Modeling and Optimization of a Cubesat Platform

David P. Malawey[1] and Hung Nguyen.[2]
Texas A&M University, College Station, TX, 7840

A low cost and low fidelity model of a small satellite (cubesat) was developed in this project. The model contained basic modules of a typical cube-sat project: structure, propulsion, power, mass and cost. Using Multidisciplinary System Design Optimization techniques on 6 bounded design variables and various inequality constrains, the mass and cost of the design were optimized. An L_{18} orthogonal array was used to perform design space exploration. MATLAB was utilized to carry both heuristic and gradient-based methods in optimizing the problem. A generic algorithm was used to solve for the configurations of the discrete variables then sequential quadratic programming was used to find the global optimum for each mass and cost. For dual objective optimization, Pareto front points were found using adjusted weighted sum. Post optimality, sensitivity analysis was performed and it concluded that the number of batteries has the highest impact on the cost of the design while the structure width is the design variable that has the highest impact on the mass. Despite of its low fidelity, the model is still very useful. It could be used to rapidly explore the design space and give preliminary setup for future development of high fidelity model.

Nomenclature

X	=	design vector	x^*	=	optimal design vector
$x0$	=	initial design vector	λ	=	weighting factor
J	=	objective value	I	=	moment of inertia
g	=	nonlinear constraint			

I. Introduction

A cubesat is a standardized type of small satellite with a form factor developed at Cal Poly in 1999 along with a document called the Cubesat Design Specification [1]. To date, hundreds of cubesats have flown in Low Earth Orbits, usually with missions that terminate after a couple of months as they combust in the atmosphere. Researchers and students have designed their own experiments to fly inside of cubesats but all cubesats have some common requirements including a sound frame, power sources, communications modules, and possibly attitude control. Figure 1**Error! Reference source not found.** shows a typical cubesat. Designs are continually advancing in the industry but there is still room for design improvements and cost reduction.

[1] Master's Student, Mechanical Engineering, College Station, TX.
[2] Master's Student, Mechanical Engineering, College Station, TX.

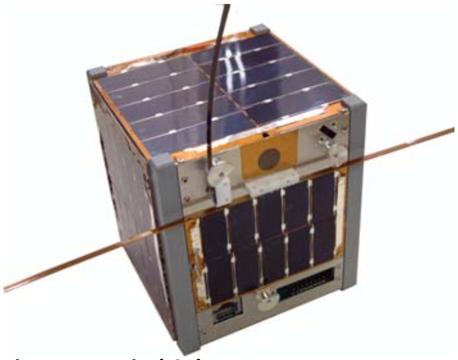


Figure 1. Typical Cubesat

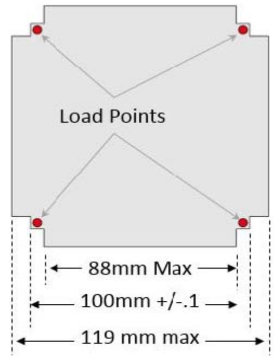


Figure 2. Design Space Dimensions (end view)

II. Problem Formulation

A. Goal
This optimization study aims to make a straightforward and flexible tool to optimize components of a cubesat platform. If the optimizer is successful it should output design information that can be applied in a real prototype to minimize mass and cost of a satellite that performs all required operations. To operate, the satellite needs to have electric power, structural soundness, and self-propulsion to control flight attitude.

B. Design Vector
The design vector is described in **Error! Reference source not found.**. The first three variables are strictly discrete and have a table of properties associated with each selection. Solar panel quantity and battery quantity are discrete but can be treated as continuous in most of the project. Structure rail width refers to the cubesat frame and is continuous.

Table 1. Design Variables

Variable	Description	Metric	Lower	Upper
X(1)	Propellant type	Gas type	1	9
X(2)	Thruster type	Model	1	3
X(3)	Structure material	Material	1	3
X(4)	Solar panels	Quantity	0	4
X(5)	Batteries	Quantity	1	Inf.
X(6)	Structure rail width	(mm)	3	20

The bounds (given by the last two columns) are driven by available data for the first three variables. More materials, propellants, and thruster models can be populated into the problem if data becomes available. The number of solar panels is limited to a maximum of four by space available on the sides of the satellite, although it is not impossible to have more. There must be at least one battery for power storage when sunlight is not available to the solar panels. Lastly, the structure rail width has a minimum of 3mm so it is practical to fit fasteners into the rail.

C. Objective

The optimization problem has the following form:

min $\mathbf{J(x,p)} = [\text{cost}(x,p)]$
s.t. $\mathbf{g(x,p)} \leq 0$

$$x_{LB,i} \leq x_i \leq x_{UB,i}$$

D. Constraints

The satellite must have enough electric power, structural rigidity, and propellant to complete the mission described by the system parameters. The constraints are listed in Table 2. The constraints listed as hard-coded mean that the constraint is resolved inside a module. For example, the propellant module takes in the properties of the propellant specified by the design vector and generates a value for how much propellant mass is needed. Since this is a deterministic solution we choose not create a "propellant mass" design variable and waste computation time on designs which may not meet the constraint. Power generated will be a combination of batteries and solar panels, while bending rigidity is a function of material and structure geometry.

Table 2. System Constraints

Effect of Constraint	Type	Bound
Power generated is sufficient	Inequality	> Power consumed
Bending rigidity is sufficient	Inequality	> baseline rigidity
Propellant amount is sufficient	(hard coded)	= mass required to counteract flight disturbances
Prop tanks qty is sufficient	(hard coded)	= enough to hold propellant volume

III. Methodology

A. N^2 Diagram

Figure 3 is a simplified representation of our N^2 diagram. Feedback has been reduced to only one pair of modules where a fixed-point iteration method is used to make the prop module and battery module agree on propellant and power amounts. The mass and cost data is actually computed in one module but for single-objective optimization, only mass is used.

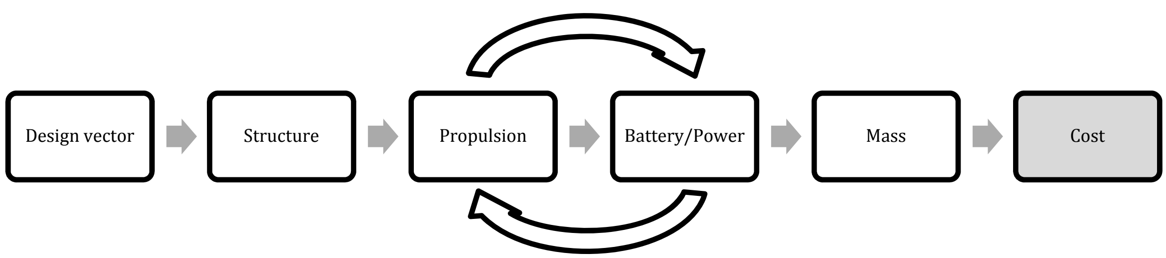


Figure 3 - Module Sequence

B. Structure Module

The structure module revolves around the X(3) and X(5) design vectors, structure material and rail width. The most important part of the structure that can be modeled simply is the rail, which gives rigidity to the satellite. Figure 4 shows the four rails of interest.

The constraint associated with this module is that the stiffness of the final design must be equal to the baseline stiffness (given by modulus *moment of inertia of the rail cross section). The cross-section is represented in Figure 5. This baseline value comes from existing satellites that have flown successfully. The moment of inertia is equal for x and y directions, and comes from Eq.(1).

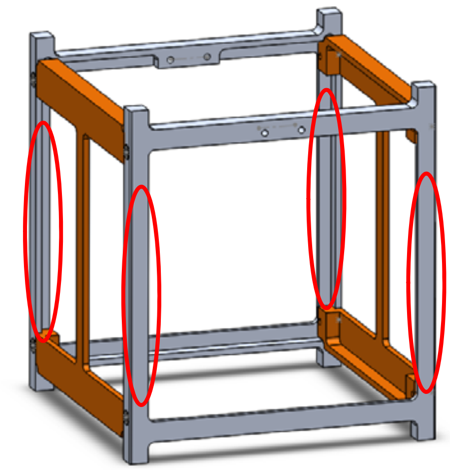


Figure 4. Cubesat Frame

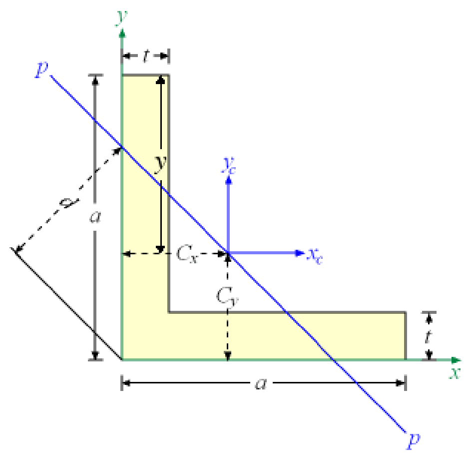


Figure 5. Rail Cross-section

$$I_x = I_y = \frac{a^2 + at - t^2}{2(2a - t)} \quad \text{Eq.(1)}$$

In the structure module, the rails cross section are assumed to be a square L beam with thickness fixed as a parameter at 2mm. So, if a flexible material is selected in X(3), the rail width "a" will need to increase to meet the rigidity constraint. Finally, the cross sectional area is calculated and combined with the material density to give a new structure mass. The mass is then used by the following modules.

C. Battery/Power Module

The battery module uses design variables X(4) and X(5), quantities of batteries and solar panels, to calculate the electric energy available to the cubesat and compare it with the requirement. Batteries of course have a fixed capacity at full charge and solar panels will offer more energy for each day of the mission. For mass the solar panels a strong advantage as a method of supplying energy but they are far more expensive.

In the module, the power consumption and duty cycles of multiple components are added to find the required energy. Parameters such as percentage of orbit spent in sunlight are estimated and can be adjusted to improve real-world accuracy. The two design variables are discrete, with zero as a lower bound for panels and 1 as a lower bound for batteries. Lastly, the electric system mass is added up and sent into the propulsion module.

D. Propulsion Module

For a successful mission the satellite needs to be oriented properly in space. Primarily the attitude must be correct to aim the antennas and ensure communication back to earth. The propulsion system must offer enough kinetic energy to keep the satellite oriented for the duration of the mission with some disturbances introduced each day. For cubesats, "delta-v" is the metric to describe how much kinetic energy can be imparted by its Attitude Control System. In reality the cubesat will not change velocity, only orientation.

Our required delta-V is a guess based on benchmarking similar systems and it increases if the length of the mission is increased. If the mass of the satellite increases then more propellant is required to achieve the same delta-V. This mass is calculated in the module.

Table 3. Propellant Properties

Propellant	Density (3500 psia, 0C) (g/cm^3)	Specific Impulse (s)	Cost($/kg)
Hydrogen	0.02	296	120
Helium	0.04	179	52
Neon	0.19	82	330
Nitrogen	0.28	80	4
Argon	0.44	57	5
Krypton	1.08	39	330
Xenon	2.74	31	1200
Freon 14	0.96	55	10
Methane	0.19	114	10
Ammonia	0.88	105	10

The propulsion module uses design variables X(1) and X(2), propellant type and thruster type. For propellant type, the interactions involved are the density of the propellant and the specific impulse of the propellant. The propellants used are listed in Table 3 with data that comes from [2]. With the propellant tank volume fixed as a parameter there is a tradeoff between a dense propellant which weighs more and a light propellant which may require multiple tanks (adding more mass for each tank). For thruster type, the general tradeoff is that a large heavy thruster consumes less battery power in operation while a compact light thruster may require higher power.

Figure 6. Thruster by The Lee Co. weighing 16.5g

Since the propellant power is fed back to the battery module and the battery mass is used in the propellant module, a fixed point iteration is used to make these two modules converge.

E. Cost (& Mass) Module

The cost module is last in line because it takes parameters from all modules and adds up the satellite price. Material, thrusters, propellant, batteries, and solar panels all have associated prices. Both raw material and machining costs are included. It's interesting to note that the cost of machining a hard material has a larger impact than buying an expensive raw material. Machining cost is calculated from a baseline estimate from a machine shop at $960 and the "machinability" factor of the material which was retrieved from [3].

Propellant cost is given by the mass of the propellant and the price per kg of propellant taken from [4]. Other component prices were found online from suppliers and manufacturers.

IV. Single Objective Optimization

F. Design Space Exploration

Our experiment plan started by assuming nonlinearity in the factor effects. Therefore, we decided to implement a design of experiments with three levels per factor. By trial and error and knowledge obtained from previous research, we defined rough feasibility ranges for each design variable. Table 4 shows the design factors and levels chosen.

Table 4. DOE factors and levels

Factor	Level 1	Level 2	Level 3
Propellant	Hydrogen (1)	Nitrogen (4)	Xenon (7)
Thruster type	moog	marotta	lee LHDB0542115H
Structure material	Al 6061 T6	Titanium	A36 steel
#panels	0	1	4
#batteries	1	2	3
Structure rail width (m)	0.003	0.006	0.008

Since we have 6 factors with 3 levels each, doing a full factorial design would take $3^6 = 729$ experiments. At the time, the team was only interested in the main effects and finding a good initial point to start, so a fractional factorial design was selected instead. We decided to use a L_{18} orthogonal array to explore the design space. This would help save computational resources while still giving us some ideas about the design space. The table describing the orthogonal array designs and resulting objective values are in the Appendix.

We evaluated the main effect each factor level on cost and mass separately and then find the level with best effect on both and mass. We want to minimize both cost and mass so we find the factor with the least mean effect. This study is also attached in the appendix. The best of cost was color-coded with yellow while the best of mass was color-coded with green.

We found that levels with the best main effect for cost and mass are different. Therefore, for sure we would need to perform multiple objective optimizations as we did in the next chapter. From the DOE analysis, we also learn the best configuration that we can use as our starting point for further analysis:

$$x_{0\_mass} = [Nitrogen,\ Moog,\ Titanium,\ 0\ Panel,\ 1\ Battery,\ 0.003\ m]$$

$$x_{0\_cost} = [Hydrogen,\ Moog,\ A36\ Steel,\ 0\ Panel,\ 1\ Battery,\ 0.006\ m]$$

G. Heuristic Optimization

We implemented a Generic Algorithm (GA) as our first attempt for optimizing the mass of the cubesat. Note that our problem formulation contains 3 discrete variables “Propellant,” “Thruster Type,” and “Structure Material.” These discrete variables might cause difficulty in optimizing by MATLAB’s Sequential Quadratic Programming or other gradient based methods. The GA however can handle discrete variables very nicely. The GA method is subject to tuning and contains randomness but we found that it could be very consistent in giving the three best discrete variables. We saw that each set of discrete variables was like a different valley in the global design space. Figure 7 shows our GA optimal result and Table 5 shows the configuration of the discrete variables which resulted.

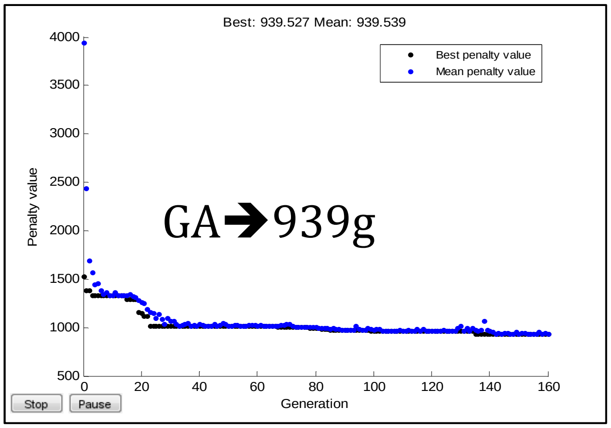


Figure 7. GA Optimal Result

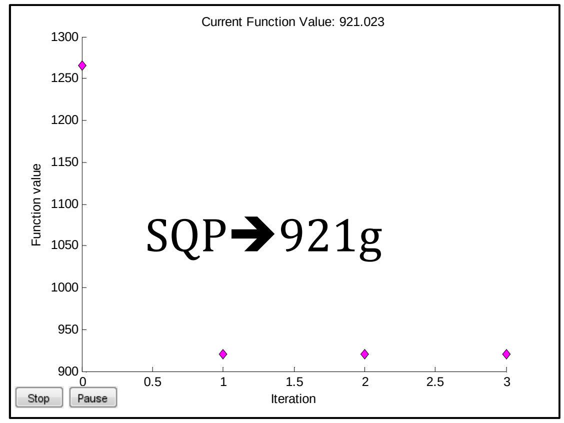


Figure 8. Gradient-Based Optimization

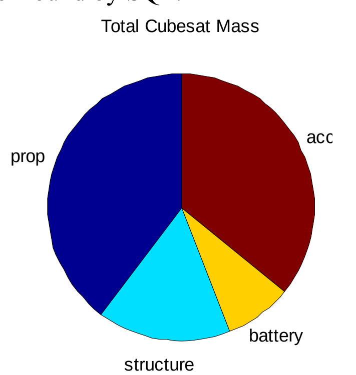
Table 5. GA Optimal Design Configuration

Discrete Variables	Optimal Value
Propellant	Xenon
Thruster Type	Lee LHDB0542115H
Material Type	Al 6061 T6

We experimented with tuning the GA. The algorithm takes between 100 and 200 iterations to converge on its own, but the maximum iterations were set to 160 because no significant improvements are made beyond this point. Usually GAs are computationally expensive but our low fidelity model could be run in a few seconds.

H. Gradient Based Optimization

We chose Sequential Quadratic Programming (SQP) as our gradient-based algorithm. SQP is one of the most powerful gradient-based algorithms, is well researched already, and is built into the MATLAB "fmincon" function. We found it effective to locate a valley in the global design space using the GA, fix the discrete variables, and then pass them to SQP to solve the continuous variables. Figure 8 shows that in four iterations the fmincon was able to take the point given by the GA and reduce it to a guaranteed local minimum. Table A within Figure 9 describes the x^* design vector found by SQP.



	prop	structure	battery	accy
mass (g)	366	149	76	330

Table A: Design Vector

	Value
total mass (g)	921
propellant	Xenon
thruster	lee LHDB054...
material	Al 6061 T6
solar panels	0.6905
batteries	1
structure width(mm)	6.3537

Table B: Relevant Data

	Value
algorithm run time (s)	6.6184
prop tanks (qty)	2
propellant mass (g)	85.2459
pwr consumption (Wh)	327.6000
solar pwr produced(Wh)	313.2000

Figure 9. SQP Optimal Configurations with breakdown components

I. Model Validation

A graphical output was designed in MATLAB to aid in verifying that the model runs as intended. Within Figure 9 there are tables that break down some of the outputs of each model. For example, the final mass may seem reasonable, but we can have more confidence if we check that each module's mass is reasonable and updating correctly from the last run of the program. The graphical output was critical to debugging, adjusting parameters, and comparing tuning scripts.

In addition to validating the model itself, there is some evidence which validates the optimizer's results. The design variables chosen by our model were also used in past successful designs. Xenon is the cold gas of choice for the NASA miniAERcam craft [5], and Lee Co thrusters are used in a cubesat thruster module sold by Micro Aerospace Solutions. It is also common knowledge that 6061 Aluminum is used in most cubesat chassis, and solar panels are maximized. We can consider our solar panels maximized because the x* always takes batteries to the minimum bound and accounts for the remaining draw with solar power.

J. Sensitivity Analysis

Table 6 summarizes some relevant sensitivity information on parameters, constraints, and bounds. The given parameters were chosen because they have some likelihood of changing in the future. The sensitivity data in the table was given by simply reoptimizing with the new parameters. The battery lower bound is important because in it is always active and is likely to change to improve the model. Our model uses a lower bound fixed at 1 (because we know the cubesat needs at least a small amount of power storage) but we would like to implement a calculation of how much storage is really needed and change this to a constraint.

Table 6. Sensitivity of Mass Objective to Parameters

Mass sensitivity to Parameters		
Parameter	**Initial Value**	**Normalized sensitivity (% J) ÷ (% Δparameter)**
Mission duration	30 days	0.2146
Solar Panel Power	2.1 watts	-0.04906
Battery Capacity	4,000 mAh	0 (qty doesn't change)
Flight Disturbance	0.5 m/s/day	0.3070
Volume of a propellant tank	16.34 mL	-0.18311
Constraint Sensitivity		
Power required	1.0 Factor of Safety	0.030797
structure bending stiffness	0.1579 [mass][length]^{3} [time]$^{-2}$	0.584332
Active bounds Sensitivity		
Battery lower bound	1.0 quantity	0.040761
Rail thickness lower bound	3.0 mm	Only active if Titanium is used!

To get the normalized sensitivity of the parameters, we perturbed each parameter in both positive and negative directions by ten percent. If the cost was increased by increasing the parameter value, it has a positive sensitivity.

In the design vector sensitivity analysis, we treated the first 3 variables as parameters like we did in the gradient based optimization. We only considered [#solar panels, #batteries, structure width] as the design vector. The optimal solution that we used was:

$$\boldsymbol{x}^* = [Xenon,\ \ \text{Lee},\ \ Al\,6061\,T6,\ \ 0.6905,\ \ 1,\ \ 0.006357\,m]$$

$$\text{J}(x^*) = [6353.7 \quad 921]$$

We then calculated the numerical gradient of J(x*) using the finite central difference method and then calculate the normalized gradient objective vector. The gradient vector is:

$$\nabla J = \begin{bmatrix} \frac{\partial Cost}{\partial X^*} & \frac{\partial mass}{\partial X^*} \end{bmatrix} = \begin{bmatrix} 2825 & 46.62 \\ 10.01 & 46.10 \\ .0455 & 67.07 \end{bmatrix} \quad \textbf{Eq. (2)}$$

The normalized gradient at the optimal is:

$$\nabla \bar{J} = \frac{x^*}{J(x^*)} * \begin{bmatrix} \frac{\partial Cost}{\partial X^*} & \frac{\partial mass}{\partial X^*} \end{bmatrix} = \begin{bmatrix} 0.3654 & 0.0292 \\ 0.0019 & 0.0418 \\ 5.42 * 10^{-5} & 0.3862 \end{bmatrix} \quad \textbf{Eq. (3)}$$

We can see that the the number of panels has a huge impact on the cost compared to the other variables while the structure width has a large impact on the mass.

V. Multi Objective Optimization

For the multi objective part of this project we introduced cost as part of the objective function. An adjusted weighted sum (AWS) was used as the objective function as shown in Eq. (4) with lambda incremented from zero to one by 200 equal steps.

$$J = cost(\lambda) + mass(1 - \lambda) \quad \textbf{Eq. (4)}$$

Figure 10 shows that the AWS method appears to capture all of the Pareto optimal design points (indicated with circles). The lambda steps were made finer, the genetic algorithm was tuned, and the weighting was adjusted to try to push the optimizer towards the low cost area instead of the low mass area and probe for more points. The details found by the four optimal points from the AWS are listed in Table 7. Pareto Optimal Designs

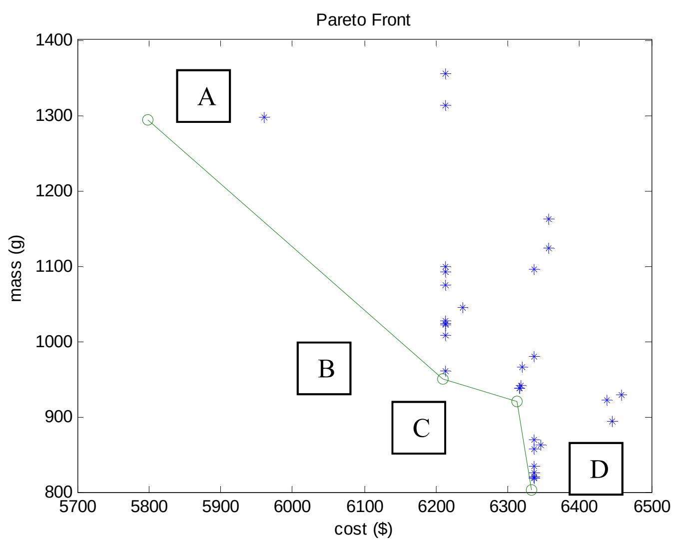


Figure 10. Pareto Front

Table 7. Pareto Optimal Designs

	A	B	C	D
Prop	Freon	Xenon	Freon 14	Methane
Thruster	Lee	Lee	Marotta	Marotta
Material	Ti	Al	Al	Steel
S.Panel	0.69	0.69	0.69	0.76
Battery	1	1	1	1
Rail (mm)	3.04	6.35	6.35	3
Cost	$6,333	$6,312	$6210	$5,798
Mass (g)	802	921	950	1,293

The blue stars in Figure 10 are points produced by only the genetic algorithm. We can infer from the arrangement of points that the GA somewhat discretizes the points into different costs and then locates fairly low masses. The SQP can locate then locate the minimum mass vector for a design in that cost area. All of the pareto-optimal points presented are viable options for final selection depending on the needs and budget of the particular cubesat mission. One open question from this pareto front is why the AWS was able to find a design which has an overall lower mass at 802 grams than the optimization using mass alone as the objective. This is a major finding and warrants a detailed investigation into the program codes.

VI. Results & Conclusion

K. Discussion of Results

Considering that the cost of delivering a fully equipped cubesat to space is often on the order of $100k for hardware and $100k for transportation, we can say that it may be safe to ignore cost and search for the minimal mass if it saves computational expense. That is, for the model we have developed and the information given by the pareto front, the weight savings of the best designs are significant enough to merit the higher prices. Once again, this tool is designed to create a cubesat *platform* and every gram and cubic millimeter saved is creating more room for the scientist to equip with important payload components.

One exciting result of the study is that it does its job almost instantaneously. For the application at hand, fast reoptimization is very useful. A multi-million dollar jet with a high manufacturing volume may be worth running weeks of high-fidelity models because it can reap the benefits for a long term payback. Each cubesat on the other hand is a one-time development and will have dozens of parameters changed from the previous mission. Now, it is easy to justify running this agile optimizer tool multiple times during the development process.

In the end the lack of many continuous design variables made the results somewhat less interesting. If more data could be gathered and put into the model, the resulting Pareto front could become more continuous and give more useful options for the decision-makers. Perhaps several tempers of 6061 aluminum could be populated into the structure module, whereas at this point the designer still has to make that important decision blindly even if the optimizer concludes with an aluminum chassis! The more data that can be poured into the program, the better.

L. Conclusion

Our project shows promising results for both of the mass and cost optimization. We combined 3 different optimization method in our design, DOE, heuristic GA and gradient based SQP and found an improved and validated solution. Each method has its own weakness, orthogonal array DOE doesn't fully explore all design options and can't handle non-discrete variables, heuristic method is expensive and subjective to tuning, and SQP can't handle discrete variables. However, when combined into a hybrid solution, they gave out a very favorable result without consuming much computational resources. On the other hand, due to our team limitation in resource and expert knowledge in the subject, our module has a very low fidelity. In the future, we would like to expand our project by increasing the fidelity of our model. We already know the impact difference of each design variable and module on the result vector, thus it is possible to invest more resources to improve the modules, and improve the performance for more realistic, results with high confidence.

Appendix: DOE Tables

Orthogonal array of design vectors for DOE:

Exp	Propellant	Thruster type	Material	#panels	#batteries	Rail width (m)	Cost ($E3)	Mass (g)
1	Hydrogen	moog	Al 6061 T6	0	1	0.003	5.4046	1.8548
2	Hydrogen	marotta	Titanium	1	2	0.006	8.7636	3.2275
3	Hydrogen	lee	A36 steel	4	3	0.008	11.884	4.9757
4	Nitrogen	moog	Al 6061 T6	1	2	0.006	8.2382	1.113
5	Nitrogen	marotta	Titanium	4	3	0.003	11.616	1.4386
6	Nitrogen	lee	A36 steel	0	1	0.006	6.0232	1.0183
7	Xenon	moog	Titanium	0	3	0.006	11.35	1.3172
8	Xenon	marotta	A36 steel	1	1	0.008	6.0203	2.1013
9	Xenon	lee	Al 6061 T6	4	2	0.003	9.1701	0.9275
10	Hydrogen	moog	A36 steel	4	2	0.006	8.2585	4.0441
11	Hydrogen	marotta	Al 6061 T6	4	3	0.008	11.1484	5.243
12	Hydrogen	lee	Titanium	1	1	0.003	6.3379	2.1708
13	Nitrogen	moog	Titanium	4	1	0.008	5.5663	1.726
14	Nitrogen	marotta	A36 steel	0	2	0.003	8.6156	1.3175
15	Nitrogen	lee	Al 6061 T6	1	3	0.006	11.863	1.1901
16	Xenon	moog	A36 steel	1	3	0.003	11.1164	1.0619
17	Xenon	marotta	Al 6061 T6	4	1	0.006	5.9764	1.2416
18	Xenon	lee	Titanium	0	2	0.008	9.2996	1.3576

DOE Analysis – Main Effects

Variable (Factor)	**Level**	**Cost ($E3)**	**ME Cost**	**Mass (kg)**	**ME Mass**
Propellant	Hydrogen	$8.633	-$0.070	3.586	$1.512
	Nitrogen	$8.654	-$0.049	1.301	-$0.773
	Xenon	$8.822	$0.119	1.335	-$0.739
Thruster Type	moog	$8.322	-$0.381	1.853	-$0.221
	marotta	$8.690	-$0.013	2.428	$0.355
	lee LHDB0542115H	$9.096	$0.393	1.940	-$0.134
Structure Material	Al 6061 T6	$8.633	-$0.069	1.928	-$0.145
	Titanium	$8.822	$0.119	1.873	-$0.201
	A36 steel	$8.653	-$0.050	2.420	$0.346
Number of Solar Panels	0	$8.139	-$0.564	1.373	-$0.701
	1	$8.723	$0.020	1.811	-$0.263
	4	$9.089	$0.386	2.800	$0.726
Number of Battery	1	$5.888	-$2.815	1.685	-$0.388
	2	$8.724	$0.021	1.998	-$0.076
	3	$11.496	$2.793	2.538	$0.464
Structure Width (m)	0.003	$8.710	$0.007	1.462	-$0.612
	0.006	$8.639	-$0.064	1.879	-$0.195
	0.008	$8.784	$0.081	3.081	$1.007

Acknowledgments

This project could not be completed without the great help of Dr. Douglas Allaire, Deparment of Mechanical Engineering, Texas A&M University. Our group are thankful and indebted to him for sharing his expertise, and sincere and valuable guidance.

References

[1] "CubeSat Design Specification (CDS)," California Polytechnic, San Luis Obispo, 2009.

[2] Juergen and Mueller, "Thruster Options for Microspacecraft: A Review and Evaluation of Existing Hardware and Emerging Technologies," *AIAA,* 1997.

[3] Quaker Chemical Corporation, "quakerchem.com," 2013. [Online]. Available: http://www.quakerchem.com/wp-content/uploads/pdf/skill_builders/no10_machinability_ratings.pdf. [Accessed 11 05 2015].

[4] D. Hsu, 30 4 2015. [Online]. Available: http://www.chemicool.com/.

[5] S. E. Fredrickson, S. Duran and J. D. Mitchell, "Mini AERCam Inspection Robot for Human Space Missions".