the extent possible, teams are paired as either main campus (A-section) teams or distance learning (Q-section) teams. For A-section teams, the project report is due in class on Tuesday April 25, 2017 at 12:10 PM. The assignment must be turned in as a printed report. Page length is not specified but a typical report is expected to be 20-30 pages. Additionally, the computer source codes that you write as part of the project must be uploaded by the same time to the T-Square "drop box" folder of at least one team member. Specify the name of this team member on the top of the printed report that you turn in during class. You do not need to print hardcopies of your Matlab code since you are turning them in on T-Square.

Q-section teams have a 1 week delay in the deadline, so their project reports are due on Tuesday May 2 at 12:10 PM ET. Q-section project reports must be emailed by one student on the team to the instructor by this time as a pdf file and Matlab code must be uploaded by the same time to the T-Square "drop box" folder of at least one team member. Specify the name of this team member on the top of the pdf report that your team turns in.

Problem Statement

Introduction

Your project is to design an optimal constellation of Earth orbiting satellites to provide a commercial service to customers whose value depends on the constellation's total coverage of the Earth, such as communications or imaging. For inspiration, look up actual companies such as Terra Bella, Planet Labs, Iridium, and Spire.

The satellites cost money to produce, launch into orbit and operate for a period of 5 years. Your goal is to maximize overall Earth surface coverage of the constellation while minimizing total costs over the 5 year period of performance. Both of these performance criteria are considered important by the project sponsor and so a multi-objective optimization approach is needed to adequately characterize the design performance trade space.

Satellite Coverage

Satellite motion is modeled to follow simple gravitational orbital mechanics for a circular orbit. We will not consider higher order effects (e.g. drag, J2, etc.) in this analysis. The orbit period, P, is a function of its altitude, h:

$$P(h) = \frac{2\pi}{\sqrt{\mu_E}}\,(r_E + h)^{3/2}$$

with r_E = 6378 km and $\mu_E = 3.986\text{x}10^5\ \text{km}^3/\text{s}^2$. The Earth rotates at a constant rate of ω_E = 15 degrees per hour with respect to a (quasi) inertial reference frame. The orbit plane's orientation with respect to the Earth is specified using parameters known as inclination, *i*, and longitude of the ascending node, Ω given in degrees or radians. For more information about satellite orbits consult a text such as *Fundamentals of Astrodynamics*, by Bate, Mueller, White, Chapter 2. In order to simulate each satellite's motion write a Matlab function that propagates a satellite's position in time relative to the Earth in its orbit based on its altitude *h*, inclination *i*, longitude of the ascending node Ω and an initial condition specifying its position in the orbit at time *t* = 0 (e.g. mean anomaly, *M*). Note that the satellites are restricted to circular orbits- this fact simplifies the orbit propagation equation considerably.

Individual satellite Earth coverage (*ISC*) is based on a simple geometric consideration of the satellite's line of sight visibility to any portion of the Earth based on the satellite's position in its orbit. We will assume the Earth is a perfect sphere of radius r_E = 6378 km, and we will restrict our satellite orbits to be circular at a given altitude *h* and inclination *i*. See for example "Visible or Illuminated Fraction of Total Surface Area of a Sphere", http://www.neoprogrammics.com/spheres/visible_fraction_of_surface.php. No consideration is given to whether the Earth surface is illuminated by the Sun (day) or in shadow (night). You will need to write a Matlab function that calculates the Earth surface region that is visible to a given satellite at a given point in time based upon its position. It is acceptable to bin the result into 1 degree increments of latitude and longitude for analytical tractability if needed.

Total constellation coverage (*TCC*) is the union of all the individual satellite coverages in the constellation at any time. No credit is given for overlapping regions which are covered by more than one satellite at the same time. Total constellation coverage at any time is described as a fraction of the total Earth surface, $0 \leq TCC \leq 1$. Total constellation coverage will vary over time as the individual satellites move in their orbits. You will need to write a Matlab function that takes the output of the previous function (individual satellite coverage) for each satellite in the constellation at a specified time and determines the *TCC*.

Overall constellation coverage (*OCC*) may be described using statistical metrics of total constellation coverage *TCC(t)* such as mean value (average), standard deviation, maximum and minimum. The coverage goal is to maximize *OCC = E*{*TCC*} = *TCC* mean value while minimizing Total Cost. In addition to maximizing *OCC*, a smaller standard deviation for *TCC* (σ_{TCC}) is also desired as a secondary consideration in differentiating between different design solutions.

Missed constellation coverage (*MCC*) may be defined as *MCC* = 1 – *OCC*. Minimizing *MCC* has the same effect as maximizing *OCC*.

Pay attention to your simulation sample rate and duration. Satellite positions will complete a full revolution in one orbit period, so you will need to simulate the constellation geometry frequently enough to accurately measure the coverage performance. Once per minute is suggested as a starting point and you can experiment with other values. On the other hand, it isn't necessary to simulate a full 5 years of orbit positions if you can determine that the constellation geometry is itself periodic given the orbits that your satellites occupy.

Although the sponsor does not give specific information about coverage goals, they do emphasize that an overall constellation coverage mean value $OCC = E\{TCC\} \geq 0.7$ (70% of the Earth surface) is required for the project to be viable. Therefore it is not worth recommending any designs which provide less than this amount of *OCC*.

Cost
Cost is normalized in millions of dollars. Your constellation design is specified by 1) the number of satellites in the constellation (specified by *n*), 2) the orbit planes the satellites are placed into (specified by *h, i,* and Ω), and 3) the position of each satellite within its orbit plane (specified by an orbit parameter such as mean anomaly, *M*).

The satellite production costs are given by an exponential production cost model with a learning curve for increased production efficiency with quantity. The unit cost of each satellite varies from $10 for quantity *n* = 1 to $5 for an infinite number according to an exponential rate decay. The cost decay rate is given by a scale factor of 25 satellites for this problem. The unit cost (*UC*) of each satellite may be determined by plugging the total number of satellites, *n*, in the constellation design into the formula.

$$UC(n) = \$5(1 + e^{-n/25})$$

The production cost of the constellation is the unit cost of each satellite times the total number of satellites.

$$PC(n) = UC * n = \$5n(1 + e^{-n/25})$$

Additionally, each satellite costs a fixed amount to operate. The expected operating cost is $1 per satellite per year for a total of $5 per satellite over the 5 year operational period of performance. The operations cost covers the cost to command and control the satellite, and collect and distribute data to the customers. There is no assumed economy of scale for operating a larger number of satellites, so:

$$OC(n) = \$5n$$

The orbits the satellites are placed into is limited to a finite set of launch options given by the launch vehicle provider. For the purposes of this study, only two launch variables are considered: orbit altitude, *h,* and inclination angle, *i*. Satellites may also be launched into different orbit planes (specified by longitude of the ascending node, Ω) by varying the launch time. All final orbits are assumed to be circular.

Some orbits are more costly to launch into than others. The launch costs are specified in the provided launch cost table, Table 1. Launch costs are assumed to be fixed in this problem and there is no economy of scale for obtaining multiple launches.

Table 1. Launch Cost (*LC*) Per Launch ($M) as a function of orbit altitude and inclination.

		Inclination (*i*), deg		
		0	*45*	*90*
Altitude (*h*), km	*3290*	$50	$40	$60
	1000	$30	$20	$40

All target orbits are circular. The launch vehicle is capable of placing up to 5 satellites into a single orbital plane per launch at the specified altitude *h* and inclination *i*. Longitude of ascending node Ω may be varied by selecting the launch time. (For equatorial orbits, another orbit parameter such as true longitude at epoch, l_0 is used to describe the satellite's position at launch time.) The individual satellites may be placed anywhere within the target orbit at no additional cost which is specified by selecting different values of initial mean anomaly *M* for each satellite. However, once satellites are placed into their orbits, they may no longer be repositioned other than by standard orbital motion. There is no reduction in launch cost for launching less than 5 satellites at a time. You do not have to consider how long it takes to arrange for a sequence of launches to build your constellation, but you do have to include the cost of each launch in the total cost of the constellation.

Total Cost (*TC*) is the sum of constellation launch costs (*LC)*, production costs (*PC*), and operation costs (*OC*): *TC = LC + PC + OC.*

Sponsor Directions
The goal is to maximize Total Constellation Coverage mean value, *OCC = E*{*TCC*}, while minimizing Total Cost, *TC*, over the 5 year period of performance. Both of these performance criteria are considered important by the project sponsor and so a multi-objective optimization approach is needed to adequately characterize the design performance trade space. Constellation performance is determined only after the satellite constellation has been fully assembled and is operational.

Project Assignment

Task 1: Problem Definition (22 points)
Using all the information provided, determine (1) a suitable set of independent design variables that adequately define the design search space, and (2) a formulation of the objective functions based on the optimization criteria and algorithms that will be used. You should strive to formulate the problem with as few design variables and constraints as possible in order to obtain an efficient solution. After you have studied the problem, write down your optimization problem statement in the following form:

Min/max {*objective1, objective2*}
By changing: {*design variables*}
Subject to: {*any constraints*}

Function [*objective1, objective2*] = analysisFunction(*design variables*)

Your analysis function should make function calls to any other intermediate functions that you develop that are required to evaluate the objectives as described in the Problem Statement.

Note that there are many ways to set up the problem, but the assessment for this task will be based on the clarity and elegance of your problem formulation. Your problem statement will also directly affect your solution methodology and results obtained, so it is worth spending time to make this part as good as possible.

<u>*Deliverables for this Task:*</u>

1. Your optimization problem statement in the form described above.
2. A written paragraph discussing why you chose to formulate the problem in this way and any equations or constraints that you introduced.
3. A complete set of your Matlab code for your analysis function named analysisFunction.m with your teammates' names written in the comments section of your code. Turn in your code via the DropBox folder on T-Square (you do not have to include your code in your printed report). Also include in your Matlab code any intermediate functions which you have written as outlined in the Problem Statement. Write your teammates' names in the comments section of each function you turn in.

Task 2: Optimization Algorithm Development (22 points)

Code *your own* genetic algorithm as a Matlab function using any methods you know or have learned in class. You must incorporate some form of elitism but the remaining selection, crossover, and mutation mechanisms are left to your discretion. Select a crossover rate and mutation rate that give good "on average" performance of your algorithm for this problem. This may take some experimentation and trial and error on your part prior to reporting your results. The choice of how to handle any constraints is also left to you.

Test your GA by minimizing the egg crate function:

$$f(\boldsymbol{x}) = x_1^2 + x_2^2 + 25(\sin^2 x_1 + \sin^2 x_2)$$

with the constraint:

$$x_2 + 2x_1 - 1 \geq 0$$

by running the algorithm for 50 generations with a population of 50 individuals. Use bounds on the design space of [-5, 5] in each dimension and a discretization resolution of the design variables of 0.01.

Note: it is possible to solve this problem with various open source codes and Matlab optimization toolboxes, but that is not the point. You are asked to write your own GA code based on what you have learned in class and to present its results. You may use external codes to check your results, if desired.

<u>*Deliverables for this Task:*</u>

1. A series of five plots that show the spread of the population at every 10th generation. Use scatter plots overlaid on contour maps of the objective function with x_1, x_2 as axes. Represent the infeasible region with a shaded area.
2. A table that lists the optimal values of x_1 and x_2, the minimum value of the function, the total number of function calls used, and the crossover rate and bit mutation rate that you selected.
3. A written paragraph description interpreting your results. Include a concise description of your constraint handling method.
4. A complete listing of your GA code named GA.m with your names written in the comments of your code. Turn in your code via the DropBox folder on T-Square (you do not have to include your code in your printed report). Also include any intermediate functions which you have written to complete this task. Write your teammates' names in the comments section of each function you turn in.

Task 3: Solve the Single-Objective Constellation Design Problem (22 points)

Demonstrate your solution to the Constellation Design Problem by applying a modified version of the genetic algorithm you developed in Task 2 to the problem statement that you formulated in Task 1. Present a single objective design solution where the Overall Constellation Coverage is at least 0.7 (i.e., $OCC \geq 0.7$) and your single objective goal is to minimize Total Cost, *TC*. If you encounter any problems applying the algorithm to solve the problem, investigate, implement, and describe any modifications that you made to the algorithm that were necessary to make it work for this problem.

<u>*Deliverables for this Task:*</u>

1. A table listing the optimal values your algorithm found for the design variables you selected in your problem statement listed in Task 1 and the resulting values of OCC and *TC*.

2. A written paragraph discussing any challenges you encountered with implementing the algorithm and your corresponding modifications to obtain a solution.
3. A written paragraph discussing the relative importance of different design variables affecting the solution and any interesting results regarding the optimum design you obtained.
4. A complete listing of your modified GA code named ConstellationGA.m with your names written in the comments of your code. Turn in your code via the DropBox folder on T-Square (you do not have to include your code in your printed report). Also include any intermediate functions which you have written to complete this task. Write your teammates' names in the comments section of each function you turn in.

Task 4: Perform the Multi-Objective Design Analysis (22 points)
Now that you have completed a working single-objective design, analyze the multi-objective design problem as posed by the sponsor. Consider several cases of multi-objective optimization with Overall Constellation Coverage (OCC) and Total Cost (*TC*) as separate objectives. Use any method discussed in class (aggregate objective function, epsilon constraint method, NBI, NSGA) to map the Pareto Frontier. Create a plot that shows the cases you simulated and the resulting Pareto frontier boundary. Note, it will be better to present $MCC = 1 - OCC$ as your coverage objective function value as this will give the desired shape to your Pareto frontier plot. We won't offer specific guidance on how many designs to simulate other than it should be enough to adequately visualize the Pareto frontier.

Deliverables for this Task:

1. A written paragraph description of the method you used to conduct the multi-objective analysis and determine the resulting Pareto frontier.
2. A scatter plot showing your multi-objective optimization results with each design solution shown as a point, the feasible space identified, and the resulting Pareto frontier you obtained clearly labeled. You may either use a different point symbol or connect points on the Pareto frontier to show it more clearly on the plot.
3. Provide a table listing the Pareto dominant values of the design variables you selected in your problem statement listed in Task 1 and the resulting values of *OCC* (or *MCC*), and *TC*. If you have more than 10 design points shown on your Pareto frontier, list only 10 representative designs in your table.
4. A written paragraph describing any trends that are evident in the Pareto frontier both in the objective functions and in the design variables and any noteworthy solutions.

Project Report Grading Rubric

Each of the four tasks is weighted equally (22 points). In addition to basic satisfaction of the deliverables, each Task will be checked for 1) Correctness of design approach, 2) Quality of results obtained, 3) Quality of any visualizations presented, 4) Quality of any requested results discussion and analysis, and 5) Completeness and quality of any code provided (is code organized, easy to understand and well commented?).

An additional 8 points are given for 1) overall report organization and polish, and 2) demonstrated ability to convey important information concisely yet thoroughly.

After the project is completed, individual team members will be asked to complete an anonymous survey via CATME to provide a peer assessment of their teammates' performance on the project. Up to 4 points will be individually given to team member scores based on the peer assessment results.