SSC20-S2-07

Hybrid Architecture Performance and Evaluation for Quantitative and Comparative Analysis

Matthew Stein, Pete de Graaf, Sean Kinser, John Muhonen, Robert Roller
The MITRE Corporation
1155 Academy Park Loop, Colorado Springs, CO 80910, 719-572-8200
mstein@mitre.org

David Voss, Amanda Salmoiraghi, Kyle Kemble
United States Space Force
7250 Getting Heights, Colorado Springs, CO 80916

ABSTRACT

As space becomes an increasingly heterogeneous blend of multi-national commercial, civil, and government systems, additional capability can be leveraged by taking a hybrid approach to space-based services. The choice of service provider must be weighed against requirements such as timeliness, quality, and confidence in the results. We present a method to quantitatively evaluate the overall performance of any space architecture from traditional monolithic systems to fully hybrid systems-of-systems that blend contributions from multiple providers with distinct capabilities. The results can inform operational, planning, and acquisition decisions for both current and future space missions.

INTRODUCTION

Recent years have shown skyrocketing numbers of satellite deployments with commercial companies taking an ever-increasing presence in space. A small satellite with state-of-the-art capabilities can be built and launched for as low as a few tens of thousands of dollars.[1] This makes space accessible to small businesses, small research groups, and even private individuals. Commercial companies are now offering space-based services with competitive pricing and capabilities, making the decision to build or buy even more complex. With the emerging abundance of vendors on the market, an ideal space architecture may involve both options, including an aggregation of services from multiple providers.

Factors to consider when designing a space-based architecture include coverage, capabilities, redundancy, vulnerabilities, and the potential risks associated with using a hybrid blend of vendors and assets. We present a unique approach to the evaluation of architecture performance which includes these attributes, and a means of optimization across these trades, now and for future acquisitions. We also consider and quantize the potential benefits of combining multiple sources of information for particular missions and measure the effect on resilience and overall capability.

EVALUATION CRITERIA

The overall performance of an architecture must be evaluated based on the use cases and requirements. The use cases will specify the services which the architecture will be required to provide, such as optical imagery at one meter resolution within two hours for a given location. Additionally, to perform a broad set of analyses across large geographic areas, it is useful to perform timing and coverage calculations first. With these calculations performed at sufficient precision, it is possible to generate statistical coverage values for all locations of interest in the form of a timing heat map.

Coverage & Timing Generation

To generate a timing heat map, all space vehicles under consideration must be propagated through

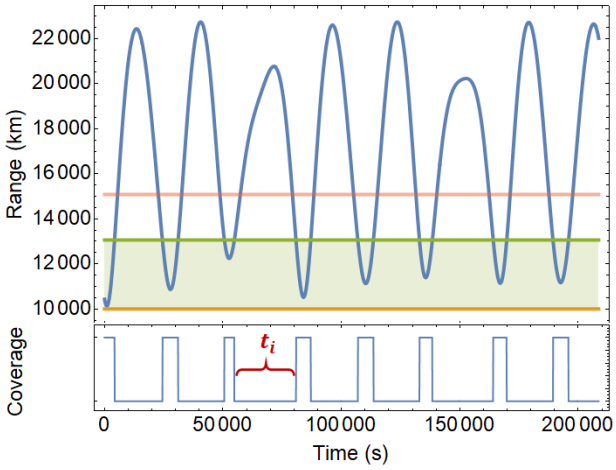

Figure 1: Example range and coverage of a particular target location from a single satellite. The blue curve represents the absolute range. The orange line represents the minimum sensing range which is approximately the altitude of the satellite. The green line represents the maximum sensing range possible which is reduced from the maximum line-of-sight range shown by the red line. The difference between the red and green lines is due to the minimum sensing elevation angle of 20° for this example. The shaded green area represents the boundaries of possible sensing coverage. A representative time gap t_i is indicated by the red bracket.

numerical simulations with coverage calculated for all locations of interest. The objective is to determine the mean time between revisits (MTBR, $\boldsymbol{\Delta t}$) for each site. This value is unique to each vehicle and each location but can be applied to a wide variety of analyses as will be shown in the following sections.

As an illustrative example, consider a satellite with an altitude of 10,000 km, an inclination of 45°, and a minimum sensing elevation angle of 20° as viewed from the ground. For an example target location with latitude 16.7288°, longitude 165.5398°, and altitude of 0 m above sea level, the range and coverage are illustrated in Figure 1. The simulation lasted 10 orbital periods, with time step size $t_{\text{step}} = 500\,\text{s}$, and the maximum sensing range was approximately 13,061 km. The coverage is represented as a step function where 1 indicates coverage and 0 indicates a lack of coverage. The rising and falling edges of the step function are considered moments of coverage and lack of coverage, respectively.
With the coverage function established, the N periods of time without coverage t_i can be calculated and assembled into a list.

$$\{t_1, t_2, t_3 \ldots t_N\} \tag{1}$$

Assuming a request for sensing was made at every time step in the simulation, there would be a delay for coverage at each moment during any particular time gap t_i. It is possible to quickly sum all potential delays directly from the gap times themselves, without tracking each individual delay:

$$t_{\text{AllDelays}} = \sum_{i=1}^{N} \left[\frac{t_i}{2} \left(1 + \frac{t_i}{t_{\text{step}}} \right) \right]$$

The MTBR is then:

$$\boldsymbol{\Delta t} = \frac{t_{\text{AllDelays}}}{(T/t_{\text{step}})}$$

where T is the total simulation time. The variance and standard deviation can likewise be calculated without tracking each individual delay. Consider Q_i to be the number of possible delay times held within some gap time t_i:

$$Q_i = \frac{t_i}{t_{\text{step}}}$$

The variance for all possible delay times is:

$$\sigma^2_{\boldsymbol{\Delta t}} = \frac{1}{(T/t_{\text{step}}) - 1} \sum_{i=1}^{N} \Bigg[f(Q_i)\, t^2_{\text{step}} + (t_i - \boldsymbol{\Delta t}) \left[(t_i - \boldsymbol{\Delta t}) Q_i - g(Q_i)\, t_{\text{step}} \right] \Bigg] \tag{2}$$

where $f(Q_i)$ and $g(Q_i)$ are coefficients defined by:

$$f(Q_i) = \frac{Q_i^3}{3} - \frac{Q_i^2}{2} + \frac{Q_i}{6}$$
$$g(Q_i) = Q_i (Q_i - 1)$$

For this example satellite and target location, the mean time to sensing is approximately $8,400 \pm 6,100$ seconds.

Computational Speedup

The choice of t_{step} directly affects the quality of the simulation with lower values of t_{step} yielding more accurate results. However, this also leads to slower simulations and typically greater memory consumption, especially if delay times were tracked individually at every time step.

By only tracking the coverage gaps t_i, memory usage and I/O are significantly reduced. In this example where t_{step} was 500 seconds, the memory and I/O were reduced approximately 43 fold as compared to tracking every possible delay time individually. The reduction in memory usage and I/O scales linearly with the choice of timing precision (t_{step} of 50 seconds would yield a ∼430 fold reduction, and so on). To be clear, it is the calculation of the MTBR that directly benefits from this approach, not the orbital propagation or overall simulation itself.

Translating the reduction of memory usage, I/O, and mathematical operations to time saved will depend on the specific hardware used for calculations. Critical factors include processor cache size, memory size, and memory bandwidth. The reduction in I/O alone, however, makes this approach much more amenable to parallel processing across many cores, which itself can yield significant speedup. The calculation of variance directly from Equation 2 yields a similar reduction in the number of mathematical operations required, as compared to tracking all possible delay times.

Simulations may require modeling tens of thousands of satellites, and each satellite must have coverage calculated for each location of interest. The speedup attained here makes global scale MTBR calculations feasible for even high precision values of t_{step}.

Ground Link Delays

If data links to the ground are continuous, either from cross-linked networks or continuous ground site coverage, then any uplink or downlink time can be simplified as the mean data product size divided by the mean network speed. If this is not the case, however, then the mean time to uplink and downlink must be calculated separately.

The mean uplink time is calculated with the same procedure as the MTBR, but the coverage analysis must include all possible ground stations simultaneously. The mean downlink time should consider the gaps in ground station coverage u_i, similar to Equation 1:

$$\{u_1, u_2, u_3 \dots u_M\}$$

where M is the total number of gaps. The mean downlink time is then

$$\boldsymbol{\Delta t}_{\text{DL}} = \frac{\sum_{i=1}^{M} u_i}{M + (S/t_{\text{step}})} \tag{3}$$

where S is the total time of simultaneous ground site and target coverage, which can be calculated from inspection of the t_i and u_i gap times as illustrated in Figure 2.

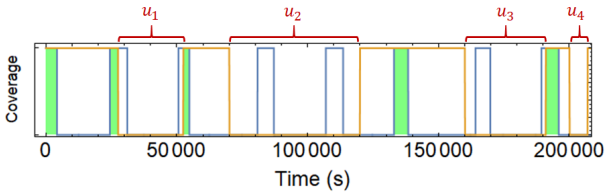

Figure 2: Coverage over the target (blue curve) and all ground stations (yellow curve). The gaps in ground station coverage are labeled by u_i and highlighted with red brackets. The periods of simultaneous coverage are highlighted in green and the widths of those periods contribute to S in Equation 3.

Use Case Information

To extract useful performance metrics, a number of descriptive and comprehensive use cases must be considered and quantized. For example, a space architecture that provides imaging services could be compared against demands for images across a variety of resolutions, time requirements, and tolerable risk.

In the case of optical imaging, the Rayleigh criterion could be used as a proxy for imaging capabilities, and is easily calculated for assets with known optical properties.[2] Alternatively, vendor-provided specifications could be used given that image processing techniques can reveal useful information beyond what the Rayleigh criterion would indicate.[3, 4] Radio frequency geolocation services could be quantized in terms of the location accuracy, and similar quantization may be defined for any other phenomenology of interest.

A timeliness requirement should specify the desired time from request to data delivery. This value must include the mean time to sensing as described earlier, but also the expected time to uplink the task preceding the sensing window, and the expected time to process and downlink data after sensing.

The risk metric must be a comprehensive view of the security of the ground and space assets, the data, and the transport means. The Architecture Score Index (ASI) is used to quantify this by starting with

the recently developed Cybersecurity Maturity Model Certification standards and expanding this to include items such as patch cadence, incidence response time, and company/agency past performance[5, 6]. ASI values can range from zero to one with zero indicating no risk and one indicating the highest possible risk. ASI values can be calculated on a per-vendor basis, but ideally the value would change based on the vendor, hardware, and communication paths used in any particular data or sensing request.

PERFORMANCE ANALYSIS & RESULTS

With the architectures and use cases defined, and timing heat maps generated, it is possible to quantitatively compare one architecture to another. The comparative quantities include the MTBR, coverage and redundancy values, performance versus use cases, and redundancy analysis. This provides the power to assess the individual merits of independent architectures as well the benefits gained by combining them. Additionally, this analytical approach allows the creation or optimization of an architecture against some set of desired performance metrics.

Example Architectures

Consider two unique space architectures that each provide imaging services, Architecture A and Architecture B. Both are owned and controlled by independent vendors and have continuous communication with their ground network.

Architecture A is composed of five assets: two assets in geosynchronous orbit, two assets at an altitude of 21,500 km and inclined at 55°, and one asset at an altitude of 700 km with an inclination of 67°. Architecture B is also composed of five assets: all assets have an altitude in the range of 430–570 km; two assets are inclined 45°, two are inclined 97°, and one is inclined 65°.

All assets are capable of optical imaging and use the Rayleigh criterion as the basis for resolution. For Architecture A, the low Earth orbiting asset is capable of 1 m resolution, and all others provide 5 m resolution. For Architecture B, three assets provide 1 m resolution, and the remaining are capable of 1.5 m resolution. All assets in Architecture A(B) have an ASI of 0.30(0.55). What follows is an assessment of the performance of each architecture independently as well as the improvement when combined.

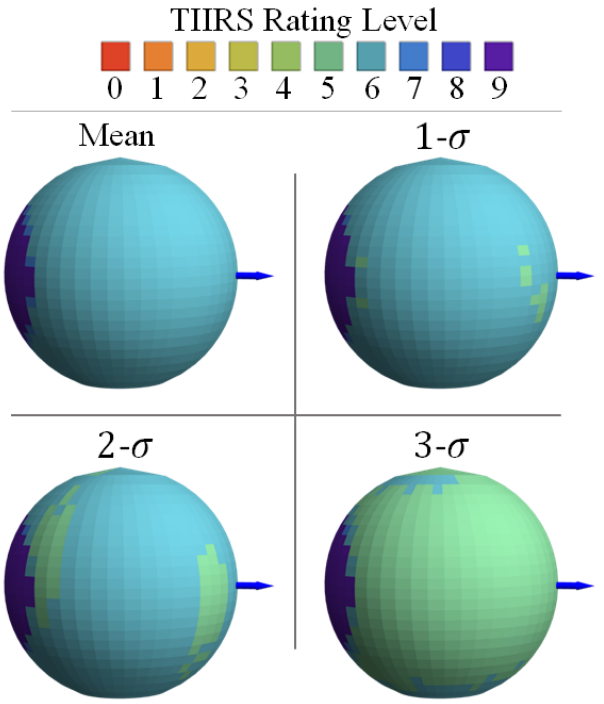

Figure 3: TIIRS rating by location for Architecture A, per confidence upper limit (CUL). The blue arrow marks Null Island at 0° N 0° E. The values can degrade as the CUL increases, making the standard deviation of the MTBR a critical piece of overall assessment.

Timing & Coverage

The Temporal Imagery Interpretability Rating Scale (TIIRS) has been proposed as a convenient means to characterize revisit times over a specific location.[7] The TIIRS ratings used in this analysis are given in Table 1 and each architecture is mapped against these values. The TIIRS rating is calculated from the MTBR with smaller values receiving higher scores. However, the TIIRS rating received by each architecture is relative to the confidence interval of interest. Therefore, the MTBR should be added to the appropriate multiple of the standard deviation, which may negatively impact the TIIRS rating as shown in Figure 3.

In this analysis, Architecture A showed degradation from TIIRS 6 to TIIRS 5 in many regions when moving to the 3-σ confidence upper limit. As a result, the operator of Architecture A could not promise daily imaging in those regions at the 3-σ confidence level. Put another way, the operator could not promise 99.7% daily reliability in imaging those regions.

Table 1: The TIIRS rating levels associated with revisit times less than or equal to those shown. Adapted from Ref. [7] and modified for this analysis.

TIIRS Level	$\Delta t \leq$
0	∞
1	1 century
2	1 decade
3	1 year
4	1 month
5	1 week
6	1 day
7	1 hour
8	1 minute
9	1 second

When designing or choosing a blend of space architectures, it is critical to understand how the assets complement one another. In this example, Architecture A initially showed global coverage at TIIRS 6 or above when looking only at the MTBR. When moving to the 3-σ confidence upper limit, only ~33 % of the globe maintains TIIRS 6 or higher, as shown in Table 2. However, when leveraging the combination of both Architecture A and B, the 3-σ confidence upper limit is maintained at ~78 % for TIIRS 6 or higher across the globe. This is shown in Table 3 and illustrated in Figure 4. The combined architecture maintains the TIIRS 9 persistence over certain regions from the geosynchronous assets in Architecture A as well as the mid-latitude TIIRS 6 coverage from Architecture B.

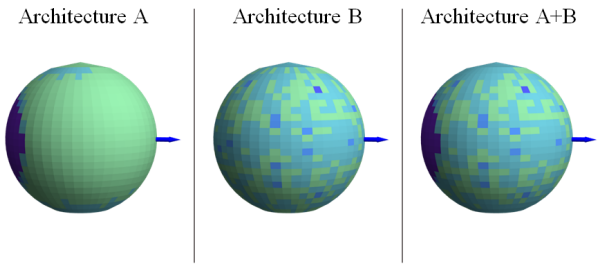

Figure 4: The 3-σ confidence upper limit of TIIRS ratings for Architecture A, Architecture B, and a combination of both architectures.

Table 2: The approximate percent of the world covered by TIIRS rating for Architecture A, rounded to whole numbers.

CUL	% of Earth covered at TIIRS rating of:									
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Mean	0	0	0	0	0	0	81	3	<1	16
1-σ	0	0	0	0	0	2	81	1	<1	16
2-σ	0	0	0	0	0	8	74	1	<1	16
3-σ	0	0	0	0	0	66	17	1	<1	16

Table 3: The approximate percent of the world covered by each TIIRS rating per architecture at the 3-σ confidence upper limit. Results are rounded to whole numbers.

Arch	% of Earth covered at TIIRS rating of:									
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
A	0	0	0	0	0	66	17	1	<1	16
B	0	0	0	0	0	39	58	3	<1	0
A+B	0	0	0	0	0	22	58	3	<1	16

Table 4: The percent of the globe covered by at least N assets with a TIIRS rating of 6 or higher.

Arch	% of Earth covered by N assets:					
	0	**1**	**2**	**3**	**4**	**$\geq$ 5**
A	0	0.2	16.8	62.4	19.1	1.6
B	0.1	1.7	16.5	29.9	20.5	31.3
A+B	0	0	0	0.5	9.3	90.1

Another important factor is the redundancy of coverage per site. This can be viewed as the number of assets that cover a particular site with any TIIRS rating, a particular TIIRS rating, or a minimum TIIRS rating. As shown in Table 4, the combination of architectures gives substantially higher redundancy values with no single location having a redundancy of less than three assets with a TIIRS rating of six or higher.

Use Case Analysis

To assess performance against a set of use cases, each architecture is tasked with 10,000 optical imaging requests randomly spaced across the Earth. Each request contains a location of interest along with timeliness, resolution, and ASI requirements. The distribution of these metrics is illustrated in Figure 5. The MTBR is used to determine if an asset is available within the specified time. If an architecture has at least one asset that meets all requirements for the location of interest, it is considered a solution to the request.

There can be zero, one, or multiple solutions per request. The quality of the solutions can be ranked by the magnitude of a vector defined by the timeliness, resolution, and ASI of each. Lower magnitudes indicate better solutions, and these can be normalized against the vector defined by the request's requirements.

After collecting all possible solutions to all requests, Architecture A(B) was able to provide solutions to 66 %(45 %) of the requests, whereas the combination of both architectures provided solutions to 74 % of all requests. By itself, however, this value is insufficient to describe the overall performance as it fails to reveal the values of the individual metrics involved.

It is possible to calculate the mean performance against any particular metric as shown in Table 5. Whereas both architectures show roughly similar timing metrics, Architecture B has significantly better resolution but with much poorer ASI values, indicating that the majority of options fail to qualify as a solution due to the ASI requirement.

Table 5: The normalized mean metrics for each architecture against the set of requests. A score of 1 indicates meeting the requirement exactly, whereas values below 1 indicate better performance than required, and values above 1 indicate failing to meeting the requirement.

Arch	Time	Resolution	ASI
A	0.58 ± 0.44	0.72 ± 0.47	0.69 ± 0.37
B	0.60 ± 0.46	0.30 ± 0.24	1.26 ± 0.68
A+B	0.59 ± 0.45	0.48 ± 0.41	1.02 ± 0.64

After normalizing all solutions to the requirements of each request, the performance can be defined as

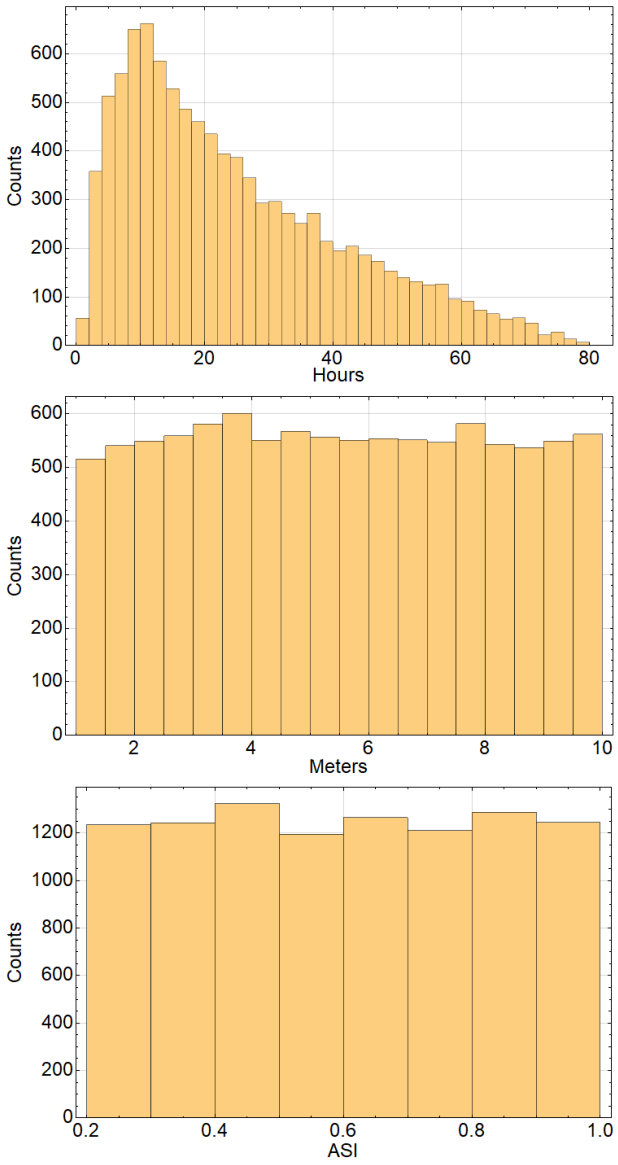

Figure 5: The distribution of time, resolution, and ASI requirements over the simulated 10,000 requests.

a shape within the n-dimensional space defined by the number of requirement metrics per request. The center point of the shape is located at the mean value per metric, and the volume is spanned in every direction by the associated standard deviation. In this example, the architectures span an ellipsoid in a 3-dimensional space as illustrated in Figure 6.

The volume of the shape is defined by the confidence level of interest. A critical performance metric for each architecture is the percent of the volume of the ellipsoid inside the box defined by the normalized

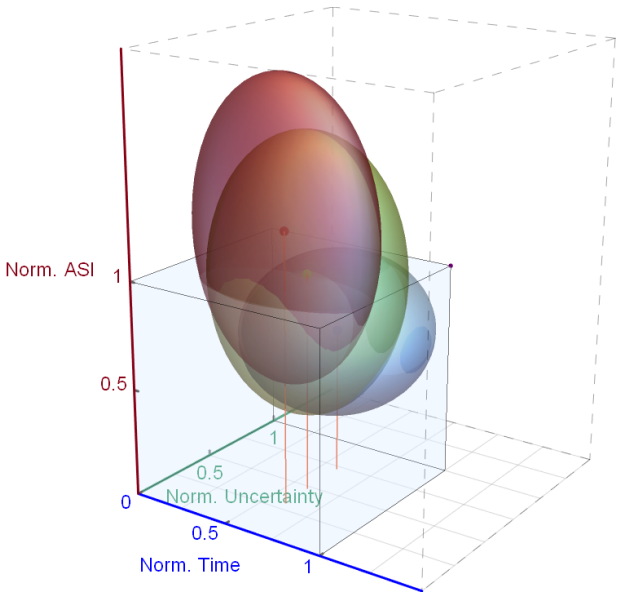

Figure 6: The performance ellipsoids at the 3-σ confidence interval for Architecture A (blue), Architecture B (red), and the combination of both architectures (green). The center point and span of the ellipsoids is given in Table 5.

Table 6: The performance of each architecture as measured by the volume of the ellipsoid defined by the values in Table 5 and by confidence interval (CI).

Arch	Performance (%) at CI of:		
	1-σ	**2-σ**	**3-σ**
A	87.9	14.9	5.2
B	23.1	3.9	1.5
A+B	46.8	8.7	2.7

request. This box spans from (0, 0, 0) to (1, 1, 1) in this case. As the confidence level increases, the overall performance of each architecture generally decreases as shown in Table 6. This is a good proxy for how often an architecture will meet all of the requirements of each request at the associated confidence interval. Compared to Architecture A, Architecture B shows an overall poor performance given that the center point of the ellipsoid is outside of the request boundaries. This indicates that, on average, the Architecture B fails to meet the requirements of the requests. This is primarily due to an ASI value that is generally unacceptable for each request.

Aggregating Sources of Information

It may be useful to combine multiple sources of information to create a single solution. For example, multiple images of the same site at approximately the same time would, in principle, increase the trust in the data products related to the images, provided the images correlated the same data. By combining sources of information in this way, the ASI value in the solution should decrease, reflecting a reduction in risk as opposed to trusting only a single source of information.

The total ASI value from multiple sources is defined as:

$$\mathrm{ASI}_{\mathrm{Total}} \equiv \prod_{i=1}^{N} \prod_{j=1}^{C_i} K^{j-1} \mathrm{ASI}_{ij} \tag{4}$$

where N is the number of unique vendors in the potential solution, C_i is the number of contributions from architecture i, and ASI_{ij} is the ASI value for architecture i and contribution j. The quantity K is a diminishing return factor to favor combinations from multiple vendors as opposed to the same vendor. K must be greater than or equal to one.

When assessing the overall performance of any solution from multiple sources, the timeliness should be the highest MTBR and the highest resolution (lowest quality) from any contribution. The ASI value is determined from Equation 4. Using this method, the performance of every architecture improves as up to two or three sources of information are allowed per request. The overall performance metrics for these cases are shown in Table 7 and Table 8, respectively. The performance is likewise illustrated in the ellipsoids in Figure 7.
While the number of potential contributions to a solution is limited only by the number of assets available, practical limitations should be considered. Analysts may have limited time to conduct data extraction, or costs could be a constraining factor. In this example, diminishing returns for performance were observed when allowing more than three sources of information per request.

Solution Redundancy Analysis

To measure how an architecture performs under high demand, or in a degraded state, it is useful to examine how many solutions were possible for each given request. As illustrated in Figure 8, the number of options available increases significantly as more sources

Table 7: The normalized mean solution metrics for each architecture against the set of requests, allowing for more than one contribution per solution.

Arch	Time	Resolution	ASI
Max Sources Per Solution = 2			
A	0.61 ± 0.44	0.63 ± 0.47	0.48 ± 0.37
B	0.65 ± 0.47	0.29 ± 0.23	1.00 ± 0.63
A+B	0.65 ± 0.46	0.37 ± 0.34	0.65 ± 0.53
Max Sources Per Solution = 3			
A	0.61 ± 0.44	0.63 ± 0.47	0.48 ± 0.37
B	0.67 ± 0.48	0.29 ± 0.22	0.88 ± 0.62
A+B	0.67 ± 0.46	0.34 ± 0.31	0.56 ± 0.50

Table 8: The performance of each architecture as measured by the volume of the ellipsoid defined by the values in Table 7 and by confidence interval (CI).

Arch	Performance (%) at CI of:		
	1-σ	2-σ	3-σ
Max Sources Per Solution = 2			
A	95.7	21.4	6.7
B	47.6	8.9	3.3
A+B	88.3	22.1	6.6
Max Sources Per Solution = 3			
A	95.7	21.4	6.7
B	59.8	12.0	4.6
A+B	93.4	26.1	7.5

of information are allowed to contribute to a solution. Furthermore, the number of requests with at least one solution increased for Architecture A(B) from 66 %(45 %) to 73 %(79 %) when considering one to three contributions per solution. The combined architecture likewise increased from 74 % to 88 %. This is shown in the reduction of the first column of the histograms in Figure 8, which represents the number of times zero solutions were found.

Optimization

Rather than analyzing a set of predefined architectures, it is useful to instead start with performance metrics which will be used to design an ideal architecture. Given these metrics, a Monte-Carlo search can be carried out over a wide range of potential architectures, discovering which combinations of assets yield the best overall performance. Coupled with information like cost-to-build versus cost-of-service, this provides a useful means of identifying the optimal system-of-systems for a given set of requirements.

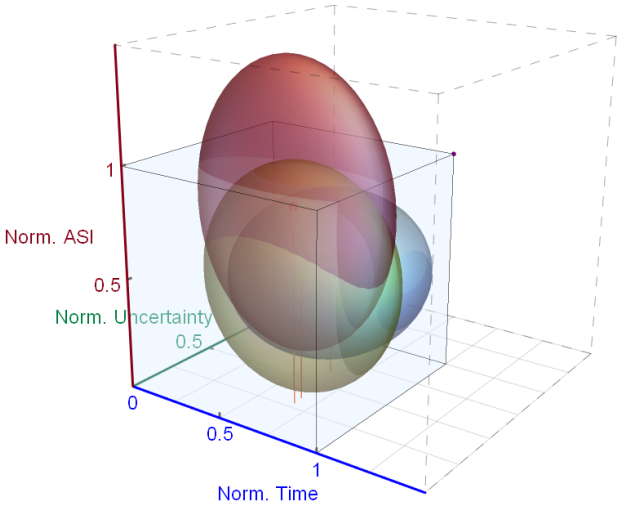

Figure 7: The 3-σ confidence upper limit of TIIRS ratings for Architecture A (blue), Architecture B (red), and a combination of both architectures (green) when allowing three contributions per solution.

In the example analysis in this paper, it is seen that adding only five low Earth orbiting assets to Architecture A improves the number of requests with solutions from 66 to 73%. This analytical method could be used to determine the number of assets required, in any mix of orbits, to obtain the desired results. For example, if 90% request satisfaction was required and only two sources of information could be considered per request, 15 satellites like those in Architecture B would be required to add to Architecture A.

The analytical results also allow for considerations of flexibility in requirements. For example, the worse metric in Architecture B is the ASI performance. An adjustment can be made to the ASI requirements to reassess performance, and then determine if the additional capability is worth the trade in ASI, or any other metric. For example, when changing the ASI metrics from the range of $[0.2, 1.0]$ as used earlier to $[0.3, 1.0]$, Architecture B alone provides solutions to more requests than Architecture A (76% versus 75%), when allowing for up to two sources per request. The gap grows even larger as more sources of information are allowed per request.

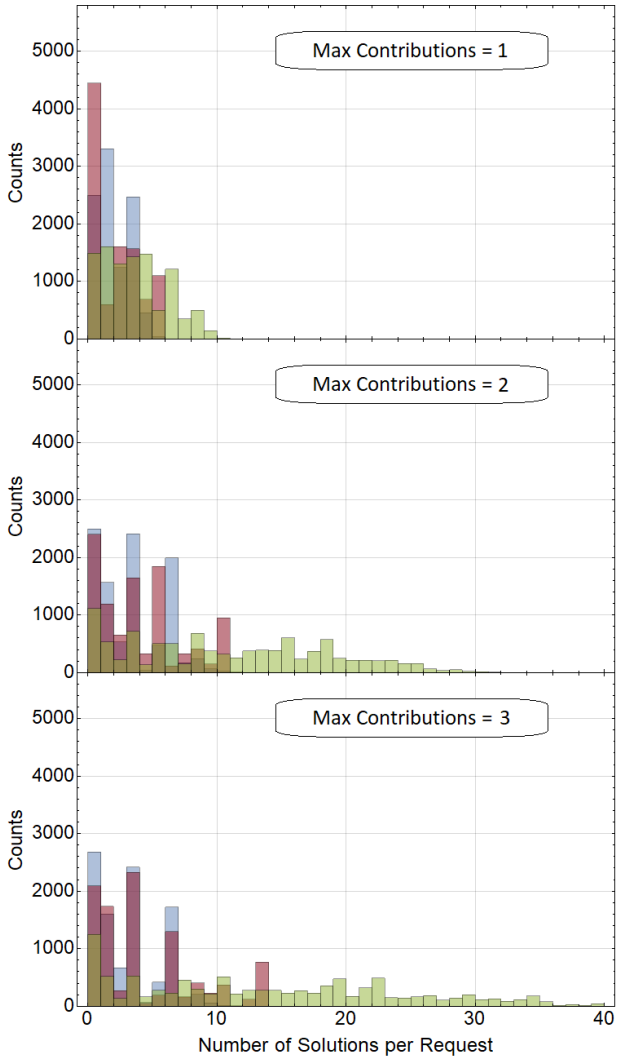

Figure 8: Number of times a request had N solutions, by architecture, and when allowing for multiple sources of information per request. Greater redundancy is indicated by the distribution moving further to the right. Blue, green, and yellow bars indicate Architectures A, B, and the combination of both, respectively.

CONCLUSION

Space architectural design is a broad and complicated field. There are near infinite possibilities in choosing where and how to place assets in space, and with what capabilities to achieve a mission or set of requirements. Presented here is a new, comprehensive approach to assessing the overall performance of space architectures which includes timeliness, quality, trust, and redundancy metrics as well as a method of assessing contributions from multiple sources of information. The result is the ability to model and assess a very large number of architectures on a global scale, and with high computational efficiency.

This model is extensible to other phenomenologies and modalities not covered here. Certain criteria within the model can also be adjusted based on the priorities of each requirement. For example, the solutions in this analysis were ranked based on the magnitude of the vector defined by the timeliness, resolution, and ASI, but a different method could be applied where timeliness is weighted more heavily than the other values. Furthermore, the model can consider timing metrics for path-agnostic, cross-linked communications. This would increase overall time-based performance, but would also require the ASI value to be updated per path or calculated in the mean given all possible paths. This will be investigated and presented in a future analysis.

The analytical method presented here can yield optimized architectures given a set of requirements. The identified optimal architectures should then be assessed with more deterministic modeling techniques using specific mission scenarios. The feedback from those simulations could be used to modify this approach by tweaking performance metrics and repeating the process. The final result is a well-defined, optimized architecture that has quantized metrics to substantiate it as the architecture of choice.

Acknowledgments

This technical data was produced for the U. S. Government under Contract No. FA8702-19-C-0001, and is subject to the Rights in Technical Data-Noncommercial Items Clause DFARS 252.227-7013 (FEB 2014).

References

1. Jeremy Straub, Thyrso Villela, Cesar A. Costa, Alessandra M. Brandão, Fernando T. Bueno, and Rodrigo Leonardi. Towards the thousandth cubesat: A statistical overview. *International Journal of Aerospace Engineering*, 2019(5063145), 2019. `https://doi.org/10.1155/2019/5063145`.

2. Max Born, Emil Wolf, A. B. Bhatia, P. C. Clemmow, Dennis Gabor, Alexander Rawson. Stokes, A. M. Taylor, P. A. Wayman, and W. L. Wilcock. *Principles of optics: electromagnetic theory of*

propagation, interference and diffraction of light. Cambridge University Press, 2018.

3. John G. Walker. Optical imaging with resolution exceeding the rayleigh criterion. *Optica Acta: International Journal of Optics*, 30(9):1197–1202, 1983. https://doi.org/10.1080/713821367.

4. Changtao Wang, Dongliang Tang, Yanqin Wang, Zeyu Zhao, Jiong Wang, Mingbo Pu, Yudong Zhang, Wei Yan, Ping Gao, and Xiangang Luo. Super-resolution optical telescopes with local light diffraction shrinkage. *Scientific Reports*, 5(1):2045–2322, December 2015. https://doi.org/10.1038/srep18485.

5. Carnegie Mellon Univeristy and The Johns Hopkins University Applied Physics Laboratory LLC. *Cybersecurity Maturity Model Certification (CMMC) 1.02*. Mar 2020. https://www.acq.osd.mil/cmmc/draft.html.

6. Sean Kinser, Pete de Graaf, Robert Roller, Amanda Salmoiraghi, Keith Scott, Matthew Stein, and David Voss. Scoring Trust Across Hybrid Space: A Quantitative Framework Designed to Calculate Cybersecurity-Specific Ratings, Measures, and Metrics to Inform a Trust Score. *(Submitted to 34th Annual Small Satellite Conference Proceedings)*, August 2020.

7. Peter Wegner. Developing a new interpretability scale to adapt to the evolution of satellite imaging. *SatMagazine*, Jun 2019. http://www.satmagazine.com/story.php?number=1377922534.