DESIGN OF A RESILIENT RIDESHARE-BASED SMALL SATELLITE CONSTELLATION USING A GENETIC ALGORITHM

Katherine Mott Wagner and Jonathan T. Black
Virginia Polytechnic Institute and State University, Blacksburg, VA 24061

Responsive and resilient space-based systems are needed to satisfy changing mission requirements and react to unforeseen challenges. This paper studies the ability of a constellation constructed from commercial-off-the-shelf parts and launched using rideshare to provide imaging coverage over a small region in the event of a disaster, such as an outbreak of wildfires. A genetic algorithm and model-based systems engineering techniques are used to evaluate rideshare constellations in both the nominal case and the case in which some satellites have failed. Novel methods for determining reachability between two orbits and for determining revisit metrics for degraded constellations are presented.

I. Introduction

The use of small satellites in both industry and academia is increasing as a result of both the miniaturization of satellite components and the availability of commercial off-the-shelf (COTS) components. As the number of nanosatellites (1-10kg satellites) and other small satellites continues to increase, new tools and methodologies are needed to accommodate the unique challenges and capabilities of these systems. The benefits of nanosatellites include the availability of a standard form factor (the CubeSat), low cost, COTS components, and short build times. The disadvantages of nanosatellites include reduced capability, shorter lifetimes, higher failure rates, and a lack of cost-effective launch opportunities.

The disadvantages outlined above can be mitigated through intelligent design that keeps the limits of nanosatellites in mind. For example, the capabilities of a large satellite can be replicated by launching several nanosatellites with different payloads, a process known as disaggregation. Short satellite lifetimes require replenishing the constellation as time passes if the mission lifetime exceeds the satellite lifetime. The low cost of each nanosatellite can make such a method more cost effective than a single satellite with a long lifetime, in some cases. Furthermore, nanosatellites are appropriate for missions of short duration or for providing a temporary solution while a more permanent system is designed and manufactured. The final two issues, high failure rates and launch limitations, are the focus of this paper.

Unabridged paper presented at the AAS/AIAA Space Flight Mechanics Meeting 2019 [1]

II. Background

Spacecraft can fail for a variety of reasons, including launch failures, radiation, thermal stresses, and electronics failures [2]. The small form factor, low-cost satellites known as CubeSats are particularly susceptible to failure due to their use of COTS parts, low budgets, and high risk tolerance. A 2017 study found that academic CubeSat missions failed 55% of the time, while commercial CubeSat endeavors failed 23% of the time [3]. Despite their failure rate, CubeSat constellations can enable critical space missions by providing rapid response due to their short build times. Additionally, their small size allows them to be launched as secondary payloads when their mission has some flexibility in the required orbital configuration.

However, the effect of the failure of one or more satellites on the ability of the constellation to perform its mission must be assessed. Previous work has measured constellation resilience based on the predicted failure rate [4] and the predicted number of satellites on orbit [5]. Stenger performed network analysis for a degraded Iridium constellation, selecting the worst-case removals in batches of 12 by finding the satellites that appeared most often in the packet paths and removing them [6]. However, this method is not mathematically rigorous for nonuniform constellations because the problem of satellite access cannot be solved recursively. It is therefore beneficial to develop a rigorous methodology for determining the satellites whose removal is most damaging to the constellation performance.

Another issue in the deployment of nanosatellite constellations is getting all of the assets into orbit. In traditional spacecraft constellations, the launch costs are a relatively small portion of the overall budget—one example scenario predicts a 14% launch cost [7]. In the case of nanosatellites, however, the cost of the spacecraft itself is much smaller, a couple million dollars at most. Some nanosatellites are as cheap as a couple hundred thousand

dollars[8]. Because launch vehicles cost tens to hundreds of millions of dollars, the use of dedicated launches for low cost missions is infeasible unless hundreds of satellites are going to the same plane[9]. However, small satellites can be launched as secondary payloads via rideshare programs for about $30,000 per kilogram[8]. Additionally, academic groups may qualify for free launch services through the Educational Launch of Nanosatellite (ELaNa) missions[10]. The downside of constellations built using rideshare alone, sometimes called ad hoc constellations, is that the irregular distribution of satellites results in large gaps in coverage compared to a symmetric constellation like a Walker constellation. Previous studies have quantified these differences, but have shown that performance can be improved through optimization of the rideshare selection. One such study used a Monte Carlo analysis to characterize the range of performance for ad hoc constellations providing global coverage[11]. Another study used a multi-objective genetic algorithm to determine an optimal rideshare manifest for providing global coverage[12]. That paper also discussed resiliency of ad hoc constellations, though only for a specific solution produced by the genetic algorithm and not as an optimization criterion.

III. Methodology

The Disaggregated Integral Systems Concept Optimization Technology (DISCO-Tech) methodology was used to formulate and solve a rideshare reconfiguration constellation optimization problem. The DISCO-Tech algorithm is modular, with each module performing a different task of the optimization. Key modules are described below and in previous works[1,13,14].

A. Optimization

DISCO-Tech uses a modified version of the epsilon nondominated sorting genetic algorithm II (eNSGA-II) to solve multiobjective optimization problems[15]. It combines the epsilon dominance feature of eNSGA-II with the archive feature of the BORG genetic algorithm but maintains the use of generations to facilitate ease in parallelization[16]. It differs from BORG and eNSGA-II in that it uses a variable length crossover operation, as described in a previous paper[13].

B. Reachability

Previous papers on constellation reconfiguration have restricted analysis to specific sets of maneuvers. One study restricted reconfiguration to in-plane maneuvers, then used a genetic algorithm to solve for the two-burn transfers yielding the best coverage in the final configuration[15]. Other studies restrict the initial and final constellations to known sets of orbits, presolving for the fuel needed to go between each combination of orbits then solving the assignment problem to find the optimal set of transfers[17–19].

A general framework was desired to determine the reachability of one orbit from another when neither the orbits are known a priori nor are the initial and final orbits confined to the same plane. Although methods exist for generating the reachable set[20–23], they rely on numerical simulation and are too computationally expensive to call for each solution during the optimization, since the reachable set will change as the initial orbit changes.

Instead, an estimation of the fuel used is generated using a linearized version of Gauss's Variational Equations (GVE). Previous research linearized GVE about the final orbit and used the resulting equations with model predictive control (MPC) to calculate the required controls to maneuver from one orbit to another[24]. This linearization serves as the basis for our approach, though the MPC process was deemed unnecessarily costly. We need only the total fuel expenditure, not the entire control history. Furthermore, we improve upon the linearization through the use of the modified equinoctial orbital elements, by treating the true longitude as an independent parameter, and by improved analysis of the validity of the linearization. It is assumed that the final value of the true longitude is irrelevant, as it can be set afterward by temporarily raising or lowering the orbit using a comparatively small amount of fuel or by holding the satellite at an intermediate stage in its orbit until the desired phasing has been reached. It is also assumed that the maximum acceleration of the spacecraft does not change over time despite the change in the spacecraft's mass.

GVE are of the form

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}) + B(\mathbf{x})\mathbf{u} \tag{1}$$

where $\mathbf{x}$ is the vector of orbital elements and $\mathbf{u} = [u_r, u_\theta, u_h]^T$ are the control accelerations in the local vertical local horizontal (LVLH) frame. This formulation uses the set of modified equinoctial orbital elements (MEOE), a set of nonsingular elements defined by Walker, Ireland, and Owens[25]. The MEOE are denoted by $\mathbf{x} = [p, f, g, h, k, L]^T$, where the true longitude L is the only rapidly changing variable. p is the semiparameter of the orbit. The remaining four elements lack obvious physical meaning but are defined as $f = e\cos(\omega + \Omega)$, $g = e\sin(\omega + \Omega)$, $h = \tan(\frac{i}{2})\cos(\Omega)$, and $k = \tan(\frac{i}{2})\sin(\Omega)$, where e is the eccentricity, i is the inclination, ω is the argument of periapsis (AOP), and Ω is the right ascension of the ascending node (RAAN). $\mathbf{f(x)} \in \mathbb{R}^6$ shows the growth of the elements in the absence of control, and $B(\mathbf{x}) \in \mathbb{R}^{6\times3}$ is the input effect matrix. These matrices can be constructed from the orbital element rate equations[25].

With the exception of true longitude, the orbital elements are constant in the absence of perturbations like oblateness effects. Form a reduced set of elements

$\mathbf{z} = [p, f, g, h, k]^T$. Since the first five elements of $\mathbf{f(x)}$ are zero for the two body problem, the growth of these elements can be written as

$$\frac{d\mathbf{z}}{dt} = \overline{B}(\mathbf{z}, L)\mathbf{u} \tag{2}$$

where $\overline{B}(\mathbf{z}, L)$ is the first five rows of $B(\mathbf{x})$. Note that the equation is affine in $\mathbf{u}$ but nonlinear in $\mathbf{z}$ due to the dependence of $\overline{B}$ on $\mathbf{z}$.

It is advantageous to treat the true longitude L as an independent parameter. The growth of L over time is

$$\frac{dL}{dt} = \sqrt{\frac{\mu}{p^3}}(1 + e\cos(\nu))^2 + \sqrt{\frac{p}{\mu}}\frac{\tan(0.5i)\sin(\nu + \omega)}{1 + e\cos(\nu)}u_h \tag{3}$$

When the thrust is sufficiently small, its impact on the true longitude can be neglected. It has been shown that this assumption holds for low-thrust nanosatellite propulsion systems and for high-thrust nanosatellites with thrusts of up to 1.25N in nearly circular, low-Earth orbits[1]. The rate of change of L is then $dL/dt = \sqrt{\mu p}q^2/(p^2)$. The values of L can be approximated either by holding the MEOE fixed at either the initial or final values or by linearly interpolating between the initial and final values and calculating the growth of L at each time step. Using this process, true longitude can be treated as a function of time alone and can be precalculated, allowing it to be treated as an independent parameter in the linear program.

Linearization of Eq. (2) about some stationary orbit $\mathbf{z}_s$ gives

$$\frac{d\mathbf{z}}{dt} = \left(\overline{B}(\mathbf{z}_s, L) + \frac{\partial\overline{B}}{\partial\mathbf{z}}\bigg|_{\mathbf{z}=\mathbf{z}_s}\Delta\mathbf{z}\right)\mathbf{u} + HOT \tag{4}$$

where $\Delta\mathbf{z} = \mathbf{z} - \mathbf{z}_s$. $\partial\overline{B}/\partial\mathbf{z}$ in the second term is a tensor of rank three. Neglect the higher order terms.

The second term results in a nonlinear equation, since the term contains a product of $\Delta\mathbf{z}$ and $\mathbf{u}$. Because this formulation is only an approximation of the total fuel used, we drop this term to maintain the linearity of the system. This simplification is valid if the magnitude of the derivatives of $\overline{B}$ are small compared to the values of $\overline{B}$ itself. We shall examine the ranges over which this assumption is valid.

Both $\overline{B}$ and $\partial\overline{B}/\partial\mathbf{z}$ vary with L, so it is necessary to examine the relative behavior of the two terms over an entire orbit. Furthermore, due to the differences in magnitude of the rates of change of the various orbital elements, it is prudent to examine each one separately. To determine the allowable extent of the nonlinearity due to a variation in one of the MEOE, set a bound on the ratio of the magnitudes of the nonlinear and linear terms causing a change in element i due to the difference in element j from the stationary orbit,

$$R_{ij} = \frac{|\Delta z_j|\sqrt{(\partial b_{i1}/\partial z_j)^2 + (\partial b_{i2}/\partial z_j)^2 + (\partial b_{i3}/\partial z_j)^2}}{\sqrt{b_{i1}^2 + b_{i2}^2 + b_{i3}^2}} \tag{5}$$

where b_{ij} is the element of B in the i-th row and the j-th column. By setting an upper limit on the value of R_{ij}, we can develop bounds on each of the orbital elements. Some elements of $\overline{B}$, such as b_{21}, go to zero at certain values of L. As a result, the ratio near these points is poorly defined. Furthermore, a large R_{ij} value occurring when $\mathbf{b}_i$ (the i-th row of B) is small still results in a small magnitude change in $\mathbf{b}_i$. In order to avoid these singularities, the denominator used in Eq. (5) is not the instantaneous value for a given L but the average value calculated by averaging $\|\mathbf{b}_i\|$ over L. Call this average value $\|\mathbf{b}_i\|_{avg}$. The bounds on Δz_j can then be written as

$$|\Delta z_j| \leq \min_{i\in[1,5]}\left(\min_{L\in[0,2\pi]}\left(R_{max}\frac{\|\mathbf{b}_i\|_{avg}}{\|\mathbf{j}_{ij}(L)\|}\right)\right) \tag{6}$$

where $\mathbf{j}_{ij} = [\partial b_{i1}/\partial z_j \;\; \partial b_{i2}/\partial z_j \;\; \partial b_{i3}/\partial z_j]$.

Using the process described above, bounds on the linearization are generated for an orbit with nominal values of $a = 7000$km, $e = 0.1$, $i = \pi/4$, $\Omega = \pi/6$, and $\omega = \pi/12$. In order to determine the impact of the initial value of each orbital element on the bounds, the initial orbital elements are varied one at a time while holding the others fixed. a is varied from 6678km to 16378km, e is varied from 0 to 0.8, i is varied from 0 to 180°, Ω is varied from 0 to 360°, and ω is varied from 0 to 360°. Fig. 1 shows the results for the most interesting relationships. The upper bound is shown in blue, the nominal value in yellow, and the lower bound in red.

The main determining factor in the semimajor axis bounds is the semimajor axis, with larger semimajor axes having larger bounds. For a low Earth orbit, a limit of $|\Delta a| = 500$km ensures the bounds on the linearization are satisfied. Similarly, eccentricity is the main determining factor on the eccentricity bounds with larger eccentricities having smaller bounds, though RAAN, AOP, and high inclination influence the eccentricity bounds as well. For an orbit with low eccentricity, bounds of about 0.1 are acceptable. The inclination bounds decrease with increasing inclination and eccentricity. However, the bounds are so large and the fuel required to enact a change in inclination so great that it is unlikely that a satellite would maneuver more than a couple of degrees in inclination, rendering the linearization valid for all practical cases barring a retrograde orbit with inclination greater than 120°. Likewise, the bounds on RAAN and AOP are large and unlikely to be exceeded, so these plots are not included.

Overall, the linearization will hold when the semimajor axis error is kept below 500km, the eccentricity error

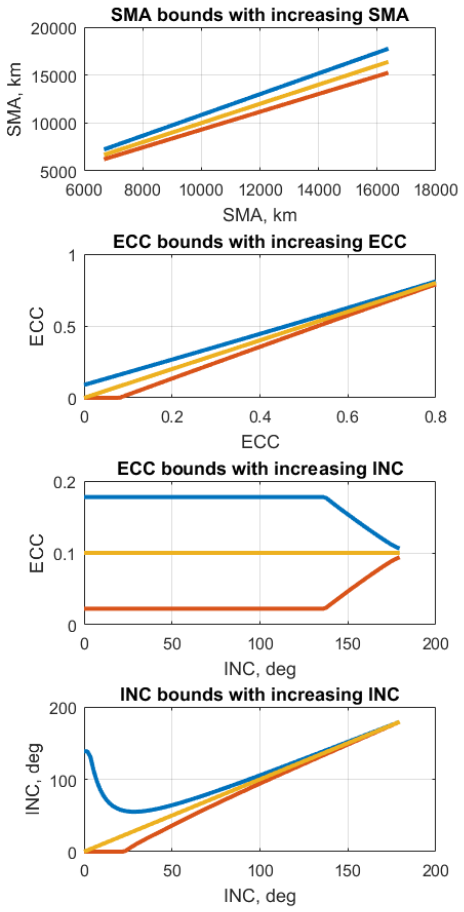

Fig. 1 Element limits for valid linearization under changing initial element values.

kept below 0.1 for low eccentricities and below 0.05 for eccentricities near 0.5, the eccentricity kept below 0.5, the inclination kept below 120°, changes in AOP kept below 50°, and changes in RAAN and inclination kept below 10° for prograde orbits and 5° for retrograde orbits.

Now that it has been shown that the nonlinear term in Eq. (4) can be neglected in the cases described above, we can use the linear equation

$$\frac{d\mathbf{z}}{dt} = \overline{B}(\mathbf{z}_s, L)\mathbf{u} \tag{7}$$

to approximate the change in the orbital elements over time.

Applying the variation of constants formula to the above equation gives

$$\mathbf{z}(t) = \mathbf{z}_0 + \int_{t_0}^{t} \overline{B}(\mathbf{z}_s, L(v))\mathbf{u}(v)dv \tag{8}$$

If we discretize the system by setting $t = k\Delta t$, using it to calculate $L_k = L(k\Delta t)$, and treating $\overline{B}$ and $\mathbf{u}$ as being fixed at each time step, the equation becomes

$$\mathbf{z}_{k+1} = \mathbf{z}_0 + \Delta t \sum_{j=0}^{k} \overline{B}_j \mathbf{u}_j \tag{9}$$

With this equation, the problem of reachability can now be formulated as a linear program (LP). To enforce the upper and lower bounds on u, create separate variables u^+ and u^-, both in $[0,u_{max}]$, such that $u = u^+ - u^-$. The LP formulation can be written as

$$\text{Minimize} \sum_k \sum_i u_{ik}^+ + u_{ik}^-$$

with decision variables :

$$u_{ik}^+ \in [0,u_{max}] \; \forall \; k \in [0..k_f], \; i \in \{r,\theta,h\}$$
$$u_{ik}^- \in [0,u_{max}] \; \forall \; k \in [0..k_f], \; i \in \{r,\theta,h\}$$

such that :

$$\mathbf{z}_{fd} - \zeta_f \leq \mathbf{z}_0 + \Delta f \sum_{j=0}^{k_f} \overline{B}_j \mathbf{u}_j \leq \mathbf{z}_{fd} + \zeta_f$$

where ζ_f is the allowable error in the final state, and the desired final state is $\mathbf{z}_{fd}$. Due to the fact that the control is held constant during each step, reaching an exact state may be impossible, requiring the inclusion of error bounds. Once the optimal solution is known, the orbital elements at each step k can be calculated using Eq. (9). The total fuel required can be approximated as the value of the objective function times Δt.

Two simple scenarios are used to test the problem formulation. First, the problem of raising a circular orbit from 1000km in altitude to 1500km in altitude using a maximum acceleration of 0.01m/s^2 is considered. The optimal low thrust orbit raise for a circular orbit is a continuous thrust in the velocity vector direction and consumes a total delta-v of $\Delta v = \left|\sqrt{\mu/a_0} - \sqrt{\mu/a_f}\right|$[26]. The time needed to complete the maneuver is $t_f - t_0 = (\mu/u_{max})\left|a_0^{-1/2} - a_f^{-1/2}\right|$. For the given problem, the required Δv according to the equation is 237.1m/s. Solving the MILP using a commercial MILP solver produces an estimated velocity requirement for the orbit raise is 235.4m/s, a deviation from the analytical solution of less than 1%.

The second scenario requires an inclination change of two degrees within ten orbits. The orbit is circular and has an altitude of 1000km. The Δv required for an impulsive transfer is $\Delta v = 2v \sin\frac{\Delta i}{2}$. If multiple small impulsive manuevers are performed rather than a single large maneuver, the required velocity change is $\Delta v = 2nv \sin\frac{\Delta i}{2n}$, where n is the number of maneuvers. Assuming the maneuver is conducted using a max acceleration of 0.01m/s^2

over the course of forty orbits (eighty maneuvers), the required Δv = 256.6m/s. Since only about three m/s is required per maneuver, it is sufficient to assume the Δv required for an instantaneous small inclination change is approximately equal to that required for an inclination change maneuver spread over a couple of minutes using low thrust. The result from the equation should be comparable to that provided by the linear program. Indeed, the linear program predicts a Δv of 253.9m/s, yielding an error of 1%. These simple examples, when combined with the mathematical validation provided above, bound the accuracy of the linearization approach in predicting the Δv required for a low Earth orbit transfer. The predicted Δv can then be used to determine the reachability of one orbit from another.

C. Dynamics

The scenario is divided into periods during which no maneuvers occur. During such a period, the orbital elements of the satellites over time are found using the mean anomaly, since $M - M_0 = nt$, where n is the mean motion. The mean anomaly is then used with Kepler's equation, $M = E - e\sin(E)$, to obtain the eccentric anomaly E. Since Kepler's equation is transcendental in E, bicubic interpolation is used to solve for E. E can then be used to obtain ν. The change in the other classical orbital elements over time is calculated using the secular growth caused by J_2 and J_4 [27]. All maneuvers are assumed to be instantaneous. The scenario is propagated until a maneuver time is reached. The change in orbital elements or in velocity is then added to the current state. Propagation then continues until the next maneuver is reached.

D. Coverage

Because the calculation of satellite revisit metrics is nonlinear and computationally expensive, we compute the rise and set times for the nominal constellation, the satellite constellation from which no assets have been removed. The method used for calculating the rise and set times is adapted from the methods developed by Alfano [28]. For each ground station, a matrix is constructed to describe the access to that station over time. The rows correspond to the sorted rise and set times, while the columns correspond to the satellites. The matrix is binary such that a one in the (i, j) place indicates that the j-th satellite can access the ground station from the i-th time until the $(i + 1)$-th time. There is a corresponding vector of times, T_k, to match the rows of the matrix. Form an access array A by concatenating the accesses for each station and a time matrix T. T_{ik} is the i-th rise or set time for the k-th station. A_{ijk} is one if the j-th satellite can access the k-th ground station between T_{ik} and $T_{(i+1)k}$. Also calculate the length of each time step, $\Delta T_{ik} = T_{(i+1)k} - T_{ik}$. The access array and time matrix can be used to calculate revisit metrics and in the resilience calculations outlined in the following section.

E. Resilience

This section discusses a method for formulating the problem of finding the combination of losses of assets most damaging to the constellation performance as a mixed integer linear program (MILP). The exact formulation will depend on the metric used in the optimization, but here we treat only the problem of finding the combination of removals that maximizes the longest gap in coverage seen by any of the ground stations (maximum revisit time over all points). Call the number of satellites in the nominal constellation n_s, the number of removals n_r, and the number of ground stations n_g.

Define the decision variable $x_j \in \{0, 1\}\, \forall\, j = 1, 2, \ldots, n_s$, a binary satellite inclusion variable that is one if the j-th satellite is active and zero otherwise. To produce the correct number of removals, we define the constraint

$$\sum_{j=1}^{n_s} x_j = n_s - n_r \tag{10}$$

It is then necessary to determine how many satellites are available to each station at any given time. The access sum, $\overline{A}_{ik}$, gives the number of satellites available to ground station k at its i-th time step,

$$\overline{A}_{ik} = \sum_{j=1}^{n_s} A_{ijk} x_j \tag{11}$$

We can then calculate whether a sufficient number of assets are available for access at each period. Define n_c as the number of assets required to be in view of the ground station simultaneously for successful access. The calculation of this access requires the introduction of a new binary variable, $Y_{ik} \in \{0, 1\}$. Y_{ik} is one if the required number of assets are accessible by station k at time i and zero otherwise. In order to force Y_{ik} to take the appropriate value, we introduce the following constraints. Note that the second is a big-M constraint. Set $M = n_s - n_c + 1$.

$$Y_{ik} \leq \frac{\overline{A}_{ik}}{n_c} \qquad \forall\, i \in [1..n_t - 1], k \in [1..n_g] \tag{12}$$

$$MY_{ik} \geq \overline{A}_{ik} - n_c + 1 \qquad \forall\, i \in [1..n_t - 1], k \in [1..n_g] \tag{13}$$

Eq. (12) ensures that Y_{ik} is zero if insufficient satellites are available. Eq. (13) ensures that Y_{ik} is one if at least n_c satellites are available.

The maximum revisit time of a ground station is the longest period for which that station is without coverage.

The maximum of the maximum revisit times is the largest gap in coverage for any station in the scenario. The formulation begins with the definition of an accumulator variable $a_{ik} \in \mathbb{R}_{\geq 0}$. The accumulator variable counts the amount of time at each step since the end of the previous pass. During a pass and immediately after the pass ends, the accumulator should be zero. The constraints below are big-M constraints. To distinguish from the big-M value used in Eq. (12), the big-M value in these constraints will be referred to as M_2. The most conservative value for M_2 is the length of the scenario plus a small constant. However, using smaller values to aid convergence is encouraged if it is guaranteed that no gap will ever exceed the value chosen for M_2.

The constraints needed to force a_{ik} to take the appropriate value are slightly different for the first time step than for the rest of the scenario. a_{1k} has the constraints

$$a_{1k} \geq \Delta T_{1k} - M_2 Y_{1k} \; \forall \; k \in [1..n_g] \tag{14}$$

$$a_{1k} \leq \Delta T_{1k} + M_2 Y 1 k \; \forall \; k \in [1..n_g] \tag{15}$$

These constraints ensure that a_{1k} will be equal to the length of the first time step if there is no access when the scenario begins. The constraints for the rest of the time period are

$$a_{ik} \geq a_{(i-1)k} + \Delta T_{ik} - M_2 Y_{ik} \; \forall \; i \in [2..n_t - 1], k \in [1..n_g] \tag{16}$$

$$a_{ik} \leq a_{(i-1)k} + \Delta T_{ik} + M_2 Y_{ik} \; \forall \; i \in [2..n_t - 1], k \in [1..n_g] \tag{17}$$

Likewise, these constraints ensure that a_{ik} will be equal to the previous accumulator value plus the time step if there is no access at the current time. Finally, a_{ik} must be zero if there is access at the current time, so

$$a_{ik} \leq M_2(1 - Y_{ik}) \; \forall \; i \in [1..n_t - 1], k \in [1..n_g] \tag{18}$$

The length of the largest gap is equal to the largest value of a_{ik}. To find this value, introduce a variable $a_{\max} \in \mathbb{R}_{\geq 0}$. Because the goal is to maximize $a_{\max}$, there must be an upper bound on $a_{\max}$ to prevent it growing unbounded. Therefore, it is required that $a_{\max}$ is less than or equal to exactly one of the values of a. To this end, introduce additional binary variables $\delta_{ik} \in \mathbb{R}^{(n_t-1)\times n_g}$. This formulation will drive $a_{\max}$ to the largest value of a and can be enforced with the constraints

$$a_{\max} \leq a_{ik} + (1 - \delta_{ik}) M_2 \; \forall \; i \in [1..n_t - 1], k \in [1..n_g] \tag{19}$$

$$\sum_{i=1}^{n_t-1} \sum_{k=1}^{n_g} \delta_{ik} = 1 \tag{20}$$

The linear programming problem is

$$\begin{array}{lll} \text{Minimize} & -a_{\max} & \\ \text{with} & \text{decision} & \text{variables :} \\ x_j & \in \{0,1\} & \forall \; j \in [1..n_s] \\ Y_{ik} & \in \{0,1\} & \forall \; i \in [1..n_t], k \in [1..n_g] \\ a_{ik} & \in \mathbb{R}_{\geq 0} & \forall \; i \in [1..n_t - 1], k \in [1..n_g] \\ a_{\max} & \in \mathbb{R}_{\geq 0} & \\ \delta_{ik} & \in \{0,1\} & \forall \; i \in [1..n_t - 1], k \in [1..n_g] \\ \text{subject to the} & & \text{constraints in Eqs. 10–20} \end{array}$$

IV. Scenario description

One benefit of both nanosatellites and rideshare launches is the ability to use them on short notice. This feature is especially beneficial in the event of an emergency, when there is insufficient time to build and deploy a traditional satellite constellation. To simulate such a scenario, consider the case of fire detection over California. The goal is to deploy a constellation of nanosatellites constructed of COTS parts using rideshare opportunities in a timely manner. The nanosatellites are identical and have the following subsystems: MPS-130 2U propulsion module by Aerojet Rocketdyne (thrust of 1.25N, specific impulse of 235s, and fuel mass of 1.4kg) *; Helios deployable helical antenna †; ISIS VHF downlink/UHF uplink full duplex transceiver †; BAox high energy density battery array †; Crystalspace P1U Vasik EPS †; CubeSat Solar panel DHV-CS-10 †; CubeADCS 3-Axis with medium wheels †; Chameleon multispectral imager (ground sample distance of 9.6m at 500km) ‡.

With the exception of the propulsion system and the imager, the components listed above serve only to estimate the cost and mass required for the satellite and do not represent a finalized design. The propulsion system dictates the maneuvers that can be performed by the satellite, while the imager dictates the image resolution, limiting the maximum altitude of the satellites.

A set of rideshare options was simulated by taking the two line elements (TLEs) of satellites launched over a thirty day period. This sampling is meant to be an example set of launches and is not indicative of the launches that would be available for an actual mission. The results will vary based on the particular set of launches available. The orbital elements corresponding to the TLEs are shown in Table 1, where each row represents a different launch with the angles are in degrees and the semimajor axis in kilometers.

* http://www.rocket.com/files/aerojet/documents/CubeSat/MPS-130%20data%20sheet%20crop.pdf

† Specifications acquired from https://www.cubesatshop.com/

‡ http://41.185.8.177/~cubespac/ClientDownloads/CubeADCS_3Axis_Specsheet_V1.1.pdf

Table 1 Rideshare orbital elements

Launch	a	e	i	ω	Ω
1	6823	0.0018	92.9	118.3	253.5
2	6823	0.0018	92.9	118.6	253.5
3	6823	0.0017	92.9	124.2	253.5
4	6823	0.0018	92.9	124.6	253.5
5	6966	0.0014	97.9	153.2	158.3
6	6966	0.0014	97.9	151.1	158.3
7	28241	0.0117	55.0	176.3	156.9
8	28243	0.0116	55.0	176.4	156.9
9	7090	0.0091	98.6	337.4	340.2
10	15700	0.5808	55.0	172.5	153.7
11	15531	0.5723	26.9	195.9	240.4
12	7161	0.0011	98.6	165.3	339.7
13	6673	0.0021	51.6	340.3	237.9
14	6837	0.0019	91.9	59.6	251.1
15	6975	0.0032	97.7	175.0	158.0
16	6784	0.0008	51.6	35.9	237.1
17	42133	0.0010	0.0	96.4	95.9
18	28820	0.0089	54.9	4.6	156.8

The optimization selects a set of launches, assigns satellites to each selected launch, and sets a reconfiguration for each launch by setting the change in orbital elements. The satellites' orbital elements will be the orbital elements of the launch plus the change in orbital elements. The satellites will be evenly distributed in true anomaly around the orbit. The transfer is considered feasible if the final orbit can be reached from the initial orbit, as described in the previous section, within ten orbits using the available fuel and thrust. A segment of the genome produced by the genetic algorithm would have the form $\mathbf{x}_i$ = [launch assignment, number of satellites, Δa, Δe, Δi, $\Delta\omega$, $\Delta\Omega$, $\Delta\nu$]. The genome is permitted to have between one and twenty segments. It is possible that multiple planes of satellites may be deployed from a single launch by assigning multiple orbital element changes to one launch. In a simulation of a real-life scenario, it would be beneficial to introduce a constraint ensuring that the mass capacity for rideshare of a vehicle is not exceeded. The total number of satellites launched is not to exceed fifty.

Because the goal of the scenario is to maximize coverage over the state of California, a set of points evenly spaced with 100 miles between them was generated within the state. A minimum elevation limit for access of five degrees was imposed. A maximum ground sample distance (GSD) of 25m is also required for access. The satellites are assumed to be able to slew sufficiently to cover the area of interest, so no constraint is imposed on the off-boresight angle of the satellite. Some of the launch opportunities, such as launch 17, are incompatible with the mission requirements due to their high altitude, which would result in a GSD that violated the access requirements. These launches are still included in the optimization to test the algorithm's ability to avoid infeasible solutions.

The quality of access provided by a given solution is evaluated using two metrics: the average time average gap (TAG) of the ground points and the maximum revisit time over all points. The TAG of a ground point is defined as

$$\bar{G} = \frac{\sum^{gaps}(Gap\ Duration)^2}{Coverage\ Interval} \tag{21}$$

TAG provides the average time until next coverage for a given ground point when starting from an arbitrary time in the scenario. The maximum revisit time over all points calculates the longest time that each point is without coverage, then takes the largest of these values. The total number of satellites is minimized in order to survey the entire solution space and to determine the coverage possible at varying asset levels.

Due to the relatively high failure rate of nanosatellites, it is necessary to consider the possibility that some of the satellites may fail prematurely. The impact of this possibility is measured by determining the worst-case maximum revisit time over all points when twenty percent of the satellites are removed from the scenario. The linear programming approach discussed in the previous section is applied in order to get this worst-case objective value. The optimization problem therefore has four objectives: minimize average TAG, minimize maximum revisit time over all points, minimize degraded maximum revisit time over all points, and minimize number of satellites.

A scenario time of ninety days is used when calculating the nominal objectives. The degraded analysis uses a ten day scenario time to limit the size of the linear programming problem. The simulation is run until ten successive generations produce no improvement in the archive. A new population is then generated using the archive and randomly generated members [16]. An initial population of 200 candidates is used, with the population being scaled each run to be four times the archive size. This process is repeated for ten runs. For comparison purposes, optimization is performed on a Walker delta constellation with up to fifty satellites and up to twenty planes. The Walker formulation does not undergo reconfiguration. It seeks to minimize the total number of planes in addition to the objectives stated for the rideshare scenario.

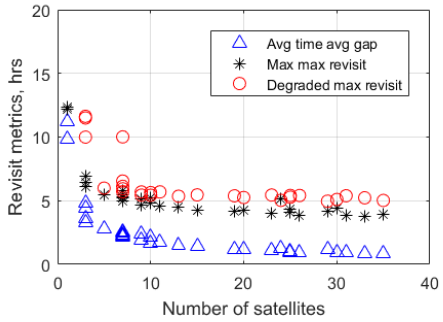

Fig. 2 Pareto frontier for rideshare-launched constellation.

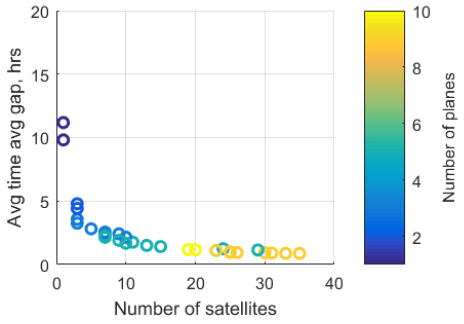

Fig. 3 Average time average gap for rideshare-launched constellation.

V. Results

The rideshare simulation produced a Pareto frontier with 31 results. The Pareto frontier is shown in Fig. 2. The TAG of the Pareto-optimal solutions ranges from 52 minutes for the larger constellations to 11.2 hours for a single satellite. The maximum revisit time over all points takes values between 3.8 and 12.4 hours. The degraded maximum revisit time over all points takes values between 5.0 and 11.6 hours, discounting the single-satellite case. Note the diminishing returns gained by adding satellites beyond the tenth. Indeed, the objective values change very slightly between 20 and 35 satellites. Fig. 3 shows the number of launches used by each solution. The theoretical FireSat-II example requires a revisit time of eight hours to identify nascent forest fires[7]. The imaging capability provided by the rideshare constellations is sufficient for detection on such a timeline. The rideshare constellation performance is inferior to proposed constellation designs such as the FUEGO program, which achieves 25 minute revisit times using dedicated launches[29].

An example rideshare constellation using 30 satellites is shown in Fig. 4. The constellation consists of two sets of near-polar orbits spaced about ninety degrees apart in RAAN, plus a pair of orbits near 50 degrees in inclination. During the ninety day simulation time, the polar orbits maintain similar relative positions, but the

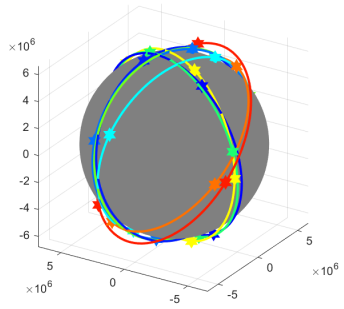

Fig. 4 Pareto-optimal constellation of 30 satellites

relative position of the 50 degree orbit with respect to the polar orbits varies.

The stagnation of the values with increasing numbers of satellites highlights the critical flaw in rideshare constellations. Because the initial launch values are fixed in such a way that may not be beneficial to the rideshare mission, the resulting constellation can have large gaps in coverage when the rideshare orbits do not overlap in a fortuitous manner. The ability to maneuver the satellites helps to mitigate the problem, but the high Δv cost to enact a change in orbital plane impedes the constellation's ability to achieve the uniform formation often used in constellation design.

There are two ways to increase the performance of the rideshare constellation. The first is to have a greater number of rideshare opportunities available. This simulation used only one month's worth of launches. By permitting the satellites to be launched over a longer time period, more rideshares become available, increasing available orbit diversity. However, spreading the launch of the constellation over a longer period of time decreases the overall life of the constellation, since the time from when to constellation is fully population to when the first satellite reaches the end of its life is decreased. The other method for increasing constellation performance and spacing between orbits is to increase the maneuvering capability of the satellites. Manuevering can be improved by either increasing the amount of fuel onboard the satellites or by using a low-thrust, high I_{sp} electric propulsion system. The latter case increases the overall Δv, but requires more time to get the constellation to its final configuration.

Compare the performance of the rideshare constellations to the Walker constellations optimized using the genetic algorithm. The Pareto frontier for the Walker case is shown in Fig. 5. Clearly, the Walker constellations offer superior performance over the rideshare constellations. A Walker constellation of four satellites has comparable performance to a rideshare constellation of 15-20 satellites. Furthermore, satellites can be added to the

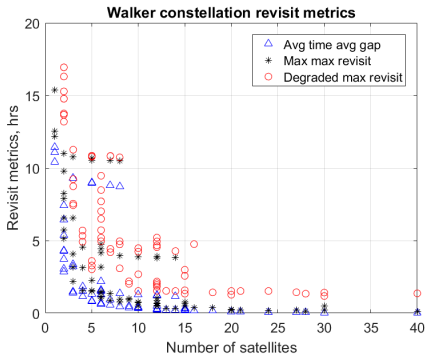

Fig. 5 Pareto frontier for Walker constellation.

Walker constellation to improve coverage until continuous coverage is reached, whereas the rideshare constellation has unfillable gaps due to the relative placement of the rideshare orbits. However, the cost of launching four satellites on dedicated rides is likely greater than the cost of the additional satellites needed for the rideshare constellation. The cost for the components listed in the previous section is $283K for everything except the propulsion system, which is still in development and does not have a published price. If the total cost is approximately $400K with the propulsion system, the satellite cost would be $6M for the rideshare constellation and $1.6M for the Walker constellation. Neither price includes the cost of testing or software development. The Walker constellation would require two to four launches to LEO, a cost of $36.8M-73.6M using Pegasus XL rockets[7]. Conversely, with a $30K per kilogram rideshare launch cost and a spacecraft weighing about 10kg, the rideshare launch cost is only $4.5M. Therefore, if the performance limitations of the rideshare constellation are acceptable, a constellation can be developed for about a quarter of the cost of a traditional Walker constellation. Fires in the state of California cause billions of dollars in damage each year, so the low cost of a fire detection constellation has the potential to pay for itself many times over.

VI. Conclusions

This paper outlines new methodologies for reachability and resilience analyses for constellations of nanosatellites. These methods leverage linear programming techniques and offer savings in computation time over other methods. It also analyzes the ability of a constellation built using only rideshare opportunities to provide coverage over California to perform fire detection. An average time average gap of less than one hour is achievable, as is a maximum revisit time over all points of less than four hours. The rideshare performance is compared to the performance of a Walker constellation. Although the Walker constellation can achieve arbitrary levels of coverage through the addition of further satellites, rideshare constellations are capable of meeting the capabilities of small Walker constellations at greatly reduced cost.

Acknowledgments

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1651272. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Support for this work was also provided by the Virginia Tech Institute for Critical Technology and Applied Science, the Ted and Karyn Hume Center for National Security and Technology, and the Virginia Space Grant Consortium.

References

[1] Mott, K. E., and Black, J. T., "Design of a resilient rideshare-based small satellite constellation using a genetic algorithm," Advances in the Astronautical Sciences AAS/AIAA Space Flight Mechanics Meeting 2019, Univelt, 2019. Pre publication.

[2] Bedingfield, K. L., Leach, R. D., and Alexander, M. B., "Spacecraft System Failures and Anomalies Attributed to the Natural Space Environment," Tech. Rep. 1390, NASA, 1996.

[3] Venturini, C. C., "Improving Mission Success of CubeSats," Tech. Rep. TOR-2017-01689, The Aerospace Corporation, 2017.

[4] Peng, Z., and Kohani, S., "The performance of the constellations satellites based on reliability," Journal of Space Safety Engineering, Vol. 4, No. 2, 2017, pp. 112–116. doi:10.1016/j.jsse.2017.07.003.

[5] Hastings, D. E., and Tour, P. A. L., "An economic analysis of disaggregation of space assets: Application to GPS," Acta Astronautica, Vol. 134, 2017, pp. 244–264. doi: 10.1016/j.actaastro.2017.02.008.

[6] Stenger, D. K., "Survivability Analysis of The Iridium Low Earth Orbit Satellite Network," Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, 1996.

[7] Wertz, J., Everett, D., and Puschell, J., Space Mission Engineering: The New SMAD, 1st ed., Space technology library, Microcosm Press, Hawthorne, CA, 2011, pp. 298,299,316,486.

[8] Berk, J., Straub, J., and Whalen, D., "The open prototype for educational NanoSats: Fixing the other side of the small satellite cost equation," 2013 IEEE Aerospace Conference, 2013, pp. 1–16. doi:10.1109/AERO.2013.6497393.

[9] Larson, W. J., and Wertz, J. R. (eds.), Space Mission Analysis and Design, 3rd ed., Space technology library, Microcosm Press and Kluwer Academic Publishers, El Segundo, CA, 1999, Chap. Communications Architecture, pp. 533–586.

[10] NASA, "ELaNa XII CubeSat Launch on NROL-55 Mission," Tech. rep., NASA, 2015.

[11] Ellis, A., Mercury, M., and Brown, S., "Global coverage from ad-hoc constellations in rideshare orbits," Proceedings of the 26th Annual AIAA/USU Conference on Small Satellites, The American Institute of Aeronautics and Astronautics, Logan, UT, 2012.

[12] Gangestad, J. W., Wilson, J. R., Gates, K. L., and Langer, J. V., "Rideshare-initiated constellations: Future CubeSat architectures with the current launch manifest," Proceedings of the 31st Space Symposium, Space Foundation, Space Foundation, Colorado Springs, CO, 2015.

[13] Mott, K. E., and Black, J. T., "Model-based heterogeneous optimal space constellation design," 2018 21st International Conference on Information Fusion (FUSION), Institute of Electrical and Electronics Engineers, 2018, pp. 602–609. doi:10.23919/ICIF.2018.8455222.

[14] Mott, K. E., and Black, J. T., "Heterogeneous Constellation Design Methodology Applied to a Mars-orbiting Communications and Positioning Constellation," AAS/AIAA Astrodynamics Conference, 2017, Advances in the Astronautical Sciences Series, Vol. 162, Univelt, Stevenson, WA, 2017, pp. 2383–2396.

[15] Ferringer, M. P., Spencer, D. B., and Reed, P., "Many-objective reconfiguration of operational satellite constellations with the Large-Cluster Epsilon Non-dominated Sorting Genetic Algorithm-II," 2009 IEEE Congress on Evolutionary Computation, Institute of Electrical and Electronics Engineers, Trondheim, Norway, 2009, pp. 340–349. doi:10.1109/CEC.2009.4982967.

[16] Hadka, D., and Reed, P., "Borg: An Auto-Adaptive Many-Objective Evolutionary Computing Framework," Evolutionary Computing, Vol. 21, No. 2, 2013, pp. 231–259. doi:10.1162/EVCO_a_00075.

[17] Legge, R. S., Jr, and Miller, D. W., "Optimization and Valuation of Reconfigurable Satellite Constellations Under Uncertainty," Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 2014.

[18] Appel, L., Guelman, M., and Mishne, D., "Optimization of satellite constellation reconfiguration maneuvers," Acta Astronautica, Vol. 99, 2014, pp. 166–174. doi:10.1016/j.actaastro.2014.02.016.

[19] Fakoor, M., Bakhtiari, M., and Soleymani, M., "Optimal design of the satellite constellation arrangement reconfiguration process," Advances in Space Research, Vol. 58, No. 3, 2016, pp. 372–386. doi:10.1016/j.asr.2016.04.031.

[20] Xue, D., Li, J., Baoyin, H., and Jiang, F., "Reachable Domain for Spacecraft with a Single Impulse," Journal of Guidance, Control, and Dynamics, Vol. 33, No. 3, 2010, pp. 934–942. doi:10.2514/1.43963.

[21] Holzinger, M. J., Scheeres, D. J., and Erwin, R. S., "On-Orbit Operational Range Computation Using Gauss's Variational Equations with J2 Perturbations," Journal of Guidance, Control, and Dynamics, Vol. 37, No. 2, 2014, pp. 608–622. doi:10.2514/1.53861.

[22] HomChaudhuri, B., Oishi, M., Shubert, M., Baldwin, M., and Erwin, R. S., "Computing Reach-Avoid Sets for Space Vehicle Docking Under Continuous Thrust," 2016 IEEE 55th Conference on Decision and Control, Institute of Electrical and Electronics Engineers, Las Vegas, NV, 2016, pp. 3312–3318. doi:10.1109/CDC.2016.7798767.

[23] Zagaris, C., and Romano, M., "Applied Reachability Analysis for Spacecraft Rendezvous and Docking with a Tumbling Object," 2018 Space Flight Mechanics Meeting, The American Institute of Aeronautics and Astronautics, Kissimmee, FL, 2018. doi:10.2514/6.2018-2220.

[24] Breger, L., and How, J. P., "GVE-Based Dynamics and Control for Formation Flying Spacecraft," Proceedings of the 2nd International Symposium on Formation Flying Missions & Technologies, NASA, Washington, D.C., 2004.

[25] Walker, M. J. H., Ireland, B., and Owens, J., "A set of modified equinoctial orbital elements," Celestial Mechanics, Vol. 36, No. 4, 1985, pp. 409–419. doi:10.1007/BF01227493, errata:[30].

[26] Weisel, W. E., Spaceflight Dynamics, 3rd ed., CreateSpace, Scotts Valley, CA, 2010.

[27] Vallado, D. A., Fundamentals of Astrodynamics and Applications, 4th ed., Microcosm Press, Hawthorne, CA, 2013, Chap. General Perturbation Techniques, pp. 609–654.

[28] Alfano, S., "Rapid Generation of Site/Satellite Timetables," Journal of Spacecraft and Rockets, Vol. 30, No. 6, 1993, pp. 760–764. doi:10.2514/3.26383.

[29] Escorial, D., Tourne, I. F., Reina, F. J., and Gonzalo, J., "FUEGO: A dedicated constellation of small satellites to detect and monitor forest fires," Acta Astronautica, Vol. 53, No. 9-12, 2003, pp. 765–775. doi:10.1016/S0094-5765(03)00052-3.

[30] Walker, M. J. H., "Errata, A set of modified equinoctial orbital elements," Celestial Mechanics, Vol. 38, No. 4, 1986, pp. 391–392. doi:10.1007/BF01238929.