Cost Effective Persistent Regional Surveillance with Reconfigurable Satellite Constellations

Robert S. Legge Jr.* and David W. Miller†

Space-based persistent surveillance provides decision makers with information necessary to effectively respond to both natural and man-made crises. This paper investigates a *reconfigurable* constellation strategy that utilizes on-demand, maneuverable satellites to provide on-demand focused regional coverage with short revisit times at greatly decreased cost when compared to traditional static satellite constellations. A general framework is introduced to guide the design and optimization of reconfigurable satellite constellations specifically tailored to stakeholder objectives while considering requirement uncertainty. The framework consists of three elements: a detailed simulation model to compute constellation performance and cost, Monte Carlo simulation, and a parallel multi-objective evolutionary algorithm developed from the ϵ-NSGA-II genetic algorithm. Additionally, a new persistence metric is developed to directly measure how well a design meets desired temporal and spatial sampling requirements and a decision model and optimal assignment process is developed to determine how to employ the option of reconfigurability to respond to specific regional events. 8 optimization runs were performed on a 1024 processor computing cluster to compare the cost-effectiveness of several constellation architectures over varied coverage requirements. Results show that reconfigurable constellations cost 20 to 70% less than similarly performing static constellations for the scenarios studied, and this cost savings grows with increasingly demanding coverage requirements.

I. Introduction

Space-based surveillance systems provide imagery and other data products to support disaster response. While traditional space-based surveillance systems emphasize generating high spatial resolution imagery, effective disaster response often also requires good temporal resolution where observation frequency is matched to event dynamics. Ideally, these persistent systems would feature sufficient spatial and temporal resolution to support damage assessment, search and rescue, and long-term recovery planning. Ground-based, air-based and space-based systems can all provide effective disaster response, but all suffer from limitations. Ground-based systems including fixed sensors and mobile response teams may become unavailable due to infrastructure damage. Air-based systems including fixed wing aircraft, helicopters and unmanned aerial vehicles require significant capital investment to provide for air-basing and personnel needs which often slows deployment and limits responsiveness. Traditional space-based constellations provide wide-area coverage without the limitations of ground and air-based systems, but require a large number of satellites to meet persistence goals.

*Technical Staff, MIT Lincoln Laboratory
†Professor, Department of Aeronautics and Astronautics, 37-327, Senior Member

New thinking is needed to develop cost-effective persistent surveillance satellite constellations. In this paper we introduce and explore a reconfigurable satellite constellation concept. The concept employs a dynamic constellation comprised of maneuverable satellites to dramatically increase satellite utilization by focusing coverage on specific regions on Earth in times of need. We also develop methodology that guides the design and optimization of cost-effective reconfigurable satellite constellations while accounting for uncertainty in future operating context. The methodology employs detailed simulation models and multi-objective evolutionary algorithms (MOEAs) to find the set of efficient designs that comprise the optimal tradeoff between maximizing system performance and minimizing cost. We also introduce a new performance figure of merit that measures how well a system meets temporal and spatial resolution goals. We utilize this methodology to calculate the cost savings of implementing a reconfigurable architecture versus a similarly performing traditional static architecture. This cost savings, called the value of reconfigurability, is the amount saved by implementing the reconfigurable approach.

This paper introduces the reconfigurable constellation concept in Section II and details the constellation design and optimization methodology in Section III. Sections IV through VII describe the three main elements of the methodology including: the multidisciplinary simulation model, Monte Carlo sampling, and multi-objective evolutionary optimization. Section VIII presents a cost-effectiveness comparison between reconfigurable and static constellations for different temporal and spatial resolution requirements. Conclusions and recommendations for future research are then given Section IX.

II. Reconfigurable Constellations

This paper investigates a new flexible constellation architecture that gives operators the ability to focus satellite coverage on different areas of the Earth in times of need. The reconfigurable constellation concept (ReCon), previously introduced by Paek et al. [1] and further elaborated by this author [2], features two operational modes: global observation mode (GOM) and regional observation mode (ROM). GOM features a non-repeating ground track that allows the satellites to provide coverage within a latitude band equal to the orbit inclination, and is similar to traditional static constellations. ROM features repeating ground track (RGT) orbits where the Earth nodal day and the satellite's orbital period are synchronized so that the ground paths repeat and the satellites provide enhanced regional coverage. The ReCon constellation normally resides in GOM providing partial global coverage. When a disaster event requiring additional coverage occurs, a subset of the constellation would maneuver, via an altitude change and proper phasing, into ROM to meet a desired level of persistence. This reconfigurable architecture trades an increase in system complexity and operational burden associated with greater satellite maneuverability for greater satellite utilization due to the ability to focus coverage. This increase in coverage leads to fewer required satellites in the constellation to meet a specified level of persistence. We hypothesize that the benefits provided by reconfigurability will significantly outweigh any costs associated with increased maneuverability and increased operational complexity.

III. Reconfigurable Constellation Design and Optimization Framework

A constellation design and optimization framework was developed to provide a process to design and optimize reconfigurable satellite constellations that perform well under uncertain operating conditions. The framework is comprised of three nested layers shown in Figure 1. The innermost layer, the simulation layer, contains a detailed, multi-disciplinary simulation model that computes the performance and cost of specific satellite constellation designs. The middle layer, the Monte Carlo sampling layer, maps uncertain disaster event location and timing distributions into a distribution of constellation performance. The outermost layer, the multi-objective optimization layer, utilizes state of the art multi-objective evolutionary optimization algorithms to find the set of efficient designs that simultaneously maximize expected performance and minimize cost. Sections IV and V explain the simulation model in more detail while Sections VI and VII provide details on the Monte Carlo sampling and multi-objective optimization layers.

In addition to finding efficient designs, the framework also allows for a direct comparison of reconfigurable and static architectures in terms of cost-effectiveness. Since static constellations are unable to reconfigure, they must provide all regional event coverage from GOM. When the framework is used to optimize both reconfigurable and static architectures for the same scenario, the resulting sets of efficient designs can be directly compared to calculate the value of reconfigurability. The value of reconfigurability is defined as the reduction in cost of a reconfigurable design when compared to a similarly performing static design. This concept is similar to the 'value of flexibility' metric which measures the benefit of incorporating flexibility into systems [3–5].

A. Persistence Figure of Merit

This paper investigates satellite constellations that provide persistent coverage, where persistence is defined as long-term, continuing coverage with a specified frequency of observation. Therefore, the performance figure of merit must capture how well the constellation coverage matches the desired persistence. If the coverage under-samples with respect to the desired persistence, then some temporal event dynamics will be missed, and overall performance should decrease. If the coverage over-samples with respect to the desired persistence, then all the temporal event dynamics will be captured, but little or no additional utility may be gained for oversampling and system cost will likely increase. In this case the performance should not increase beyond ideal sampling assuming that all the desired information is gathered during each observation. Additionally, the performance metric must also account for several other factors that influence optical imaging quality including spatial resolution and solar illumination. This section first motivates the need for a new performance metric tied directly to persistence requirements and then introduces the newly developed persistence metric.

Statistical metrics like average and maximum revisit time are often used to evaluate the performance of satellite constellations, however, these metrics exhibit several undesirable traits that make them poor measures of persistence. Previous literature [6–8] has shown that the two objectives of minimizing average revisit time and minimizing maximum revisit time are often in tension, and that improving one often degrades the other. This basic tension illustrates

how these statistical measures fail in measuring persistence. Minimizing average revisit time involves simply increasing the number of observations in a fixed time window regardless of when they occur. Minimizing maximum revisit time simply attempts to reduce the longest gap in coverage with no consideration of other observations. Recently, response time has been proposed as a better measure of persistence **[Cite Roger Thompson Work]**. This metric provides significant improvements over average and maximum revisit time but it is still statistical in nature and therefore cannot easily incorporate additional factors like spatial resolution. Instead, we propose a new metric that takes a micro-scale approach to measuring persistence and can easily incorporate spatial resolution and illumination constraints. This new persistence metric is comprised of two utility functions that measure how well the desired temporal and spatial resolution are met. Most literature on satellite constellation design does not account for spatial resolution, other than imposing fixed requirements. The ReCon concept, however, features satellites at different altitudes (caused by the GOM and ROM modes of operation) and also satellites in ROM that ensure nadir passes over the disaster event. Therefore, it is important to include the effect of spatial resolution on system performance. The persistence utility function is written as a combination of these two effects:

$$U = (U_\tau + \Delta U) \times U_{GSD} \tag{1}$$

Where U_τ is the temporal utility term, U_{GSD} is the spatial utility term, and ΔU is a correction term to remove the order dependance of sequential observations with different spatial resolution. The rest of this section explains these three terms in greater detail.

The temporal utility term is based on the time elapsed since the last observation was made, and is described mathematically by:

$$U_\tau = \min\left(\left[\frac{\tau}{T}\right], 1\right) \tag{2}$$

Where T is the desired persistence and τ is the time since the last observation was made. This utility function, shown in Figure 2 (top), starts off at 0 when $\tau = 0$ and ramps up to a maximum value of 1 when τ is equal to the desired persistence T. Therefore, if the scene was recently sampled, the additional utility generated for another observation would be low. If $\tau \geq T$, then the utility for another observation is set to 1 regardless of how large τ is. While capping the maximum per-observation utility at 1 may seem counter-intuitive, there is an opportunity cost associated with undersampling. Given the overall objective of maximizing the total utility generated during each event response, any period where utility is not accumulating is lost opportunity in terms of maximizing total utility. Since τ is undefined for the first observation of the day, U_τ is set to 1.

Ground sample distance (GSD) is a measure of the spatial resolution (information content) contained in optical imagery. Similar to an event having a desired sampling rate, the event will also have a desired GSD driven by the desired end data product. Rather than enforcing strict GSD requirements, we introduce a GSD based utility function matched to the degradation in the optical information content as GSD increases past the desired GSD. This degradation

is matched to the change in the National Imagery Interpretability Rating Scale (NIIRS) as predicted in the General Image Quality Equation (GIQE) [9, 10]. The GIQE predicts that the change in NIIRS based on increasing GSD (assuming good sharpness) is given by [10]:

$$\Delta_{\text{NIIRS}} = -3.32 log_{10}\left[\frac{x}{X}\right] \tag{3}$$

Where X is the desired GSD and x is the actual GSD of an image. The GSD utility function is then given as a function of Δ_{NIIRS} as follows:

$$U_{GSD} = \min\left(\max\left(1 - \frac{\Delta_{\text{NIIRS}}}{\delta}, 0\right), 1\right) \tag{4}$$

Where δ is a scaling parameter that maps the drop in U_{GSD} to the drop in Δ_{NIIRS}, so that $U_{GSD} = 0$ when $x \geq 2^{\delta}X$. We use $\delta = 2$ in this analysis, which means that U_{GSD} is zero when the actual GSD is four times greater than the desired GSD. Figure 2 (bottom) shows the GSD utility curve as a function of x/X. The plot shows that $U_{GSD} = 1$ when $x \leq X$ and $U_{GSD} = 0$ when $x \geq 4X$. When $X \leq x \geq 4X$, the utility is governed by the scaled NIIRS relation. This spatial resolution utility function also exhibits similar behavior to the temporal utility function in that no additional utility is generated by spatial over-sampling.

The overall performance for a specific event is then defined as the cumulative sum of all utility generated during the specified event duration, and, since we are interested in optical imagery in this paper, only observations made during daytime generate utility. Future work could easily modify the persistence metric to vary the utility of optical imagery as a function of solar beta angle as shown in [2]. Performance is defined as the sum of the persistence utility generated by all observations provided by the constellation during each event response. The total performance for an event response P_E is written mathematically as the sum of the persistence utility generated over all observations during each day and then summed over all the days during the event response:

$$P_E = \sum_{days}\left(1 + \sum_{i=2}^{N_{obs}} U\left(\tau_i, x_i\right)\right) \tag{5}$$

The correction term ΔU, previously introduced in Equation 1, fixes the problem where the order of two observations with different GSD affects total utility. Consider the situation where observation O1 has $1.8m$ GSD while observation O2 has $1.0m$ GSD. If these two observations are separated by $0.2hr$ (and $T = 1hr$, $X = 1m$), then the utility for the first observation is $U_1 = 0.58$ and the utility for the second observation is $U_2 = 0.2$, and then the total utility generated is 0.78. If the order of the observations was reversed, the breakdown would be: $U_1 = 1.0$, $U_2 = 0.12$, and the total utility is 1.12. This difference is caused by the utility function not properly accounting for the situation where the second observation has low temporal utility but provides better GSD. This is fixed by introducing the following utility correction factor that increases the utility of a second observation with better GSD to ensure that the order of observations does not affect total utility. The correction factor is defined as 0 for the first daily observation and for the

remaining ith daily observations is:

$$\Delta U_i = \max[0,\ U(\tau_{i-1}, x_i, \Delta U_{i-1}) - U(\tau_{i-1}, x_{i-1}, \Delta U_{i-1})] \times \left(\frac{1 - U_{\tau,i}}{U_{GSD,i}}\right) \tag{6}$$

IV. Simulation Model

The simulation model calculates the performance and cost of constellation designs for possible future scenarios. Inputs to the simulation model include: which architecture to model (static or reconfigurable); a set of design variables specifying the constellation pattern; a list of parameters specifying fixed system quantities; and a list of targets (location and timing) representing all regional disaster events that the system must respond to over its lifetime. The simulation model is comprised of five modules as shown in Figure 3. The simulation setup module generates the initial constellation pattern and the satellite module sizes the optical payload, satellite bus, and propulsion system. The cost module computes the constellation cost by aggregating the cost of the optical payload, the satellite bus, and launch and then applies quantities of scale effects. The astrodynamics module employs a campaign-based simulation to track the state of the constellation throughout its lifetime. For reconfigurable systems, the simulation model determines how to respond to each event. The simulation model then tracks the coverage provided by the constellation for each event and tracks the depletion of individual satellite propellant over time. The final module is the performance module which computes the overall life-cycle performance of the system defined as the mean event performance generated during all regional event responses. This performance model is tailored to rate how well the system provided the desired level of temporal and spatial resolution coverage for each regional event. The remainder of this section summarizes the important modeling processes and a more detailed explanation can be found in [2].

A. Constellation Pattern

GOM for both static and reconfigurable designs consists of satellites in a Walker Delta pattern. The Walker Delta pattern features satellites in circular orbits with common semi-major axis and inclination that are divided into N_p equally spaced orbital planes, each containing N_{sp} satellites. A third parameter, F, controls the anomaly phasing of satellites in adjacent orbital planes. Satellites in adjacent orbital planes are shifted in M by $F \times PU$, where $F = 0, \cdots, \left(N_p - 1\right)$ and the pattern unit (PU) is given by $PU = 2\pi/\left(N_p N_{sp}\right)$ [11, 12]. Therefore, five design variables specify the pattern: the altitude, inclination, N_p, N_{sp}, and F.

B. Satellite Properties

Values for satellite dry mass and stowed volume during launch are needed to estimate satellite bus cost and launch cost. Rather than constructing detailed models to estimate satellite mass and volume, we decided use analogous estimation based on the properties of ten recently launched optical Earth observation satellites (shown in Table 1). This approach ensures that the estimated satellite footprint is grounded in reality. Figure 4 shows the curve fits for satellite dry mass

and satellite stowed volume as a function of optical aperture diameter.

Propulsive capability is required for initial constellation deployment, station-keeping, aerodynamic drag makeup, performing reconfiguration maneuvers, and disposal through de-orbiting at end-of-life. The propulsion system is sized based on the total ΔV required (ΔV_T) and M_d. Propulsive maneuvers during initial deployment correct for launch vehicle injection errors and properly phase the satellites in the constellation pattern. Phasing relates to the situation where several satellites are launched together that will ultimately reside in different orbital slots in the constellation pattern. These satellites will either need to spread out in M if they will reside in the same orbit plane, or Ω if they will reside in different orbit planes. Changing M separation is trivial, since a small change in altitude between two satellites will cause a difference in mean motion, and, therefore, M separation over time. Changing Ω is much harder and can be accomplished via costly propulsive maneuvers or natural differences in orbital precession. Differential orbital precession, caused primarily by the Earth J_2 gravity variation, can be harnessed to slowly change the relative Ω of satellites over time. For deployment, the satellites are launched to an altitude lower than the intended GOM altitude to allow for phasing. The satellites then use differential orbital precession to achieve their final GOM orbit planes in a limited deployment period, and the amount of ΔV required to raise the altitude to GOM (ΔV_{dep}) is then determined. The launch vehicle injection error ΔV (ΔV_{LV}) was set to $22 m/s$ which is able to correct 3σ error for contemporary launch vehicles [2], and the station-keeping ΔV (ΔV_{SK}) is set to $10 m/s$ per year [13] to maintain absolute station-keeping in the constellation pattern. Aerodynamic drag makeup ΔV (ΔV_{drag}) is estimated using a parametric atmospheric density relation as a function of altitude for solar mean [12] and a constant ballistic coefficient of $75 kg/m^3$. De-orbit ΔV ($\Delta V_{deorbit}$) was calculated to lower perigee to $75 km$ to ensure quick end-of-life disposal. The total ΔV needed by each satellite is then:

$$\Delta V_T = \Delta V_{dep} + \Delta V_{LV} + \Delta V_{SK} + \Delta V_{drag} + \Delta V_{deorbit} + \Delta V_{recon} \tag{7}$$

Where ΔV_{recon} is the amount of ΔV allotted for reconfiguration maneuvers and is zero for static designs. The propellant mass and total propulsion system mass are then calculated from ΔV_T and M_d. Since responsiveness is important in the disaster response scenario studied in this paper, we only consider high thrust chemical monopropellant systems able to perform fast reconfiguration maneuvers. However, efficient electric propulsion could potentially be used for deployment, de-orbit and to return the satellites to GOM after an event response, and should be considered in future work. Two parameters describe the propulsion system: the specific impulse I_{SP} measures propellant efficiency and dry mass fraction ξ_p is defined as the propulsion system dry mass as a fraction of the total propulsion system wet mass. ξ_p is set to 0.2 assuming that propulsion system dry mass is roughly 20% of the total propulsion system wet mass. I_{SP} is set to $240 s$ as a conservative estimate of the performance of new 'green' monopropellant technology [14, 15].

The propellant mass is then given by the rocket equation as:

$$M_{prop} = -M_d \left(\frac{\zeta - 1}{\xi_p (\zeta - 1) - 1} \right) \tag{8}$$

Where $\zeta = \exp(\Delta V_T / g I_{SP})$. The total propulsion system mass is:

$$M_P = \frac{M_{prop}}{\left(1 - \xi_p\right)} \tag{9}$$

And the total satellite wet mass is:

$$M_w = M_d + M_P \tag{10}$$

C. Cost

Modeled constellation cost is comprised of four elements: payload cost, satellite bus cost, launch cost, and economies of scale effects. The payload cost is, in turn, modeled as the cost of the optical telescope assembly (OTA) and the imaging sensor. OTA cost is parametrically modeled as a function of aperture diameter. Early work on developing parametric cost models for terrestrial telescopes found that the cost of such systems scaled primarily with aperture diameter raised to the powers of 2 - 3 [16]. However, recent research has shown that these models do not accurately predict the cost of space telescopes [16–20], and recently developed models indicate that space OTA cost scales with aperture diameter raised to the power of 1.4 to 2.0 [20]. We have used a power of 1.6 in this paper. Sensor cost was modeled from the 'Detectors Subsystem Cost for CCD Detectors' cost estimating relationship (CER) contained in the NASA Instrument Cost Model (NICM) [21], which gives the sensor cost as a function of mass. Absent available FPA mass data for existing Earth observation satellites, M_{FPA} is estimated by scaling the mass of the Kodak Model 1000 Camera System by aperture diameter [12]. This instrument was used on the IKONOS satellite and had an aperture size of $0.7m$ and a FPA mass of $16kg$ [22]. We estimate that M_{FPA} will scale with the cube of the scaling ratio, which is a ratio of the aperture diameter to $0.7m$. The scaling ratio is given as $R = \frac{D}{0.7m}$ and then the FPA mass is given as $M_{FPA} = 16kg \times R^3$. Combining the OTA and sensor cost yields the total payload cost C_{pay}.

$$C_{pay} = 38000 D^{1.6} + 60615 D^{2.67} \quad (\$k\ FY2010) \tag{11}$$

Satellite cost is modeled by blending the output from the Small Satellite Cost Model (SSCM) focused on smaller satellites and the Unmanned Spacecraft Cost Model 8 (USCM8) focused on larger satellites. Both models estimate cost based on the mass of the satellite, and we use the dry mass of the satellite M_d for the cost estimate. The USCM8 model estimates both non-recurring engineering (NRE) cost and recurring engineering (RE) cost and takes in the payload cost broken down into NRE and RE. Absent a detailed payload cost breakdown, we have assumed that the

payload RE cost is 40% of the total cost predicted by Equation 11. The SSCM also predicts a combined NRE plus RE cost, and we again assume that the satellite RE cost is 40% of the total cost predicted by the SSCM. The SSCM was used for $M_d \leq 400kg$ and the USCM8 cost model was used for $M_d \geq 200kg$, and linear blending was used to smooth out the transition between models from $200kg$ to $400kg$. Figure 5 illustrates example output of the satellite cost model as a function of aperture diameter. The top plot shows the USCM8 (solid) and SSCM (dashed) estimated NRE and RE cost as a function of aperture diameter while holding all other satellite variables constant. Also shown in the plot is the region where the models are blended, corresponding to spacecraft dry mass between 200 and $400kg$. The plot shows that this is the region where both models show the most agreement and therefore the blended curves (in the bottom plot) are fairly smooth. Additionally, a learning curve is applied to the RE cost to reflect the cost savings associated with producing multiple identical satellites. While there are limited examples of large scale satellite production, a learning curve of 90% has been proposed as a good rule of thumb for complex aerospace systems [23] and is used in this paper. This means that the second unit will cost 90% of the cost of the first unit, and the fourth unit will cost 90% the cost of the second unit and so on.

Launch cost is determined by a detailed launch vehicle assignment optimization which captures the effect on launch cost of satellite wet mass (M_w), satellite stowed volume ($\rho_{sc} \times M_w$), orbital inclination, GOM altitude, and the spread of GOM orbital planes. The deployment strategy is as follows. The satellites are launched to an altitude less than their intended GOM altitude to allow for phasing caused by differential orbital precession. This altitude difference will affect both the amount that the satellite can drift in Ω during a fixed deployment time and also the amount of ΔV that each satellite will expend maneuvering from the deployment altitude to the final GOM altitude. Increasing the deployment altitude difference can potentially lower launch cost by allowing satellites with different orbit planes in GOM to share the same launch, but will increase satellite cost by increasing maneuverability requirements. Therefore, the launch altitude is optimized to minimize total constellation cost in the simulation model. Five U.S. launch vehicles, shown in Table 2, that span a wide range of capability are included in the model. The reduction in payload mass capability of each launch vehicle caused by increased launch altitude and increased inclination is modeled using published response surfaces [24]. Several modifications to these response surfaces were made to enable estimation over the full range of inclinations and launch vehicles considered in this paper. Retrograde inclinations past the range given by Lafleur et al. were extrapolated and the Athena Ic was modeled with the response surface given for the Athena I with a scaled payload mass to reflect increased capability. The Falcon Heavy was modeled with the Delta IV Heavy response surface, also with a scaled payload mass to reflect it's increased capability. An assignment optimization is then conducted to determine the lowest cost suite of launch vehicles that can launch all the satellites subject to launch vehicle payload mass and volume constraints and a maximum deployment time of 3 months. An unlimited supply of launch vehicles was assumed and no restrictions were placed on launch azimuth.

D. Campaign Based Simulation to Model Event Response

A campaign-based simulation models the system response for a given set of regional targets distributed in space and in time throughout the system's lifetime. This campaign-based simulation process is shown in Figure 6. The constellation is initially deployed into GOM through the previously explained launch and deployment process. While in GOM, the simulation model propagates the constellation using a numerical Keplerian propagator with J_2 perturbations and accounts for station-keeping and drag makeup ΔV expenditure. When the first regional event occurs that requires a response, an assignment process determines the optimal combination of satellites to maneuver into ROM to best meet the desired coverage while consuming the least amount of ΔV. The assigned satellites are then maneuvered into ROM and access times and GSD are computed for coverage provided of the regional event location for the duration of the event. After the event response is finished, the satellites in ROM return to GOM to await the next response. Section V describes the assignment and reconfiguration strategy in more detail.

V. Regional Event Response

A. Reconfiguration Strategy

The goal of reconfiguration is to move satellites from GOM to a specific orbital slot in ROM that provides coverage to the disaster event location. The maneuver consists of an altitude change in addition to proper phasing to achieve the correct ROM orbit. The maneuver is modeled as a double Hohmann transfer which first places the satellite in a circular drift orbit with a greater altitude difference from ROM to speed phasing with the desired ROM orbital slot. All maneuvers are modeled as in-plane and impulsive. Propulsive plane changes are not used due to their immense ΔV cost.

Figure 7 shows the reconfiguration strategy in more detail. The satellite starts off (label A) with an initial orbital position described by Ω_0, M_0, and a_{GOM}. The satellite then performs a Hohmann transfer (label B) with initial propulsive maneuver ΔV_{T1a} and final propulsive maneuver ΔV_{T1b} to achieve a new semi-major axis of $a_D = a_{GOM} + \Delta h_D$. The transfer has a duration of t_{T1} during which time the orbital state changes by $\Delta\Omega_{T1}$ and ΔM_{T1} by J_2 induced orbital precession and mean motion, respectively. The satellite then waits in the drift orbit (label C) for duration t_D during which time the orbital state changes by $\Delta\Omega_D$ and ΔM_D. When proper orbital phasing occurs, the satellite then performs a second Hohmann transfer (label D) with initial propulsive maneuver ΔV_{T2a} and final propulsive maneuver ΔV_{T2b} to achieve the RGT orbit (label E) with semi-major axis a_{ROM}. The second transfer has a duration of t_{T2} during which time the orbital state changes by $\Delta\Omega_{T2}$ and ΔM_{T2}. The total reconfiguration time is $t_R = t_{T1} + t_D + t_{T2}$. The rest of this section explains how the drift time t_D is computed to ensure proper phasing to place the satellite into a ROM orbit that provides RGT coverage for the specific disaster event location.

Each RGT orbit can be described by the unique angle Λ that specifies the shift in longitude of the ground track which is a function of the satellite's orbital state and the RGT type (an integer number of orbits N_o repeating in an

integer number of days N_d) [25]:

$$\Lambda = N_o\Omega + N_d M \tag{12}$$

For any satellite in GOM, with a given Ω, this equation specifies the mean anomaly needed to place the satellite on the RGT orbit that passes over the event ground location. Additionally, Λ for a specific ground location changes over time due to the effects of mean motion and orbital precession as follows:

$$\Lambda_f = \Lambda_0 + \left[N_o\dot{\Omega}_{ROM} + N_d\left(n_{ROM} + \dot{M}_{ROM} + \dot{\omega}_{ROM}\right)\right]t_R \tag{13}$$

Where Λ_0 is the initial RGT angle, and Λ_f is the RGT angle at the end of the reconfiguration maneuver of duration t_R. Using Λ_f in Equation 12, and rearranging for M yields:

$$M_f = \frac{\left(\Lambda_f - N_o\Omega_f\right)}{N_d} \tag{14}$$

Where Ω_f is the satellite's Ω at time $t = t_R$. This relation gives the necessary M at the end of reconfiguration to achieve the correct ROM orbit including J_2 orbital precession effects. The satellite's final orbital state (Ω_f and M_f) is dictated by the reconfiguration maneuver strategy and is determined by the following relations:

$$\Omega_f = \Omega_0 + \Delta\Omega_{T1} + \Delta\Omega_D + \Delta\Omega_{T2} \tag{15}$$

$$M_f = M_0 + \Delta M_{T1} + \Delta M_D + \Delta M_{T2} \tag{16}$$

Where Ω_0 and M_0 describe the satellite's original state, $\Delta\Omega_{T1}$ and ΔM_{T1} is the change in orbital parameters during the first Hohmann transfer, $\Delta\Omega_D$ and ΔM_D are the change in orbital parameters during time spent in the drift orbit, and $\Delta\Omega_{T2}$ and ΔM_{T2} are the change in orbital parameters during the second Hohmann transfer. Equations 15 and 16 can then be expanded to:

$$\Omega_f = \Omega_0 + \dot{\Omega}_{T1}t_{T1} + \dot{\Omega}_D t_D + \dot{\Omega}_{T2}t_{T2} \tag{17}$$

$$\begin{aligned} M_f = M_0 &+ \left(n_{T1} + \dot{\omega}_{T1} + \dot{M}_{T1}\right)t_{T1} + \left(n_D + \dot{\omega}_D + \dot{M}_D\right)t_D \\ &+ \left(n_{T2} + \dot{\omega}_{T2} + \dot{M}_{T2}\right)t_{T2} \end{aligned} \tag{18}$$

Where t_{T1} and t_{T2} are the transfer times for Hohmann transfer 1 and 2 respectively which are equal to half the orbital

period of the eccentric transfer orbits. Relations for n, $\dot{\Omega}$, $\dot{\omega}$, and $\dot{M}$ are given as [12]:

$$\dot{\Omega} = \frac{3nR_{\oplus}^2 J_2}{2p^2} \cos i \tag{19}$$

$$\dot{\omega} = \frac{-3nR_{\oplus}^2 J_2}{4p^2} \left(5\cos^2 i - 1\right) \tag{20}$$

$$\dot{M} = \frac{-3nR_{\oplus}^2 J_2}{4p^2} \left(3\cos^2 i - 1\right) \sqrt{1-e^2} \tag{21}$$

Where the orbital parameter (p) is given as: $p = a\,(1-e)$, and $R_{\oplus}$ is the equatorial radius of Earth.

Given the known parameters i, a_{GOM}, a_{ROM}, and Δh_D, then the only remaining variable in the problem is the drift orbit duration t_D. The problem is then reduced to finding the minimum t_D that satisfies Equation 14. Selecting different drift orbit altitudes (through the internal variable Δh_D) will change t_D, and will also change the total round-trip reconfiguration ΔV which is the sum of the ΔV required to complete the two reconfiguration Hohmann transfers and the ΔV to return to GOM after the event response. Figure 8 shows the tradeoff between minimizing reconfiguration time and minimizing reconfiguration ΔV for a situation where a_{ROM} is $20km$ less than a_{GOM}. The top plot shows how t_D decreases as the drift orbit altitude is further from the ROM altitude. Also, the ascending pass RGT favors drift orbits higher than ROM while the descending pass RGT favors drift orbits lower than ROM. Each ground location has two different RGT orbits that pass overhead, one that passes over while ascending in latitude coverage (ascending pass) and one that is descending in latitude coverage (descending pass). The middle plot shows the time to first pass t_{fp}, which is defined as the amount of time that it takes for reconfiguration as well as the additional amount of time spent in ROM waiting to provide the first coverage for the disaster event. Here we see that in some cases, spending more propellant to increase Δh_D reduces t_D and t_R, but does not reduce t_{fp}. This is caused by the situation where the faster reconfiguration places the satellite into ROM faster, but since only a single pass is provided per day to the event location, the actual time to first pass remains the same. In this case, the operator would likely choose to save propellant by reconfiguring slower, with no increase in t_{fp}. The bottom plot shows that ΔV_R increased with increasing $|\Delta h_D|$ and hits a minimum value at the amount of ΔV needed to complete the round trip maneuver from GOM to ROM and back.

Several filtering steps are then taken to eliminate the dominated solutions in the t_{fp} vs. ΔV_R tradeoff to reduce the reconfiguration decision space. A dominated solution is one where another option provides either the same or lower t_{fp}, while simultaneously providing the same or lower ΔV_R. Several examples of dominated solutions are present in Figure 8. First, we see that choosing a drift orbit in between GOM and ROM altitudes increases t_{fp} without reducing ΔV_R and is therefore a dominated solution. Second, higher drift orbits are dominated since they yield larger t_{fp} for a given ΔV_R than lower drift orbits. Third, as was previously explained, sometimes increasing $|\Delta h_D|$ decreases t_R but not t_{fp}, and therefore these solutions are also dominated. This filtering process is illustrated in Figure 9. The top plot shows t_{fp} vs. ΔV_R for all reconfiguration options shown previously in Figure 8, and the bottom plot shows only the non-dominated reconfiguration options for both ascending and descending pass RGT.

B. Determining the Optimal Set of Satellites to Reconfigure

The next part of the reconfiguration process must determine the optimal way to respond to the event. The satellite assignment problem must determine for each event response: 1. how many satellites to reconfigure; 2. which specific combination of satellites to reconfigure; 3. how fast should the satellites be reconfigured (selection of drift orbit altitude Δh_D); and 4. which RGT orbit should each reconfigured satellite be placed in (selection of ascending or descending pass coverage). Additionally, the assignment process should penalize assignment of satellites that have less than the average amount of propellant left in the constellation (to avoid premature propellant depletion of some satellites) and should prevent assignment of satellites that have run out of reconfiguration propellant or satellites that have failed. Solving this optimization problem concurrently ensures that globally efficient solutions are found, but results in a computationally complex combinatorial optimization problem. The total decision space for a single event response is given as:

$$N_D = \sum_{k=1}^{N_T} \left[(1 + 2N_{\Delta h_D})^k \binom{N_T}{k} \right] \tag{22}$$

Where N_T is the total number of satellites in the constellation and $N_{\Delta h_D}$ is the number of non-dominated drift orbit altitude options for each satellite. If the number of non-dominated solutions is around 5 total per satellite (i.e.$(1 + 2N_{\Delta h_D}) = 5$), the the total non-dominated decision space N_D is: 2.2×10^9 for $N_T = 12$; 2.8×10^{18} for $N_T = 24$; and 7.7×10^{37} for $N_T = 48$. The size of the decision space is prohibitively large for enumeration based optimization and therefore an alternative assignment optimization strategy was needed.

To overcome the computational complexity of the combinatorial assignment problem, we have implemented a dynamic programming based optimization routine that efficiently finds the optimal assignment of satellites. The routine minimizes reconfiguration cost per expected event performance generated for maneuvering 1 of N_T total satellites, 2 of N_T total satellites, up to N_T of N_T satellites. Total reconfiguration cost C_R is defined as the sum of the individual satellite reconfiguration costs $C_R = \Sigma C_{R,i}$, which is defined as the ΔV_R with a penalty added for satellites that have less than the average amount of propellant in the constellation as follows:

$$C_{R,i} = \Delta V_{R,i} - \min\left(0,\ \Delta V_{sat,i} - \frac{1}{N_T}\sum_{k=1}^{N_T} \Delta V_{sat,k}\right) G_{pen} \tag{23}$$

Where $\Delta V_{sat,i}$ is the *ith*'s satellite ΔV_{recon} and G_{pen} is a penalty function gain which is set to 0.1 in this paper. G_{pen} can be used to control how tightly satellites are kept at the same propellant level. The penalty function works by increasing the satellite reconfiguration cost proportional to the difference in its remaining ΔV_{recon} compared to the average remaining ΔV_{recon} in the constellation. Additionally, no penalty is applied for satellites with higher than the average propellant left.

The satellite assignment problem outputs the estimated total utility generated during event response as a function of the total amount of ΔV_R for reconfiguring 0 of N_T satellites, 1 of N_T satellites, etc., as shown in Figure 10. Here

we see the effect of diminishing returns as the number of satellites to be reconfigured increases, the additional P_E generated steadily decreases.

C. Deciding How to Respond to Each Event

In an operational system, decision makers would select the appropriate response from the optimal assignment choices shown in Figure 10. A decision model employing decision rules emulates this decision making process in the simulation model to enable the rapid evaluation of candidate constellation designs during optimization. The decision model must capture two primary characteristics of the decision making process. The first recognizes that there are two competing objectives for decision makers: to provide effective coverage for the current event response, and to conserve system resources for later event responses. The second recognizes that the decision maker will likely alter their preferences based on the current state of the system as well as changes to forecast demand. These two characteristics are captured by a decision model which converts the competing objectives of maximizing current event response performance and minimizing overall fuel expenditure into a single objective to be maximized using a weighing variable α. This single objective is written as:

$$J = -\alpha \sum (\Delta C_R) + (1 - \alpha) P_E \tag{24}$$

Where J is the objective to be minimized, $\sum (\Delta C_R)$ is the total reconfiguration cost across all satellites, P_E is the total event performance, and α is the decision model weight variable. In an operational system, the decision maker would have to balance the use of resources so that the system would not run out of propellant early, or have unused propellant at the end of life. The α variable samples different decision maker preferences between these two objectives. α values that are too high place too much emphasis on conserving propellant, likely leaving unused propellant in the satellites at system end of life. This will correspond to lost opportunity for generating utility. Conversely, α values that are too small place too much emphasis on generating utility. This will cause the system to run out of propellant early and the system will then be unable to effectively respond to later events. To avoid selecting α apriori, we decided early on to make α a design variable letting the optimization process select the appropriate α for a given constellation design. In this way, the optimization process will likely select α so that most of the reconfiguration propellant is used for event responses.

The second characteristic that the decision model needed to capture was the likely situation where the decision maker would change their preferences during the system lifetime as information regarding propellant expenditure became known. To model this behavior, α is continuously adapted during the regional simulation based on remaining propellant. The adapted α is given as:

$$\alpha = \begin{cases} \min\left[\left(-\delta_P G_{DM} + 1\right) \alpha_0, 1\right] & \delta_P \geq 0 \\ \alpha_0 / \left(\delta_P G_{DM} + 1\right) & \delta_P < 0 \end{cases}$$

Where δ_P is related to the difference between the predicted and actual remaining propellant, G_{DM} is a gain term controlling how responsive the decision maker is to changing α based on new information, and α_0 is the initial weighting which is a system design variable. δ_P is calculated by comparing the average remaining propellant in the constellation with an estimate of how much propellant should be remaining. This estimate is based on assuming a constant depletion of propellant during the system's lifetime. If $\delta_P > 0$ then the system is using propellant at a slower rate than expected and α can be decreased for future events to avoid having excess propellant at the end of the system lifetime. If $\delta_P < 0$ then the system is using propellant faster than expected and α should be increased so that the system does not run out of propellant early. δ_P, normalized to the interval -1 to 1 is given as:

$$\delta_P = \frac{\left(\overline{\Delta V_{sat}} - \kappa \Delta V_{recon}\right)}{\Delta V_{recon}} \tag{25}$$

Where $\overline{\Delta V_{sat}}$ is the average satellite reconfiguration propellant remaining, ΔV_{recon} is the starting reconfiguration propellant, and κ is the fractional remaining lifetime of the constellation. We have set $G_{DM} = 3$ in this paper.

Once α is determined, then the decision model objective function can be calculated using Equation 24. Figure 10 (bottom) shows J for three different α weightings for the assignment output introduced in Figure 10 (top). In this example, when $\alpha = 0.08$, J is maximized when 10 satellites are reconfigured. As α increases, and more emphasis is placed on conserving propellant, the number of satellites that maximizes J is reduced. When $\alpha = 0.4$, zero satellites are reconfigured and regional coverage is only provided from GOM similar to a static constellation.

The design variable α_0 is part of the constellation design and should inform the decision maker as to what operational preferences should be used in order to maximize overall system lifecycle performance. The reasoning behind this is that the optimization process designed the system to perform well over the range of uncertain parameters with the specific value of α_0. In the case where no additional information is gathered, changing α_0 will only yield suboptimal system performance. It is likely, that new information regarding the system and future event probabilities will arise, in which case a new optimization should be performed to determine the new α_0 weighting that should be used.

VI. Monte Carlo Simulation

We have modeled two uncertainties including the timing and location of future disaster events. A two-dimensional probability distribution function (PDF), based on estimates of the distribution of future disaster events [26], is used to draw the locations of future disaster events. The estimates are normalized by total economic loss risk data for a number of disaster types including cyclones, earthquakes, floods, and volcanoes. The PDF, shown in Figure 11, is comprised of a 2.5 minute grid of global multi-hazard total economic loss risks with equal weighting between the four disaster types. The number and specific timing of events during the constellation lifetime is also uncertain. We have modeled this uncertainty as a normal distribution of the time between disaster events. The shape of the distribution is controlled by two parameters representing the mean time between events and the standard deviation of the time between events.

These two parameters should be tailored to specific stakeholder goals and objectives. In this paper we assume that the the mean time between events is 3 months and the standard deviation is 1 month. The minimum time between events is constrained to 2.5 weeks to ensure that events do not overlap given a 14 day event duration. While overlapping events is not a focus here, it is an important topic for future research since the availability of resources must then be split between the two events. Additionally, decision makers would then have to consider the possibility of leaving satellites in GOM as a reserve in case a second event were to occur during the current event response. Satellite failures could also be introduced as an uncertainty in future work.

The uncertain event ground locations and uncertain event rate are then sampled together in order to construct a list of targets (target deck) for each Monte Carlo simulation run. Each target deck represents one possible future operating context for the system and is comprised of a list of ground locations (φ and λ) and the time of each event. The number of events in the target deck is determined by sampling the event rate distribution to see how many events occur within the constellation lifetime. The process starts off by randomly sampling the event rate PDF to determine the time of the first event. Next, another random sample is taken from the event rate PDF which corresponds to the time between the first and second events. Therefore, the time of the second event is the cumulative sum of the first two samples. This process is continued until the cumulative sum exceeds the system design lifetime of 5 years. Available computational resources limited the maximum number of Monte Carlo samples to no more than 24, which gives a 68% confidence interval size of around 3% of median value. Therefore, the actual median performance value is within ±1.5% of the performance value used to determine the population fitness with 68% confidence.

VII. Multi-Objective Evolutionary Optimization

Multi-Objective optimization is used to find the set of efficient designs for each architecture and operational scenario that maximize performance while simultaneously minimize cost. The multi-objective optimization problem formulation is expressed as:

$$\begin{aligned} &\min \mathbf{J}(\mathbf{x}, \mathbf{p}) \\ \text{subject to: } &\mathbf{g}(\mathbf{x}, \mathbf{p}) \leq 0 \\ &\mathbf{h}(\mathbf{x}, \mathbf{p}) = 0 \\ &x_{i,LB} \leq x_i \leq x_{i,UB} \ \ (i = 1, \ldots, n) \end{aligned}$$

Where $\mathbf{J}$ are the objectives to be minimized, $\mathbf{g}$ are inequality constraints, $\mathbf{h}$ are equality constraints, $\mathbf{p}$ are fixed parameters, and $\mathbf{x}$ is the design vector comprised of design variables with bounds $x_{i,LB}$ and $x_{i,UB}$. The objectives are to maximize the median event performance of the constellation over N_{MC} Monte Carlo samples while simultaneously minimizing constellation cost. Four inequality constraints are enforced: the altitude (GOM, ROM and any drift orbit altitudes) must be greater than $300km$; the propulsion system mass fraction (M_P/M_w) must be less than 0.42; the maximum revisit time for partial global coverage (from -60° to 60° latitude) in GOM must be less than 24 hours;

and N_T must be less than equal to 36 satellites. The number of design variables in the design vector is different for reconfigurable and static constellation designs. For the reconfigurable architecture there are nine total design variables including: five continuous variables, three integer variables, and one categorical variable as shown in Table 3. For the static architecture, N_o/N_d, ΔV_R, and α_0 are eliminated and Δalt is converted into a GOM altitude variable with bounds of 300km to 1000km, resulting in six total design variables.

A. Modified ϵ-NSGA-II Algorithm

Traditional optimization techniques attempt to find the single "globally optimal" design that best achieves a single objective like maximizing performance or minimizing cost. However, complex, real-world systems often contain multiple competing objectives. In these cases, multi-objective optimization attempts to find the set of non-dominated designs where improving one objective always results in worsening another. Absent further information or preferences, one non-dominated design cannot be said to be better than another. Therefore, the goal of multi-objective optimization is not to output the best design, but to give designers more information about the direct tradeoff of multiple competing objectives, hopefully leading designers to better informed decision making. The goal of multi-objective optimization in this paper is to find the non-dominated set of designs that maximize median event performance over uncertain operating conditions while simultaneously minimizing total system cost. The satellite constellation optimization problem presented here is difficult because: 1. the objective space is discontinuous and non-linear, 2. the design vector contains continuous, integer, and discreet (categorical) variables, and 3. functional evaluations are computationally expensive. Given these difficulties, we have implemented a novel optimization routine that builds on ϵ-NSGA-II [27,28] by adding several additional features from the ϵ-MOEA [29] and Borg-MOEA [30] optimization routines. The added features include: ϵ-dominance archiving to avoid deterioration; ϵ-Progress to detect stagnation; time continuation to recover from stagnation; multiple recombination operators to improve search; and a new termination criterion to provide a common stopping condition for all optimization runs presented in this paper. The rest of this section summarizes these features.

Laumanns et al. [29] introduced the ϵ-dominance archiving concept to solve the deterioration problem by preserving diversity and guaranteeing convergence. Deterioration occurs when a fixed population size forces the algorithm to remove non-dominated designs [31] and can eventually lead the algorithm to sacrifice convergence for diversity. ϵ-dominance allows the decision maker to specify a desired resolution for each objective, denoted by ϵ, which effectively divides the objective space into rectangular boxes with side length ϵ. In order for a solution to dominate another solution, it must occupy an ϵ-box that is better in one or more of the objectives. If two solutions occupy the same ϵ-box, only the solution closer to the lower left corner (assuming minimization) is saved, while the other is eliminated. This process ensures that solutions do not bunch close together on the Pareto front and ensures that the computational cost is commensurate with the desired resolution of the final non-dominated front.

The ϵ-Archive concept, first implemented in ϵ-MOEA [29], maintains an archive of ϵ-box non-dominated solutions

external to the normal population. This ensures preservation of the best solutions and mitigates deterioration. ϵ-NSGA-II uses the ϵ-Archive to seed a new population after restart while ϵ-MOEA and Borg-MOEA use the ϵ-Archive to generate one of the parents during recombination, which is implemented in this paper. Hadka et al. [30] introduced ϵ-Progress as an efficient method for measuring convergence using ϵ-dominance. During the ϵ-Archive update process, a counter tracks how many solutions are accepted into the archive that reside in new ϵ-boxes. This effectively uses ϵ as a minimum threshold for measuring improvement. The number of new ϵ-boxes added to the archive is tracked for each generation. ϵ-Progress is then defined as the cumulative number of new ϵ-boxes added to the archive over the most recent five generations divided by the archive size. When the ϵ-Progress metric drops below a specified value (0.2 is used in this paper), then the optimization algorithm is determined to have stagnated and optimization is restarted.

While ϵ-Dominance and the ϵ-Archive significantly improves diversity and convergence, stagnation can still occur. Time continuation has been proposed [32, 33] as a way of recovering from stagnation. Traditional evolutionary optimization techniques employ a single large population of fixed size. In the time continuation paradigm, the optimization process is split between many 'epochs'. When the population in one epoch stagnates, a new epoch is started by seeding the new population with a diverse set of good performing members of the last epoch. Early research on time continuation kept the population size constant between epochs; however, recent research has found that maintaining the population size proportional to archive size improved convergence for complex problems [34]. Therefore, we allow the population size to vary from epoch to epoch based on maintaining a 4:1 population size to archive size ratio, shown to perform well in previous research [28].

The recombination operator is the primary search mechanism for evolutionary algorithms and is crucial to the success of the optimization process. Deb et al. [35] classified crossover operators into two categories: variable-wise and vector-wise operators. Variable-wise operators treat each variable separately and, therefore, do not perform well for problems with significant linkage between design variables. Vector-wise operators use linear combinations of the parents' entire variable vectors to create an offspring variable vector. This strategy preserves the linkage between variables and performs well for problems with coupled variables [36]. Crossover operators can also use a single parent, two parents or many parents to create offspring. Recent research has shown that multi parent operators can improve convergence for complex problems [37]. Given these factors, we have implemented a random multi-operator approach where the optimization algorithm randomly selects an operator with equal probability when producing each offspring. This approach is similar to the approach taken in Borg-MOEA, but we chose not to implement the adaptive scheme where the probability of selecting different operators is adjusted based on the historical success of the offspring generated by each operator. Nebro et al. [38] showed that the adaptive scheme did not provide significant advantages over the random approach for bi-objective problems like the one studied here. The diverse set of operators implemented in the modified ϵ-NSGA-II optimization routine are: 1. Uniform mutation ($P_c = 1/n$, 1 parent); 2. Simulated binary crossover ($\lambda = 20$,$P_c = 0.9$, 3 parents [39, 40]); 3. Unimodal normal distribution ($\sigma_\zeta = 0.5$, $\sigma_\eta = \frac{0.35}{\sqrt{n}}$, 3 parents [41]); 4. Parent-centric crossover ($\sigma_\zeta = \sigma_\eta = 0.1$, 3 parents [42]); 5. Differential evolution ($CR = 0.6$, $F = 0.5$, 4

parents [43]); and 6. Simplex crossover ($\varepsilon = \sqrt{n+1}$, $n+1$ parents [44]) where n is the number of design variables. Polynomial mutation is also applied to all offspring except those produced by uniform mutation using recommended settings from Deb et al. [45]: distribution index of 100 and a mutation probability of $1/n$.

The optimization routine restart procedure is triggered when either stagnation is detected via ϵ-Progress or when the population to archive size ratio differs from 4:1 by more than 25%. When a restart is triggered, the new population size is calculated to best achieve a 4:1 ratio, and the population is regenerated through a modified injection scheme. The original injection scheme proposed for ϵ-NSGA-II [28] added all of the archive members to the new population and then the remaining slots were randomly populated. Instead, we use a modified injection scheme introduced by Hadka et al. [30] for Borg-MOEA where the remaining population slots are filled by mutating randomly selected archive members using uniform mutation with a probability of $1/n$, again where n is the number of design variables [30]. This strategy maintains a higher level of elitism than simple random generation of new population members. Additionally, the crossover tournament size is adjusted to keep the tournament size proportional to the population size as is done in Borg-MOEA [30]. Figure 12 shows the overall modified ϵ-NSGA-II optimization process. The main loop is similar to the original NSGA-II algorithm with the exception of the offline archive, restart procedure, and termination criterion. The optimization routine continues until the rate of improvement drops below a specified threshold. The rate of improvement is defined as the number of archive updates per 100 functional evaluations, averaged over the last 2500 functional evaluations, as a percentage of the archive size. This improvement rate is similar to ϵ-Progress introduced previously, however, it adds in a smoothing term to overcome short-term variations in the rate of convergence. The optimization process is then terminated when the rate of improvement drops below 2.5 giving a common stopping condition for all of the optimization runs presented in this paper.

B. Parallel Computing & Computational Resources

We recognized early on that parallel computing was needed to overcome the long functional evaluation computation time encountered in the reconfigurable satellite constellation problem. For example, the simulation model run time for a 36 satellite constellation was over $20s$ for static designs and over $76s$ for reconfigurable designs. We chose to implement a master-slave approach, which utilizes a master processor that is responsible for all of the optimization functions (recombination, archiving, restarts...) and communicates with a set of slave processors to perform functional evaluations. Previous research has shown that the master-slave approach suffers significant loss in efficiency due to generational synchronization with heterogeneous computational systems [46,47]. This is caused by the situation where faster processors must wait for slower processors to finish execution, which effectively slows all resources down to the speed of the slowest processor. This behavior is also encountered for systems with functional evaluations of varying computational time like the simulation model with varying constellation size considered in this paper. To overcome this problem, we implement a novel batch processing approach, which attempts to even out the computational burden sent to each slave processor for each generation. This process splits up the total computational task for each generation

into a set of discrete computation blocks, each with similar computational burden. Additionally, computation blocks with longer estimated processing times are sent to the slave processors first so that blocks with shorter processing times can be used later to even out computation time differences. This approach yielded parallelization speedup efficiencies of around 50%. Future work could implement a stationary optimization routine similar to the one used in Borg-MOEA to eliminate generational synchronization efficiency losses.

All optimization runs were performed on MIT Lincoln Laboratory's LLGrid [48] computing cluster running MATLAB R2014a. While we had access to 1024 processors in total, each optimization run was limited to 256 processors in order to mitigate the generational synchronization efficiency loss by ensuring that there were enough computation blocks for the algorithm to even out computational load. Limiting each optimization run to 256 processors allowed four optimization runs to be performed in parallel to utilize the full 1024 processor capability. In total, 8 optimization runs were completed over a span of 10 days, requiring almost 4 million simulation model calls. Without parallel computing, this effort would have taken approximately 20 years of non-stop computation by a modern computer processor (an Intel Core i5-2410M was used as reference). Post processing included performing a 96 sample Monte Carlo simulation on the final non-dominated fronts using a 16 core Dell T7200 Precision workstation equipped with two E5-2665 eight core processors running Ubuntu 12.04 and MATLAB R2014a.

VIII. Results and Discussion

This section presents a comparison of optimized static and reconfigurable designs for four operational scenarios. These scenarios were chosen to investigate how changing the desired temporal and spatial resolution parameters would affect the cost effectiveness of static and reconfigurable architectures and the resulting value of reconfigurability. For scenario 1, an ideal design would provide persistent coverage with $1hr$ temporal resolution and $1m$ spatial resolution over a 5 year lifetime for event locations drawn from the natural disaster PDF. Scenarios 2 through 4 vary the desired temporal and spatial resolution to $0.5hr$-$1m$, $1hr$-$0.5m$, and $0.5hr$-$0.5m$, respectively.

Figure 13 shows the optimization run statistics for the reconfigurable architecture as a function of the number of completed functional evaluations. The top plot shows how the population to archive size ratio and ϵ-Progress metric changes throughout the optimization. As discussed earlier, when either exceeds the a specified limit, the optimization routine restarts. The population size dynamically adjusts to the archive size during restart, shown in the middle plot. Optimization continues until the ϵ-Termination criterion, common to all optimization runs presented in this paper, is met, as shown in the bottom plot. For scenario 1, the reconfigurable architecture optimization terminated after 17,887 functional evaluations (427,288 simulation model calls with a Monte Carlo sampling size of 24), while the static architecture optimization terminated after 25,405 functional evaluations (609,720 simulation model calls). The average number of functional evaluations for scenarios 1-4 was around 21,902 for reconfigurable designs and 17,869 for static designs.

Several post-processing steps prepared the data for analysis and comparison following optimization termination.

First, a non-domination sort procedure was used to find the set of non-dominated designs encountered over all functional evaluations performed during optimization. For scenario 1 there were 245 unique non-dominated designs for the reconfigurable architecture and 263 unique non-dominated designs for the static architecture. Roughly 40 well spaced designs were selected through non-domination sorting for further evaluation with a 96 sample Monte Carlo simulation. This additional sampling solved two problems. First, due to computational time limitations, the Monte Carlo sampling size used in the optimization routine was limited to 24 samples giving a 68% confidence interval size for the median performance of around 3%. Increasing the sample size to 96 halved this to around 1.5%. Second, since the optimization process was evaluating designs over several independent Monte Carlo simulations, it is prone to identify 'favorable' Monte Carlo sample draws. The 96 sample final evaluation gives an unbiased estimate for the median performance. Figure 14 shows the final non-dominated fronts for the reconfigurable and static architectures.

Direct comparison of the reconfigurable and static non-dominated fronts shows that reconfigurable designs completely dominate the static designs in terms of maximizing performance while minimizing cost. The value of reconfigurability (VoR) was previously defined as the reduction in total system cost, while maintaining the same performance, made possible by adopting the more efficient reconfigurable architecture. VoR is determined by computing the horizontal distance between the reconfigurable and static non-dominated fronts in Figure 14. Additionally, a bootstrap re-sampling with replacement procedure is used to measure sampling error effects in the VoR calculation. 1000 bootstrap samples are generated for each non-dominated design to construct 1000 different splines representing the reconfigurable and static non-dominated fronts. VoR is then calculated as the horizontal distance between these splines. Figure 15 shows the mean VoR (thick line) as well as the high and low 3σ VoR values (thin lines) as a function of the normalized event performance, defined as a the median event performance $\tilde{P_E}$ divided by the maximum possible event performance P_{max} if the system exactly met temporal and spatial resolution goals. The top plot shows VoR in terms of FY2010 dollars while the lower plot shows VoR in terms of a percentage of the static architecture cost. Here we see that VoR is between 20 to 50% over most of the performance range, meaning that the reconfigurable architecture is 20 to 50% less expensive than a similarly performing static architecture. Also, VoR generally increases with increasing performance indicating that reconfigurable designs are better able to provide the desired coverage.

Figure 16 shows design details corresponding to the reconfigurable (colored black) and static (colored gray) non-dominated designs as a function of normalized performance. The plotted design details include: the total number of satellites (N_T), aperture size (D), GOM altitude (h_{GOM}), inclination (i), number of orbit planes (N_p), and total ΔV (ΔV_T). Here we see that the non-dominated designs for the two architectures are significantly different. In the low performance region ($\tilde{P_E}/P_{max} < 0.6$), static designs tend to feature more satellites at a lower altitude when compared to reconfigurable designs. However, despite the lower altitude, static designs feature similar aperture sizes, since static designs must have larger apertures to maintain good spatial resolution for off-nadir passes, while reconfigurable designs ensure nadir viewing. The reduction in the number of satellites for reconfigurable designs is caused by increased per-satellite utilization enabled by the ability to reconfigure. This is the fundamental cost reduction driver for recon-

figurable architectures. In the higher performance region ($\tilde{P_E}/P_{max} > 0.6$), the static architecture runs up against the maximum satellite constraint which is set to 36 satellites for the analysis presented in this paper. Therefore, to further improve performance for static designs, altitude and aperture size must increase, inflating overall system cost.

Reconfigurable designs, except for very low performance designs that feature small numbers of satellites, tend to feature prograde inclinations of around 60° and a single satellite per orbit plane. This difference is likely caused by the tension between minimizing launch cost and maximizing performance. For small constellation sizes, it is difficult for two satellites to share a single launch because the orbit planes are too far apart in Ω to be serviced by differential orbital precession during the three month deployment period. Therefore, a dedicated launch for each satellite is needed, which increases launch cost. However, the time of pass for a specific event location in an RGT orbit for the reconfigurable architecture is a function of Ω only, and greater diversity in Ω leads to better persistence. This explains why high performance reconfigurable designs tend to feature only one satellite per orbit plane. Static designs tend to feature multiple satellites per orbit plane, fewer total orbit planes, and retrograde inclinations of around 120°. Therefore, for these static designs, the performance benefits of retrograde inclinations must outweigh the increase in launch cost to launch to higher inclinations. There is, however, some chatter between prograde and retrograde inclinations for low performing static designs, which indicates that the performance difference between prograde and retrograde designs is small. Reconfigurable designs also feature nearly twice the total propulsive capability of static designs in order to support reconfiguration maneuvers. The maximum propulsion system mass fraction constraint (set to 42% which corresponds to around $980 m/s$ using the propulsion system parameters specified earlier) is active for many non-dominated reconfigurable designs and relaxing this constraint will likely increase the value of reconfigurability.

The performance metric used in this thesis directly measures both temporal and spatial resolution for persistent surveillance and alleviates some of the problems encountered with traditional satellite coverage metrics. However, it is useful to see how the static and reconfigurable non-dominated designs found using the new performance metric compare in terms of the traditional metrics. Figure 17 shows a comparison of the static and reconfigurable non-dominated designs in terms of average and maximum revisit time, mean response time and mean GSD as a function of overall system cost. The traditional metric values plotted are the median metric value over all event responses during the system lifetime and over the 96 Monte Carlo samples. The metric value for each event response is defined as the daily value averaged over days 4 to 14 of the event response allowing the reconfigurable constellation time to reconfigure into ROM. The desired metric values are plotted as horizontal dashed lines. Figure 17 shows that the reconfigurable architecture provides better average revisit time and mean response time, and significantly better maximum revisit time. However, the static architecture provides better mean GSD for low-cost designs. This likely means that, for static designs, it is less costly to increase aperture size and therefore increase spatial resolution than to increase temporal resolution. Consequently, the optimization process places more emphasis on providing better mean GSD for these designs. This comparison shows that the traditional metrics tend to show the same trends as the new persistence metric introduced in this paper.

COmparing the results for different spatial and temporal resolution requirements shows that that VoR generally increases with increasing temporal and spatial resolution requirements and is more sensitive to increased temporal resolution, as shown in Figure 18. This indicates that the reconfigurable architecture is better able to handle increased temporal resolution requirements due to its more efficient use of satellites. The mean VoR over the normalized performance range of $0.5 \leq \tilde{P_E}/P_{max} \leq 1$ is 36.5% for scenario 1, 46.7% for scenario 2, 48.5% for scenario 3, and 55.4% for scenario 4.

IX. Conclusion

This paper introduces a new reconfigurable satellite constellation concept that is able to provide persistent coverage to support effective disaster response. The ability to reconfigure the constellation allows system operators to actively change the constellation pattern to focus coverage on a specific region. This improves satellite utilization and leads to smaller, more cost effective constellations. The results show that reconfigurable constellations cost 20 to 70% less than similarly performing static constellations despite additional maneuverability for reconfigurable designs.

This research has shown that a reconfigurable constellation provides cost-effective persistent regional coverage when the region to be covered is unknown or uncertain *a priori*. However, utilizing the reconfigurable constellation requires a significant departure from the way we traditionally operate satellite systems. First, the capability to reconfigure, primarily enabled by increased satellite maneuverability, must be built into the system upfront. Second, the reconfigurable satellite constellation will only be as responsive as the decision making and satellite commanding processes. Effective employment of this type of system will require: new optimization tools to quickly identify reconfiguration options; decision support tools to display this information to decision maker to aid in making quick decisions; and quick satellite commanding to enact reconfiguration.

This paper also develops and demonstrates a general framework to guide the design and optimization of reconfigurable satellite constellations specifically tailored to stakeholder objectives while considering uncertainty in the locations of future disaster events. Each constellation optimization was performed in less than 30 hours, demonstrating that parallel computing coupled with sophisticated optimization routines enable rapid spiral development of satellite constellations. This approach quickly identified efficient designs without overly restricting the design space. The framework is novel in that it avoids many of the assumptions and simplifications of past research by: 1. explicitly considering uncertainty in future operating conditions; 2. concurrently optimizing constellation pattern design, satellite design, and operations design; and, 3. utilizing multi-objective evolutionary algorithms and large scale parallel computing. The methodology is easily tailored to consider other constellation design problems and in a future paper the authors will show that it can quickly find efficient reconfigurable and static designs using several asymmetric constellation patterns.

Additionally, this paper introduces a new persistence figure of merit that captures how well realized coverage matches the desired temporal and spatial resolution. This metric eliminates without being skewed by statistical outliers,

a limitation of several traditional constellation coverage metrics. The optimization routine, using this new persistence metric, found designs that provided a good balance between meeting temporal and spatial resolution goals.

Acknowledgments

This work is sponsored by the Department of the Air Force under Air Force Contract #FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

References

[1] Paek, S. W., *Reconfigurable satellite constellations for geo-spatially adaptive Earth observation missions*, Master's dissertation, Massachusetts Institute of Technology, 2012.

[2] Legge, R. S., *Optimization and Valuation of Reconfigurable Satellite Constellations Under Uncertainty*, Doctoral dissertation, Massachusetts Institute of Technology, 2014.

[3] de Neufville, R., de Weck, O., Lin, J., and Scholtes, S., *Identifying real options to improve the design of engineering systems*, CRC press Boca Raton, FL, 2009, pp. 75–98.

[4] Wang, T. and De Neufville, R., "Real options "in" projects," *9th Real Options Annual International Conference, Paris, France*, 2005.

[5] Wang, T. and de Neufville, R., "Identification of Real Options "in" Projects," *4th Conference on Systems Engineering Research, Los Angeles, CA*, 2006.

[6] Williams, E. A., Crossley, W. A., and Lang, T. J., "Average and maximum revisit time trade studies for satellite constellations using a multiobjective Genetic Algorithm," *Journal of the Astronautical Sciences*, Vol. 49, No. 3, 2001, pp. 385–400.

[7] Lang, T. J., "A parametric examination of satellite constellations to minimize revisit time for low Earth orbits using a genetic algorithm," *Advances in the Astronautical Sciences*, Vol. 109, 2001, pp. 625–640.

[8] Ferringer, M. P. and Spencer, D. B., "Satellite constellation design tradeoffs using multiple-objective evolutionary computation," *Journal of Spacecraft and Rockets*, Vol. 43, No. 6, 2006, pp. 1404–1411.

[9] Leachtenauer, J. C., Malila, W., Irvine, J., Colburn, L., and Salvaggio, N., "General image-quality equation: GIQE," *Applied Optics*, Vol. 36, No. 32, 1997, pp. 8322–8328.

[10] Thurman, S. T. and Fienup, J. R., "Analysis of the general image quality equation," *Proc. SPIE 6978, Visual Information Processing XVII, 69780F*, 2008.

[11] Walker, J. G., "Continuous whole-earth coverage by circular-orbit satellite patterns," Technical Report 77044, Royal Aircraft Establishment, 1977.

[12] Wertz, J. R., Everett, D. F., and Puschell, J. J., *Space Mission Engineering: The New SMAD*, Microcosm Press, 2011.

[13] Wertz, J. R., Collins, J. T., Dawson, S., Koenigsmann, H. J., and Potterveld, C. W., "Autonomous Constellation Maintenance," *Mission Design & Implementation of Satellite Constellations, Proceedings of an International Workshop, held in Toulouse, France, November 1997*, 1997, pp. 263–273.

[14] Spores, R. A., Masse, R., Kimbrel, S., and McLean, C., "GPIM AF-M315E Propulsion System," *49th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, San Jose, California, 15-17 July 2013*, 2013.

[15] Dinardi, A. and Persson, M., "High Performance Green Propulsion (HPGP): A Flight-Proven Capability and Cost Game-Changer for Small and Secondary Satellites," *26th AIAA/USU Conference on Small Satellites, Logan, Utah, 13-16 August 2012*, 2012.

[16] Stahl, H. P., "Survey of cost models for space telescopes," *Optical Engineering*, Vol. 49, No. 5, 2010, pp. 053005–053005–8.

[17] Stahl, H. P., Stephens, K., Henrichs, T., Smart, C., and Prince, F. A., "Single-variable parametric cost models for space telescopes," *Optical Engineering*, Vol. 49, No. 7, 2010.

[18] Stahl, H. P., Henrichs, T., Luedtke, A., and West, M., "Update on parametric cost models for space telescopes," *Proc. SPIE 8146, UV/Optical/IR Space Telescopes and Instruments: Innovative Technologies and Concepts V, 81460F*, 2011.

[19] Stahl, H. P., Henrichs, T., Luedtke, A., and West, M., "Update on multivariable parametric cost models for ground and space telescopes," *Proc. SPIE 8442, Space Telescopes and Instrumentation 2012: Optical, Infrared, and Millimeter Wave, 844224*, 2010.

[20] Stahl, H. P., "Survey of cost models for space telescopes," *Optical Engineering*, Vol. 49, No. 5, 2010.

[21] Habib-Agahi, H., Mrozinski, J., and Fox, G., "NASA instrument cost/schedule model," *Aerospace Conference, 2011 IEEE*, 2011, pp. 1–19.

[22] Kramer, H. J., *Observation of the Earth and its Environment: Survey of Missions and Sensors*, Springer, 2002.

[23] NASA Headquarters Cost Analysis Division, *NASA Cost Estimating Handbook*, 2008th ed., 2008.

[24] Lafleur, J. M., Fleming, E. S., and Saleh, J. H., "Response Surface Equations for Expendable Launch Vehicle Payload Mass Capability," *Journal of Spacecraft and Rockets*, Vol. 49, No. 1, 2012, pp. 185–189.

[25] Avendano, M. E., Davis, J. J., and Mortari, D., "The Lattice Theory of Flower Constellations," *Proceedings of the 2010 Space Flight Mechanics Meeting Conference. San Diego, CA*, 2010.

[26] Dilley, M., *Natural disaster hotspots: a global risk analysis*, Vol. 5, World Bank Publications, 2005.

[27] Reed, P., Minsker, B. S., and Goldberg, D. E., "Simplifying multiobjective optimization: An automated design methodology for the nondominated sorted genetic algorithm-II," *Water Resources Research*, Vol. 39, No. 7, 2003.

[28] Kollat, J. B. and Reed, P. M., "The value of online adaptive search: a performance comparison of NSGAII, ε-NSGAII and εMOEA," *Evolutionary Multi-Criterion Optimization*, Vol. 3410, Springer, 2005, pp. 386–398.

[29] Laumanns, M., Thiele, L., Deb, K., and Zitzler, E., "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary computation*, Vol. 10, No. 3, 2002, pp. 263–282.

[30] Hadka, D. and Reed, P., "Borg: An auto-adaptive many-objective evolutionary computing framework," *Evolutionary computation*, Vol. 21, No. 2, 2013, pp. 231–259.

[31] Hadka, D. and Reed, P., "Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization," *Evolutionary Computation*, Vol. 20, No. 3, 2012, pp. 423–452.

[32] Goldberg, D. E., "Using Time Efficiently: Genetic-Evolutionary Algorithms and the Continuation Problem," Tech. rep., Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, 1999.

[33] Srivastava, R. P., "Time continuation in genetic algorithms," Tech. Rep. IlliGAL Report No. 2001021, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign, 2002.

[34] Tang, Y., Reed, P., and Wagener, T., "How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration?" *Hydrology and Earth System Sciences Discussions*, Vol. 10, No. 2, 2006, pp. 289–307.

[35] Deb, K. and Jain, H., "Self-Adaptive Parent to Mean-Centric Recombination for Real-Parameter Optimization," Tech. Rep. KanGAL report 2011001, Kanpur Genetic Algorithms Laboratory, Indian Institute of Technology Kanpur, 2011.

[36] Deb, K., Sinha, A., and Kukkonen, S., "Multi-objective test problems, linkages, and evolutionary methodologies," *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, ACM, 2006, pp. 1141–1148.

[37] Patel, R. and Raghuwanshi, M., "Multi-objective optimization using multi parent crossover operators," *Journal of Emerging Trends in Computing and Information Sciences*, Vol. 2, No. 2, 2010, pp. 33–39.

[38] Nebro, A. J., Durillo, J. J., Machín, M., Coello, C. A. C., and Dorronsoro, B., "A Study of the Combination of Variation Operators in the NSGA-II Algorithm," *Advances in Artificial Intelligence*, Springer, 2013, pp. 269–278.

[39] Agrawal, R. B. and Deb, K., "Simulated binary crossover for continuous search space," *Complex Syst.*, 1995, pp. 115–148.

[40] Deb, K. and Kumar, A., "Real-coded genetic algorithms with simulated binary crossover: Studies on multimodel and multiobjective problems," *Complex Syst.*, Vol. 9, No. 6, 1995, pp. 431–454.

[41] Kita, H., Ono, I., and Kobayashi, S., "Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms," *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, Vol. 2, IEEE, 1999.

[42] Deb, K., Joshi, D., and Anand, A., "Real-coded evolutionary algorithms with parent-centric recombination," *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on*, Vol. 1, IEEE, 2002, pp. 61–66.

[43] Storn, R. and Price, K., "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*, Vol. 11, No. 4, 1997, pp. 341–359.

[44] Tsutsui, S., Yamamura, M., and Higuchi, T., "Multi-parent recombination with simplex crossover in real coded genetic algorithms," *Proceedings of the genetic and evolutionary computation conference*, Vol. 1, 1999, pp. 657–664.

[45] Deb, K. and Deb, D., "Analyzing Mutation Schemes for Real-Parameter Genetic Algorithms," Tech. Rep. KanGAL report 2012016, Kanpur Genetic Algorithms Laboratory, Indian Institute of Technology Kanpur, 2012.

[46] Ferringer, M. P., Clifton, R. S., Thompson, T. G., Spencer, D. B., Melton, R. G., Reed, P. M., Crossley, W. A., and Coverstone, V. L., "Efficient and Accurate Evolutionary Multi-Objective Optimization Paradigms for Satellite Constellation Design," *Journal of Spacecraft and Rockets*, Vol. 44, No. 3, 2007, pp. 682–691.

[47] Reed, P. M., Kollat, J. B., Ferringer, M. P., and Thompson, T. G., "Parallel evolutionary multi-objective optimization on large, heterogeneous clusters: An applications perspective," *Journal of Aerospace Computing, Information, and Communication*, Vol. 5, No. 11, 2008, pp. 460–478.

[48] Bliss, N. T., Bond, R., Kepner, J., Kim, H., and Reuther, A., "Interactive grid computing at Lincoln Laboratory," *Lincoln Laboratory Journal*, Vol. 16, No. 1, 2006, pp. 165.

[49] Oxfort, M., "The RapidEye Mission," *WGISS-24 Host Workshop*, 2007.

[50] Davies, P., Chizea, F., Cawthorne, A., Carrel, A., Gomes, L., da Silva Curiel, A., and Sweeting, S. M., "Commissioning of the NigeriaSat-2 High Resolution Imaging Mission," *26th AIAA/USU Conference on Small Satellites, Logan, Utah, 13-16 August 2012*, 2012.

[51] Chern, J.-S., Ling, J., and Weng, S.-L., "Taiwan's second remote sensing satellite," *Acta Astronautica*, Vol. 63, No. 11, 2008, pp. 1305–1311.

[52] Tam, W. H., Debreceni, M. J., Hersh, M. S., and Nye, C. D., "Low cost Derivative Tanks for Spacecraft and Launch Vehicles (updated as of May 2012)," 1999, Retrieved on 05/11/2014 from http://www.psi-pci.com/Technical_Paper_Library/AIAA%2099-2831%20Diaphragm%20Tank%20Updated%20List%20May%202012.pdf.

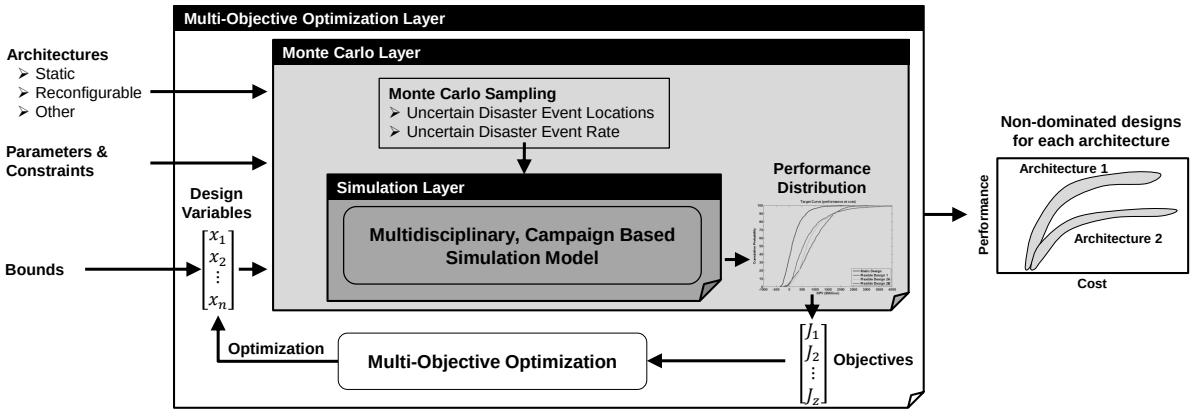

Figure 1. Constellation design and optimization framework

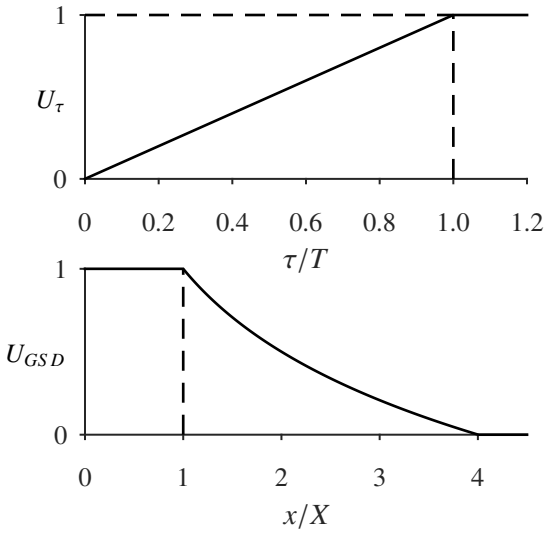

Figure 2. The persistence metric is comprised of a temporal utility term (top) and a spatial utility term (bottom)

[a]eoPortal. Satellite Missions Database. https://directory.eoportal.org. Accessed: 11-07-2013.
[b]SSTL. Xenon propulsion system datasheet. http://www.sstl.co.uk. Accessed: 11-07-2013.
[c]Jean-Phillipe Donnio. The satellite encyclopedia. http://www.tbs-satellite.com/tse/online/index_alpha.html. Accessed: 11-08-2013.

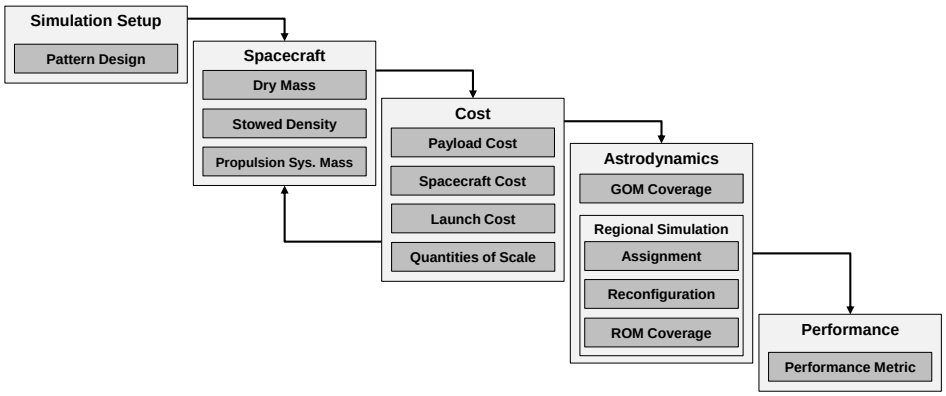

Figure 3. The simulation model is comprised of five modules arranged to minimize feedback and maximize computational efficiency

Table 1. Properties of Selected Optical Earth Observation Satellites

Spacecraft	Aperture cm	Wet Mass kg	Dry Mass kg	Stowed Size m, m, m	Stowed Volume m^3	Density kg/m^3	References
RapidEye	14.5	166.4	154.4	1.17, 0.78, 0.94	0.858	193.9	[49],[a]
EROS-A	30	260	230	2.23, 1.2, 1.2	3.21	81.0	[a]
NigeriaSat-2	38.5	286	274	0.79, 1.2, 1.2	1.456	188.2	[50],[ab]
FormoSat-2	60	746	665	2.4, 1.6, 1.6	6.14	121.4	[51],[a]
WorldView-1	60	2500	2090	3.6, 2.5, 2.5	22.5	111.1	[52],[a]
Quickbird 1,2	61	951	911	3.0, 1.6, 1.6	7.68	123.8	[22, 52],[c]
Pléiades 1B	65	1015	940	3.0, 2.5, 2.5	12	84.6	[a]
Ikonos 1,2	70	728	658[†]	1.83, 1.57, 1.57	4.51	161.4	[a]
GeoEye-1	110	1955	1811	4.35, 2.7, 2.7	31.7	61.7	[a]
WorldView-2	110	2800	2390	5.7, 2.5, 2.5	35.6	78.6	[52],[a]

[†] Estimated using $\Delta V = 300 m/s$ and $I_{SP} = 300 s$

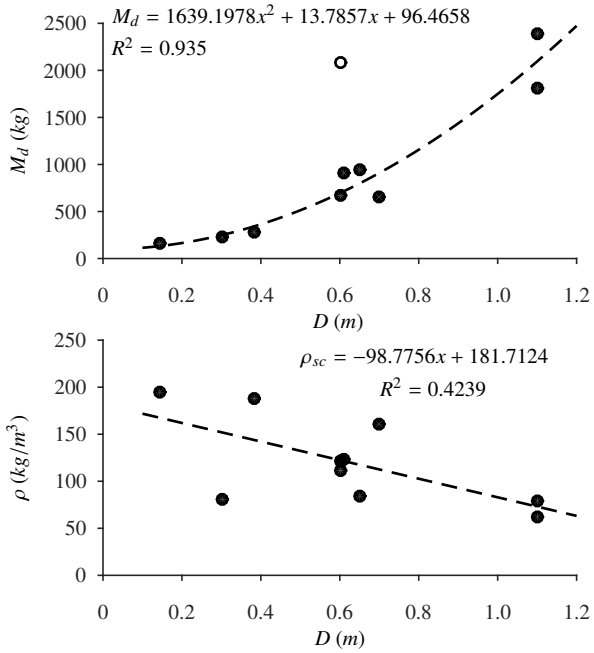

Figure 4. Parametric relations for spacecraft dry mass and stowed density as a function of aperture diameter

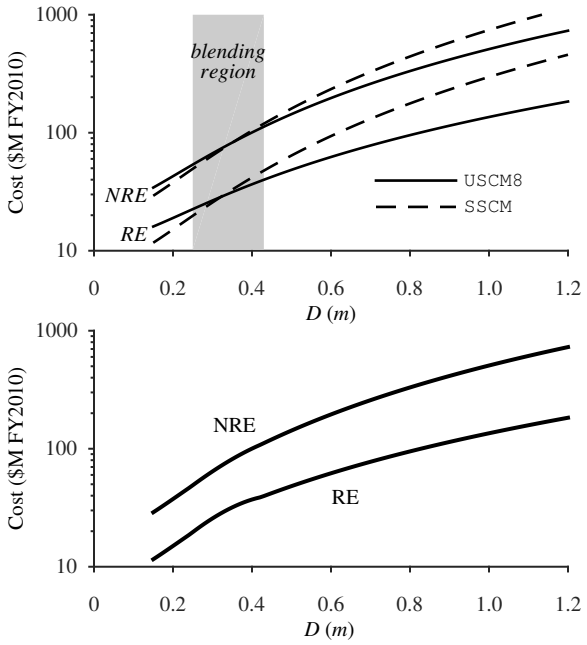

Figure 5. The spacecraft cost model for both NRE and RE is a blended combination of two cost models. The SSCM and USCM8 cost models cross over within the blending region yielding a smooth blended model

Table 2. Assumed properties of selected U.S. launch vehicles

Launch Vehicle	Mass to LEO† kg	Cost $M (FY10)	Payload Volume m^3
Pegasus XL	443	30	1.87
Athena Ic	700	41	14.5
Minotaur IV	1650	50	11.4
Falcon 9 v1.1	10450	56.7	146
Falcon Heavy	53000	100	146

† Payload mass to 28.5° inclination, 200km

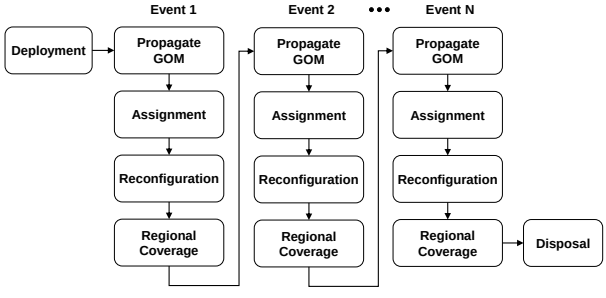

Figure 6. Campaign-based life cycle simulation

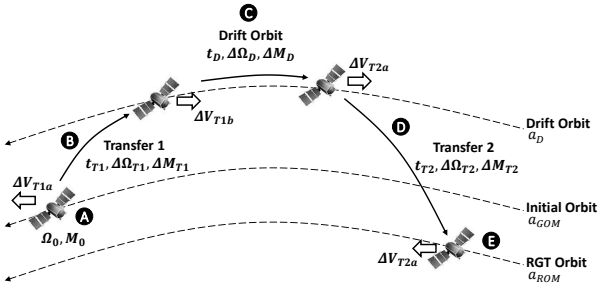

Figure 7. The satellite reconfiguration strategy involves two in-plane Hohmann-transfers: one to move the satellites into a drift orbit for faster phasing with the desired ROM orbital slot (labeled B), and one to move the satellites into ROM (labeled D).

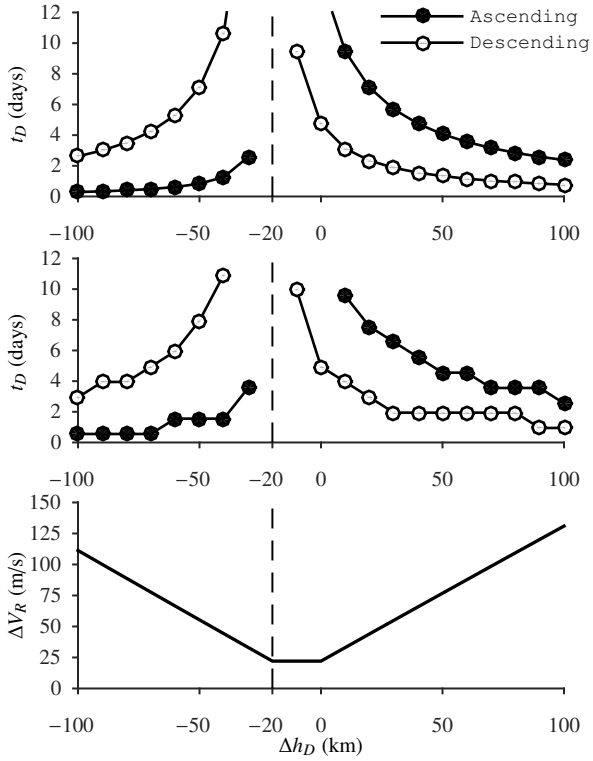

Figure 8. Drift Time (t_D), time to first pass (t_{fp}) and ΔV_R as a function of Δh_D for ascending and descending pass RGT orbits

Table 3. Reconfigurable architecture design variables and bounds

#	Variable Name	Symbol	Type	Bounds
x_1	RGT type	N_o/N_d	cat.	$\frac{31}{2}, \frac{15}{1}, \frac{29}{2}, \frac{14}{1}, \frac{27}{2}, \frac{13}{1}$
x_2	GOM altitude offset	Δalt	cont.	−50 to 50 km
x_3	# orbit planes	N_p	int.	1 to 36
x_4	# satellites per plane	N_{sp}	int.	1 to 24
x_5	Inclination	i	cont.	50° to 130°
x_6	Phasing parameter	F	int.	0 to $N_p - 1$
x_7	Aperture size	D	cont.	0.1 to 1.2 m
x_8	ReCon ΔV	ΔV_{recon}	cont.	0 to 1000 m/s
x_9	Decision model weight	α_0	cont.	0 to 1

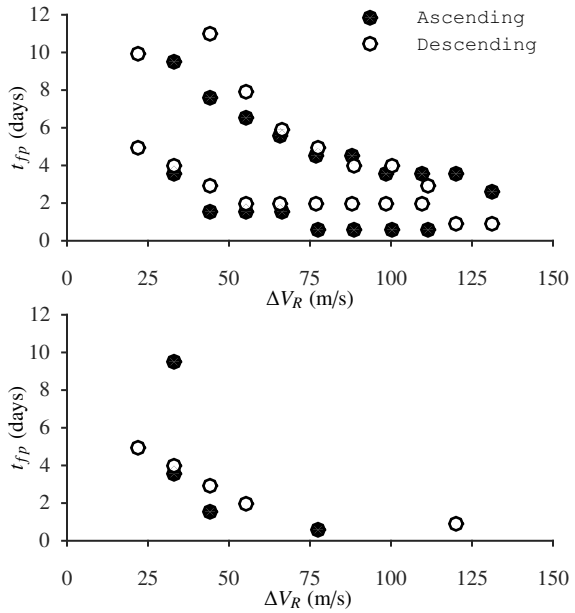

Figure 9. Tradeoff between minimizing time to first pass and minimizing reconfiguration propellant usage

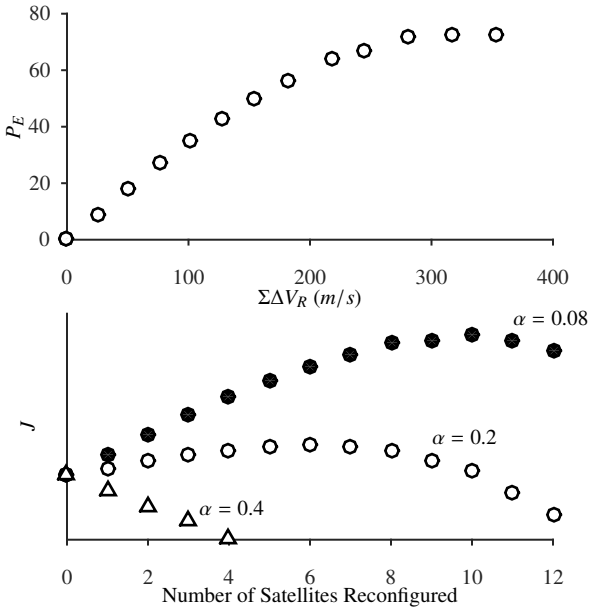

Figure 10. The decision model weight directly affects how many satellites should be reconfigured to minimize the objective J. In this example, $\alpha = 0$ results in zero satellites reconfigured while $\alpha = 0.08$ results in 10 satellites reconfigured.

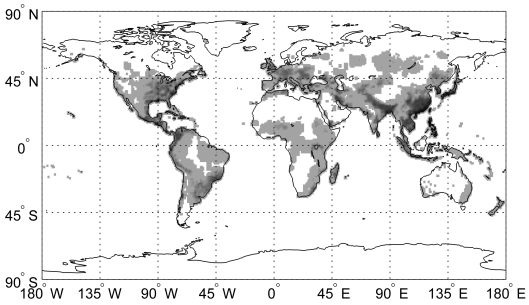

Figure 11. Natural disaster risk PDF with increasing probability indicated by darker shading

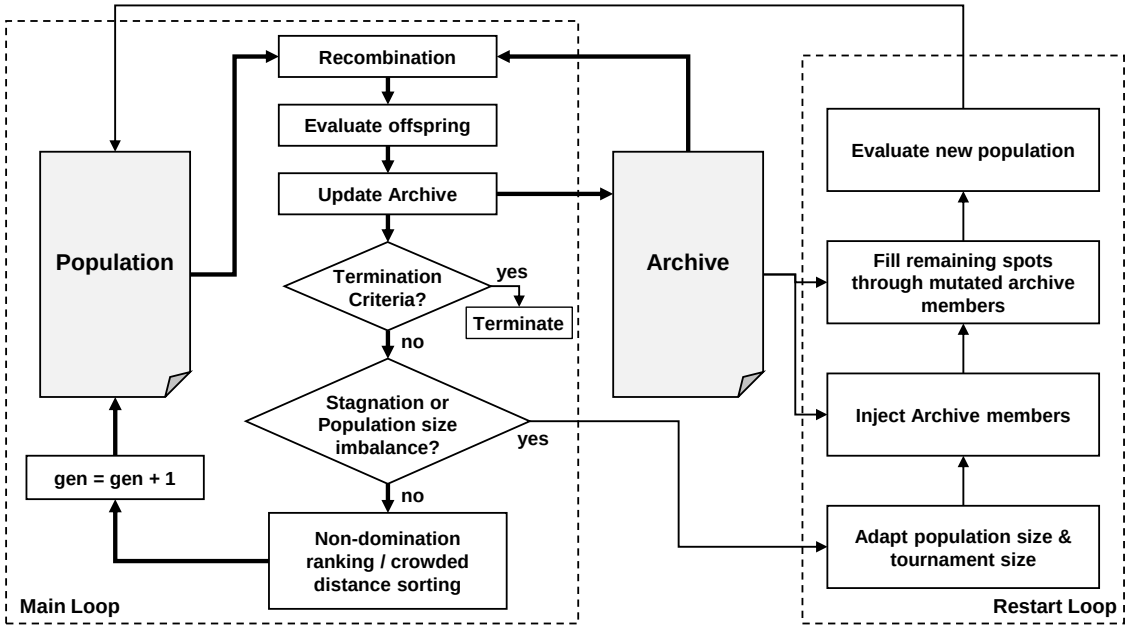

Figure 12. Modified ϵ-NSGA-II optimization process

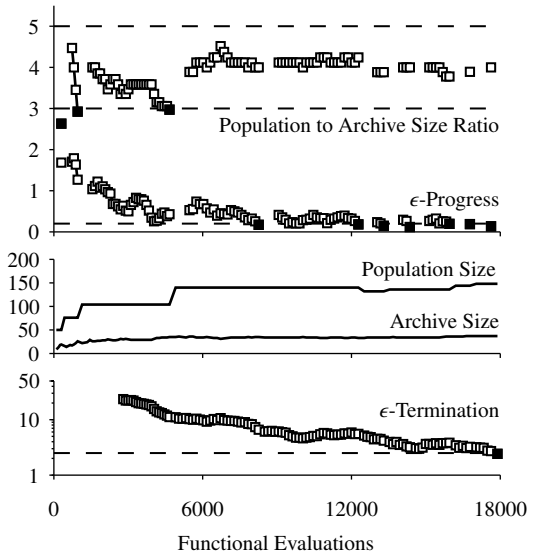

Figure 13. Symmetric pattern scenario 1 optimization run data and convergence

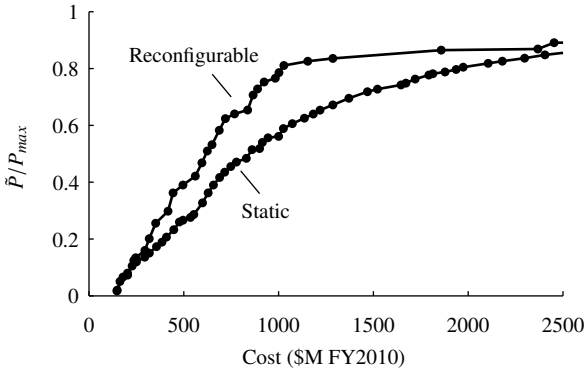

Figure 14. The reconfigurable non-dominated front completely dominates the static non-dominated front in terms of maximizing performance while minimizing cost

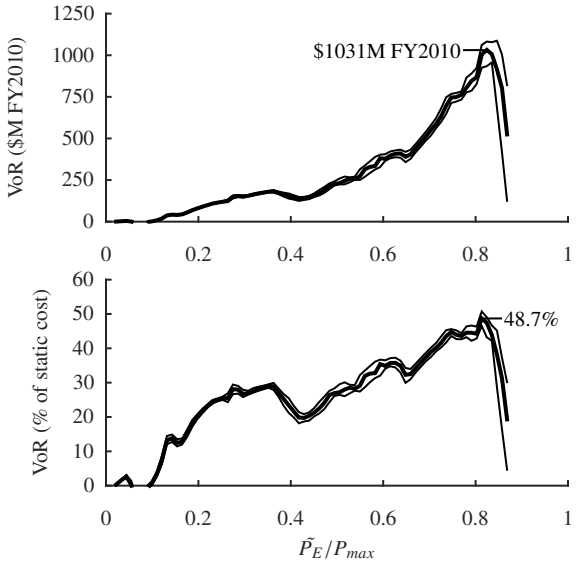

Figure 15. The value of reconfigurability, which is equal to how much cheaper the reconfigurable designs are when compared to iso-performance static designs, is 20-50% of the static architecture cost

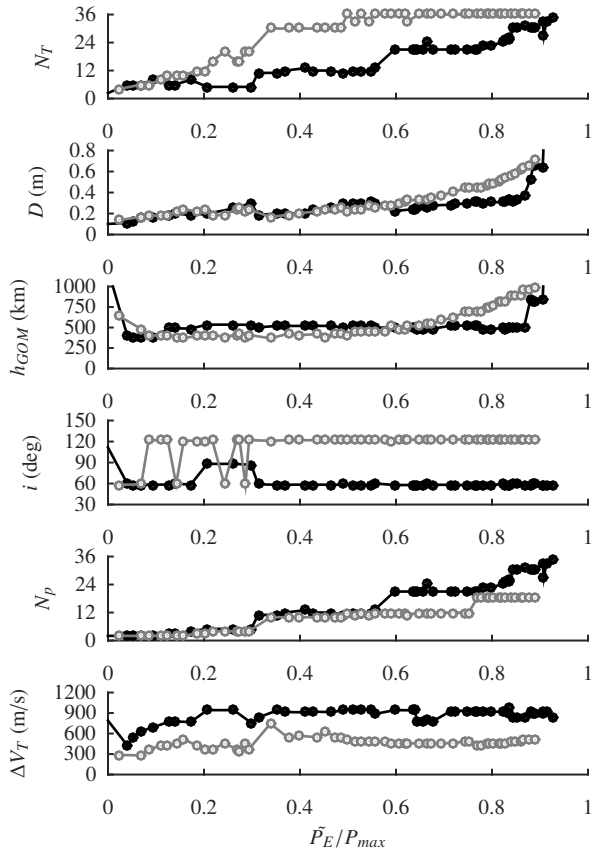

Figure 16. Design details for the reconfigurable (colored black) and static (colored gray) non-dominated fronts

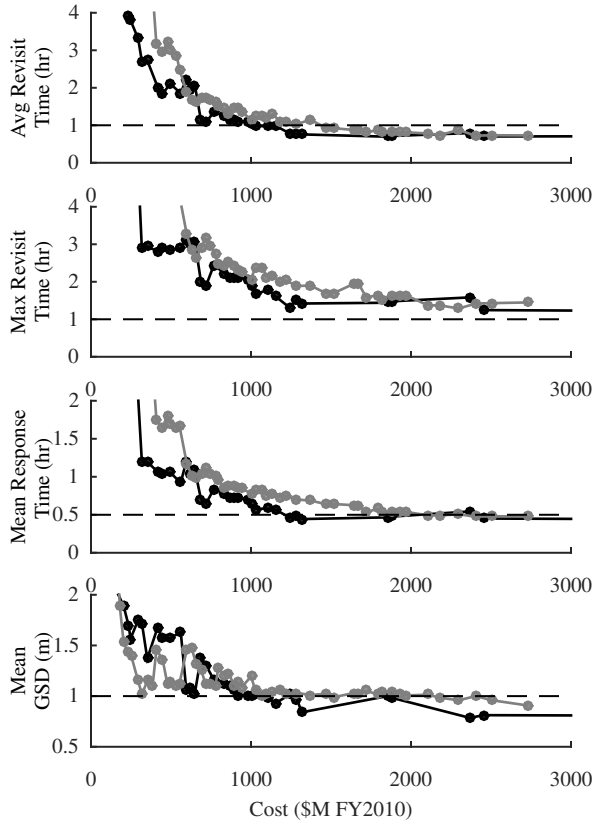

Figure 17. When compared with traditional figures of merit, non-dominated reconfigurable designs (colored black) generally outperform iso-cost non-dominated static designs (colored gray)

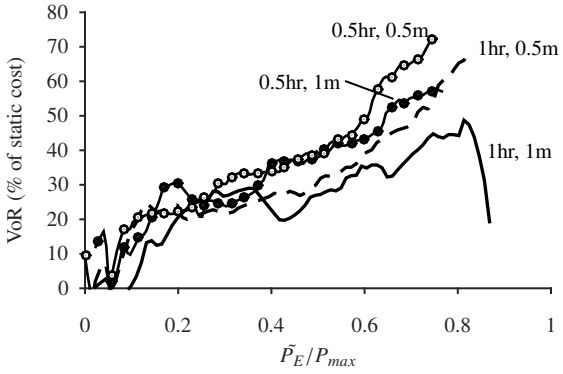

Figure 18. The value of reconfigurability increases with increasing coverage requirements