Federating Planning of Observations for Earth Science

Andrew Branch[1], **Yuliya Marchetti**[1], **James Mason**[1], **James Montgomery**[1], **Margaret C. Johnson**[1], **Steve Chien**[1], **Longtao Wu**[1], **Benjamin Smith**[1], **Lukas Mandrake**[1], **Peyman Tavallali** [1 (former)]

[1] Jet Propulsion Laboratory, California Institute of Technology

Corresponding Author: andrew.branch@jpl.nasa.gov

Abstract

We present an adaptive sensing method for federated planning observations of Earth science phenomena. Adaptive sensing is a technique that assists in studying these events by utilizing online analysis of the phenomena to determine the locations most advantageous for sensing. We use a federated planning framework to select observation requests for a set of heterogeneous assets and submit those requests to specific asset planners to produce detailed plans. We develop and demonstrate two constraint optimization algorithms to maximize the utility of the observation requests while following operational constraints of the assets. We show that adaptive sensing with federated planning is capable of collecting observations of significantly better quality than a baseline sampling approach.

Introduction

The large spatial scales and quick evolution of complex Earth science phenomena, such as extreme weather, often makes comprehensive sensing prohibitively difficult. Adaptive sensing can focus limited resources and observational capacity to extract the maximum quality science data. This technique utilizes online analysis of the target phenomena to direct sensing assets to collect the most advantageous observations. Figure 1 illustrates this concept. Observation utility is calculated based on a model of the phenomena and some scientific goal, this utility is used to plan observations for a set of assets, the plan is executed and the collected data is assimilated back into the model. This cycle then repeats while the target phenomena is active.

When studying these complex phenomena, it is often the case that a set of heterogeneous assets is used, with different sensing capabilities. Additionally, some or all of these assets may be operated by external organizations with an existing operations pipeline. To implement adaptive sampling in these systems, it could be prohibitively difficult to directly integrate into the existing planning software. Instead, they may expose a limited observation request interface that could be used. The federated planning paradigm, presented in this work, addresses this by using a multi-tiered planning approach, seen in Figure 2, to decouple the utility function from the individual asset operations pipelines.

Here we describe an effort to apply adaptive sensing with a federated planning paradigm to perform observational planning for the study of large scale storms such as hurricanes and typhoons. We utilize a numerical model of a hurricane and simulated assets to demonstrate the efficacy of these methods in improving the science return from a set of asset observations. The work performed here is part of a larger effort (Tavallali et al. 2020).

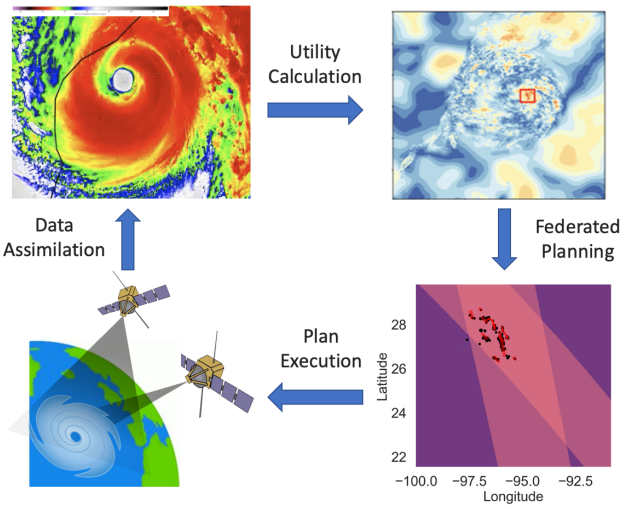

Figure 1: An illustration of the adaptive sensing concept. A utility function is created based on a model of the target phenomena, this utility function is used to generate a plan, which is then executed. Finally the data from those observations are assimilated back into the model and the cycle repeats.

Related Work

This project extends prior work on event-based satellite observation scheduling, where alerts from other assets were used to trigger observations of flooding (Chien et al. 2019) and volcanic activity (Chien et al. 2020). The work completed here builds on this by incorporating observation utility into the planning process instead of using simple event → observation triggers. Nag et al. (2020) presents a

tool to schedule distributed spacecraft observations to maximize science return, but does not consider non-spacecraft assets or a federated approach to planning. An AAV path planning algorithm to maximize the information gain of science measurements in tropical cyclones is discussed in Darko et al. (2022). This is similar in that it is selecting the most beneficial observations, but it is only concerned with a single asset. Clement and Barrett (2003) develops a method for coordinating activities between multiple self-interested assets, but does not consider utility maximization and uses a different hierarchy for multiple agents. Chien et al. (2000) describes three frameworks for coordinating multiple agents. This describes the use of multiple levels of planning, similar to the federated approach described here, but does not directly address distributing goals (e.g. requested observations) between assets to maximize a black-box utility function. Robinson et al. (2017) describes a planning method for assigning observation requests to a set of heterogeneous assets to maximize utility based on Bayesian logistic regression, however it does not utilize a complex data model for calculating utility of observations. Fioretto, Pontelli, and Yeoh (2018) describes a multitude of approaches for general multi-agent constraint optimization, but does not address the specific problem discussed here. Aghighi and Bäckström (2015) performs a theoretical analysis of Cost-optimal and Net-benefit planning; a similar problem formulation to what is described here. Parjan and Chien (2023) address the problem of assigning observations to a set of spacecraft, modeled as a distributed constraint optimization problem, but does not address the federated problem or planning to maximize a complex utility function. Squillaci, Roussel, and Pralet (2021) and Levinson et al. (2022) both present MILP based methods for planning observations across a set of spacecraft, but they do not consider utility maximization or the federated problem. Schaffer et al. (2018), Le Moigne et al. (2017), and Maillard et al. (2023) discuss planning based tools for developing spacecraft missions, based on maximizing the science return. These tools focus on the design of such missions, while we discuss planning approaches for mission operations.

Problem Statement

In the federated planning paradigm, the objective of the federated planner is to select a set of observations to request from each available asset such that the utility of the final collected observations are maximized, while minimizing the cost of acquiring those observations. This problem is represented as a constraint optimization problem with inputs of a utility function, a set of assets to produce requests for, and a set of constraints, encoded as objective functions, related to those assets. These constraints utilize common specifications of the assets such as orbit path, max slew angle and rate, and AAV endurance to determine potential observations and allow for the federated planners optimization algorithm to avoid selecting requests that are self-conflicting and infeasible for the individual asset planners to schedule. It is generally expected that the federated planner will not be able to encode all operational constraints for all assets and thus will rely on the asset planners to account for all constraints and produce a final observation schedule.

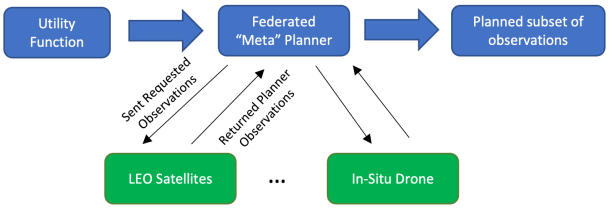

Figure 2: An illustration of the Federated Planning Paradigm. A federated planner takes the utility function as input and produces a set of requested observations for each potential asset. Those sets of requests are sent to a specialized asset planner which then produces planned observations for each asset.

Algorithms

Greedy

The greedy algorithm is a simple hill-climbing approach. Starting with an empty set of requests for each asset, each iteration adds one observation request to the set for one asset. The request/asset pair is determined by that which best improves the objective function. To do so, it calculates the utility gain given the addition of a single request/asset pair, for all such pairs. This continues until the objective function cannot be improved any further.

Simulated Annealing

A standard simulated annealing approach is used to perform hill-climbing search, with the ability to escape local maxima (Kirkpatrick, Gelatt Jr, and Vecchi 1983). The algorithm starts with an empty set of requests for each asset. Neighbors are selected in the following way. On each iteration, an observation request can be added to the solution, removed from the solution, or swapped for a different request. Which of these three options is selected at random, along with the observation request and the asset. If the resulting neighbor is an improvement over the previous solution, then it is kept and the search continues. If it is not an improvement, then it is kept with probability $e^{-\Delta E/t}$, where ΔE is the change in the objective function score and t is the current temperature value. The temperature is determined by the equation, $t = t_{start} * exp\_value^i$ where t_{start} is the starting temperature, exp_value is a number less than 1 indicating how quickly the temperature should decrease, and i is the current planning iteration. In our implementation we also include a dwell value which indicates how many neighbors to evaluate before moving to the next iteration and the next temperature value. The search terminates when the temperature reaches a pre-determined minimum value. For our experiments, we used the following parameter values: $t_{start} = 20000$, $exp\_value = 0.97$, $t_{end} = 0.05$, and $dwell = 100$. Generally, a longer search with a more gradual reduction in temperature will produce a better result, with diminishing returns. These values were determined experi-

mentally by determining when the solution was no longer improving with longer search times.

Experiments

We performed multiple simulation experiments to evaluate the performance of federated planning for adaptive sampling. These experiments focus on the federated planning and plan execution components. The objective is, given a utility function, to select observations to request such that the utility of the final taken observations is maximized. These experiments are performed with multiple asset types where each asset type is subject to unique operation constraints. This experiment consists of 10 runs where each run utilizes a different utility function and variations in the assets operational constraints. Both constraint optimization algorithms and a baseline planning method are evaluated on all 10 runs.

Experiment Procedure

Figure 3 outlines the procedure for each experiment run and the relevant data passed to each operation. First two preplanning steps are completed, the utility function is generated from the model and the planning constraints for each asset are determined from known asset specifications. The resulting utility function and set of constraints are provided to the federated planner. The federated planner uses one of the two algorithms outlined above to produce a set of observation requests for each asset. These observation requests are then passed to the individual asset planners to generate the final plans. Note that these sets of requests are the only input to these planners from any of the previous steps. This decouples them from the utility function, simplifying integration with existing operations pipelines. Thus the science objectives of the assets are entirely driven by the federated planner while the individual asset planners only attempt to maximize fulfilled requests. These plans are then executed. For this experiment we assume no deviation from the plans during execution, so all planned observations will be collected. Periodically during execution, replanning, if enabled, is triggered. After the experiment simulation is complete the final utility score of all taken observations can be calculated.

Utility Function

As part of the adaptive sensing strategy, we introduce a data-driven utility function and framework that determines where and which observations to sample, given a large number of possible locations and types of measurements. A utility function computes a certain value of one or a set of observations jointly, based on some specified science objective. Specifically, we focus on the overall reduced variance of forecast ensembles from an Ensemble Kalman Filter (EnKF) system (Zhang, Minamide, and Clothiaux 2016). EnKF models are extensively adopted for numerical weather prediction and modeling of complex dynamic systems, such as hurricanes.

We develop a utility function based on forecast sensitivity (Ancell and Hakim 2007; Xie et al. 2013), which is widely used to provide insight into the performance of the

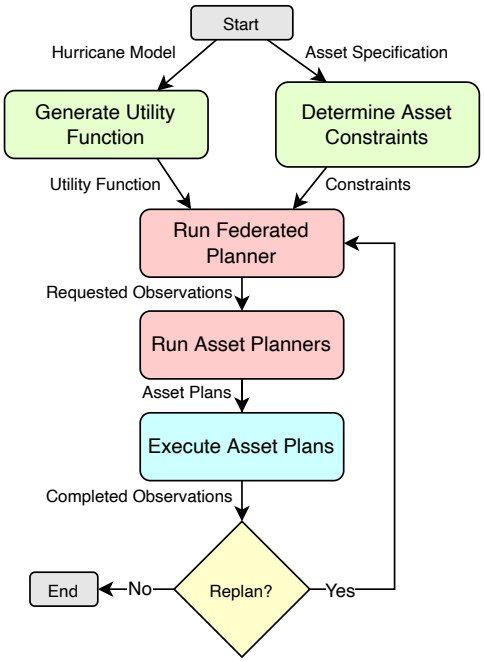

Figure 3: A flowchart of the procedure used for this experiment. The input to each step is labeled next to the arrows. Before planning, the utility function and all asset constraints are determined via input files. These are incorporated into the federated planner, which produces a set of requests for the individual asset planners. These asset planners produce a final plan for each asset. Those plans are then executed. If replanning is enabled, the process periodically repeats from the federated planner.

numerical forecast models that predict physical phenomena. In this context, the change in the forecast variance for a certain physical variable of interest, e.g. sea level pressure in case of a hurricane, can be approximated by a simple linear relationship as derived in Torn and Hakim (2008), therefore, the reduction in variance utility function could then be further obtained through the ordinary least squares (OLS) total sum of squares decomposition $\sum_i (y_i - \bar{y})^2 = \sum_i (\hat{y}_i - \bar{y})^2 + \sum (y_i - \hat{y}_i)^2$, where y_i is the actual value of the variable of interest, $\bar{y}$ is the mean of y_i and $\hat{y}_i$ are the corresponding predicted values of the variable of interest. It can then be defined as a percent variance explained, a generalized version of coefficient of determination or R^2, $\eta^2 = 1 - \frac{r}{v}$, where $v = \sum_i (y_i - \bar{y})^2$ is a total sum of squares and $r = \sum (y_i - \hat{y}_i)^2$ as residual sum of squares. The utility function for each variable of interest simply becomes:

$$u_S = \eta_S^2 \times v,$$

where S is a set of locations of interest. In other words, given the simplified assumptions of an EnKF-based forecast sensitivity metric, we could cast the utility function as measuring the model or explained sum of squares improvement given a set of locations and types of variables at those locations for predicting a specified variable of interest, e.g. sea level pressure.

In order to avoid the drawbacks of OLS and to generalize the problem, we substitute linear modeling with a non-linear machine learning method of ensemble trees. In particular, we estimate the utility u_S, by predictions $\hat{y}_i$ obtained from a Random Forest model, which also allows the use of all the locations jointly without over-specifying the problem (i.e. an OLS model will fit data perfectly with an expanded set of any inputs). The ensemble tree methodology is able to extract individual contributions of each location to the prediction of the variable of interest (Kuzmin et al. 2011), similarly to OLS contributions from linear regression coefficients. As a result, we can select a set of locations and the types of measurements for those locations that have the highest contributions to the prediction of the variable of interest, and therefore, possibly the highest utility values for observing these locations, which will also depend on the variance of the variable of interest, v, for each ensemble. Finally, the formulation of the utility function through ensemble trees also makes the computation relatively efficient in near-real time.

Since a large set of locations can be used to compute their impact on the forecast ensemble variance reduction, many of these locations will carry very small, but non-zero utilities as shown in Figure 4. To reduce the complexity of the planning problem, we filter the potential targeted observations to only contain a certain number of locations with the highest individual utilities, i.e. the utility if only that point was observed. For our experiments we limit our set to a 1000 locations. Although the utility function used in the federated planner calculates the joint utility of all requested observations, we found that the potential added value to the joint utility for a given location is unlikely to be significantly larger than each individual utility.

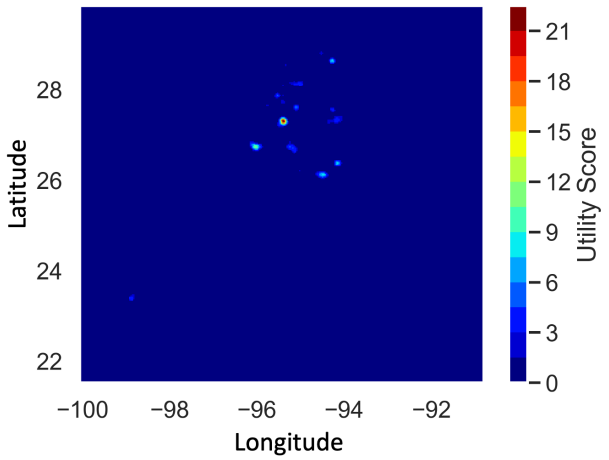

Figure 4: An example utility function used in the experiment.

Assets

Two types of assets were used in this experiment, low Earth orbit (LEO) satellites and autonomous aerial vehicles (AAV). The LEO satellites were roughly based on the Planet Labs SkySat constellation. Each satellite has a push-broom sensor with a field-of-view of 2 degrees, the capability to slew 15 degrees off-nadir in the across-track direction, and a slewing speed of 5 degrees per second. Along-track slewing and taking observations while slewing is not possible. Two satellites were used, resulting in two flyovers of the target region for each experiment run. To vary the flyovers, different SkySat orbits were used for the different experiment runs. The CLASP Planner was used as the asset planner for the LEO satellites (Maillard, Chien, and Wells 2021). The federated planner provides a set of observation requests to CLASP which uses a greed, Squeaky Wheel Optimization planner to produce a final plan for each LEO asset. As the LEO satellites use a pushbroom sensor, the set of taken observations will be larger than the set of requested observations. The utility score of the final plans are calculated using all taken observations, even those not requested.

The AAV assets were based on the Raytheon Coyote, which have been previously used for hurricane observations (Cione et al. 2016). Each AAV has a max endurance of 1 hour (3600 seconds) and a velocity of 30 m/s, resulting in a total travel distance of approximately 108km. The Google OR-Tools Vehicle Routing Problem solver is used as the AAV asset planner (Perron and Furnon 2021). This is a generalization of the Travelling Salesman Problem for N agents. Given a set of requested observations, a graph is constructed connecting each location to every other location with an edge of cost equal to the distance between the two observation locations. Additional, a start/end node is added with a 0 cost edge to every other location, so the resulting asset paths can start and end at any observation. The path length of each asset is also constrained based on the total endurance. As the federated planner can request more observations than are feasible, the solver is permitted to drop observations.

However, as this solver does not have access to the utility function, it will only attempt to maximize the number of observations completed. The solver uses the "Path Cheapest Arc" strategy to produce the initial solution. This strategy always extends the current path with the next cheapest location. This initial solution is then optimized through local search. Similar to the LEO assets, all locations along the AAV path length are included in the taken observations, even if they were not requested.

Objective Function

The objective function used for the federated planning is the linear combination of the utility function and the costs for four types of constraints. The objective function is shown below, where O_{All} represents the observation requests for all assets, O_a represents the observations requests for a single asset a, U is the utility function previously discussed and C_{1-4} are the constraints listed below.

$$F(O_{All}) = U(O_{All}) - \sum_{Assets}^{a} C_1(O_a) + C_2(O_a) + C_3(O_a) + C_4(O_a)$$

The utility function is calculated based on the combined asset observations and the constraints are calculated on a per asset basis. Each asset does not utilize every constraint type. One type of hard constraint (Max Requests) and three types of soft constraints (Observation Cost, Clustering, and LEO Slew) are used. Hard constraints, when violated, produce a cost of ∞, ensuring that combination of observation requests is not selected. Soft constraints instead produce a cost that increases depending on the severity of the constraint violation. This allows for minor violations and is important for the correct function of the simulated annealing algorithm. These constraints utilize basic asset specifications to roughly determine feasible observations for each asset. For the LEO asset this includes orbit path, max slew angle, and slew rate and for the AAV asset this includes endurance. In practice, specific asset planners will incorporate additional constraints that are not represented in the federated planner.

Max Requests Constraint The Max Requests Constraint places a hard limit on the number of observations that can be requested for a single asset. This constraint applies only to the AAV assets. During the federated planning procedure, it is difficult to estimate the path length based on the current set of requested observations. This constraint limits the number of total requests to a set that is reasonable for the AAVs to complete. This, of course, also depends on the locations of those requests, which is controlled by the Clustering Constraint. The equation below is used to calculate this constraint for an asset a. If the set of requested observations for asset a is greater than max_obs_a then the constraint is violated. max_obs_a is a tunable parameter based on the number of AAV assets and their endurance.

$$C_1 = \begin{cases} 0 & \text{if } len(requests_a) <= max\_obs_a \\ \infty & \text{otherwise} \end{cases}$$

Clustering Constraint The Clustering Constraint is a soft constraint that encourages observations for a single asset to be group spatially. This constraint applies only to the AAV assets. The limited endurance of the AAVs makes it critical that all requests are clustered into a small area, compared to the entire experiment region. This cluster helps to maximized the number of requests that a single AAV can fulfill. The equation below is used to calculate this constraint for an asset a. Any observation that is greater than $cluster\_size_a$ from the centroid costs $cost_c$. These two parameters are tunable based on the number of AAV assets and their endurance.

$$C_2 = \sum_{requests_a}^{r} \begin{cases} cost_c & \text{if } dist(r, centroid) > cluster\_size_a \\ 0 & \text{otherwise} \end{cases}$$

LEO Slew Constraint The LEO Slew constraint encourages observation requests to be feasible based on all the other requests. The possible observations that a LEO spacecraft can take are limited by its ability to slew from one pointing angle to another. For the spacecraft to take observations that are off-nadir in the across-track direction, it must slew the spacecraft such that the instrument is pointed at the target location. As this slewing procedure takes time, observations that are close in distance on the along-track axis but far in the across-track axis cannot both be observed. The equation below is used to calculate this constraint for an asset a. For each $request_a$ that is in conflict with any other request for that asset, $cost_s$ is added to the constraint. An observation is in conflict with another if they cannot both be observed by asset a in a single overflight. As only across-track slewing is allowed, determining if two observations are in conflict only requires calculating the spacecraft pointing angle at the time of each observation and calculating if the time between observations is greater than the time required to slew between those two pointing angles.

$$C_3 = \sum_{requests_a}^{r} \begin{cases} cost_s & \text{if } conflict(r, requests_a) \\ 0 & \text{otherwise} \end{cases}$$

Observation Cost Constraint The Observation Cost Constraint applies a cost for the total number of requests per asset. The asset planners only attempt to maximize the number of fulfilled requests and cannot reason about the observation utility, as such, making significantly more requests than are feasible can reduce the quality of the resulting final observations. Each asset is allowed to have a different cost per observation, however for our experiment, all assets used the same cost. This constraint limits the number of observations that will be requested. Observations that provide low marginal utility will not be added as this constraint cost will offset the positive utility value. The equation below is used to calculate this constraint for an asset a. $cost\_per\_obs_a$ is a tunable parameter for each asset.

$$C_4 = len(requests_a) * cost\_per\_obs_a$$

Replanning

Replanning can be performed periodically to account for requested observations that were not fulfilled by the asset planners or were not observed during plan exeuction. This in done by re-running the federated planner, generating a new set of observation requests, and then generating new asset plans using the asset planners. The exact replanning strategy used is dependent on the assets operational constraints. For this experiment, we used a simple replanning strategy. Replanning is performed in any time interval over a fixed duration that does not contain any observations. This is to allow time for planning to complete and for the updated plans to be sent for execution. The time inverval is selected such that replanning was performed between overflights of the LEO satellite assets and after the completion of the AAV asset flights. When the replanning is performed by the federated planner, the previously taken observations are accounted for when calculating the utility function.

Baseline

A baseline method representative of non-adaptive search was developed for comparison against the adaptive approached. In this baseline method, the LEO spacecraft takes continuous nadir observations over the target region. AAVs were given N random target locations in a small search region near the area of minimum sea level pressure in the hurricane model. That region was selected as the scientific goal used to produce the utility function was minimizing the uncertainty in the modeled minimum sea level pressure. The AAV paths are planned using the black box Vehicle Routing Problem planner. No replanning is performed in the baseline method.

Results

An example plan using the federated planning approach is shown in Figure 5. The reachability swaths for the two LEO assets are shown in light red. These swaths represent all possible observations that the LEO assets can take. Each black dot is an observation that has been requested from one of the assets. The smaller dark red swaths represent the taken observations from the LEO assets. The dark red paths represent the two executed AAV paths.

Over all the experiment runs, the Simulated Annealing and Greedy algorithms outperformed the Baseline method. There was little difference, with respect to the final utility of all taken observations, between the Simulated Annealing and Greedy algorithms. Figure 6 shows the resulting utility of all taken observations, grouped by the three different algorithms, Baseline, Simulated Annealing, and Greedy.

The wall clock runtime of the two federated approaches are shown in Figure 7. Here we see that although the Simulated Annealing algorithm did not result in an improved observation utility. The required runtime to achieve those results was significantly lower than that of the greedy algorithm.

In addition to comparing the federated planning methods to the baseline approach, we also compared the capability of replanning versus no replanning for those approaches.

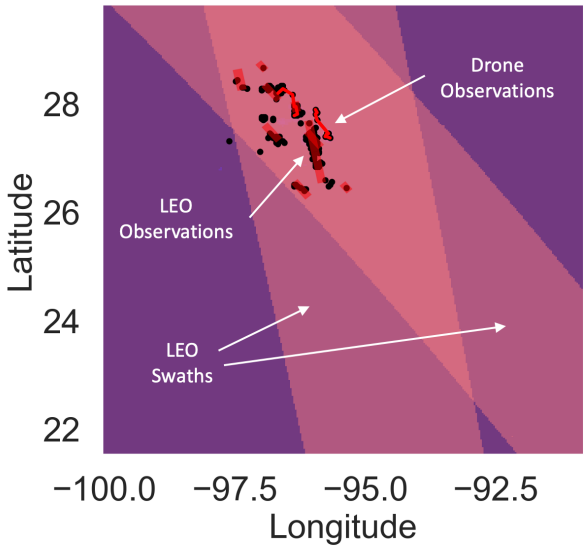

Figure 5: An example execution from the experiment. The LEO swaths, representing all possible LEO observations, are represented in light red. The actual LEO observations and the AAV paths are represented in dark red. The black dots mark requested observations.

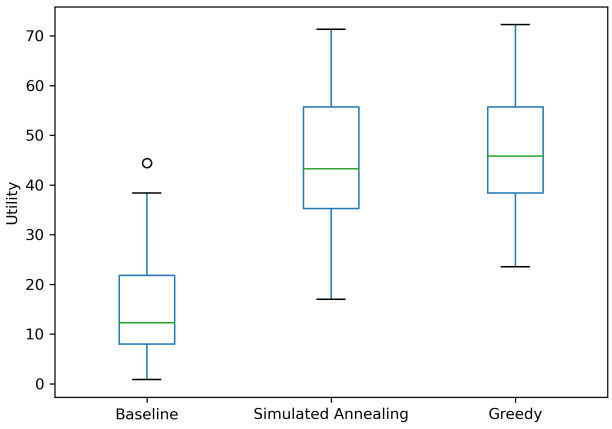

Figure 6: The utility of collected observations for each algorithm over all experiment runs.

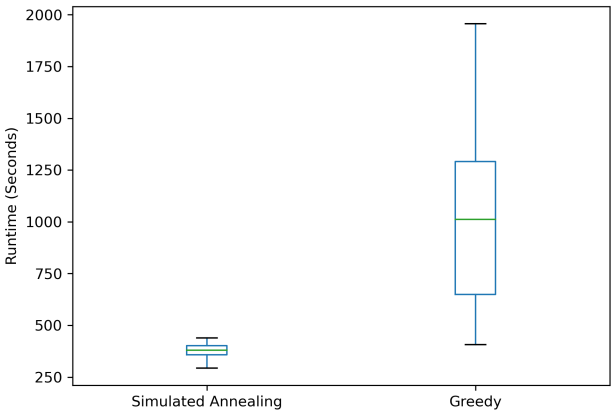

Figure 7: The wall-clock runtime for each algorithm over all experiment runs.

In Figure 8 we see a minor improvement with replanning enabled. The number of experiment runs with low utility scores reduced, resulting in the median increasing by approximately 5 points and the spread of the distribution over all runs to decrease.

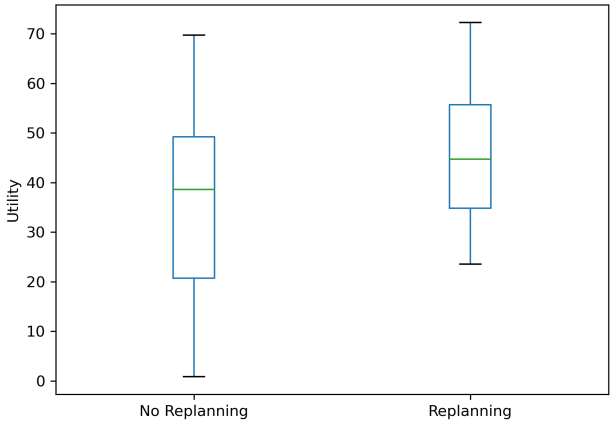

Figure 8: The utility of collected observations with replanning enabled versus disabled over all experiement runs.

Discussion

The experiment results demonstrate that the federated planning paradigm and adaptive observation approach can produce observation plans that improve results over baseline observation methods, based on the utility of the completed observations. Little difference was seen in the resulting observation utility of the two federated planning algorithms. However, the runtime for the Simulated Annealing approach is significantly shorter than that of the Greedy approach. Each timestep of the Greedy approach requires evaluating the change in utility for each potential observation request, resulting in significant wasted computation for evaluating low utility observations at each timestep. Simulated Annealing is able to avoid this by evaluating the utility only once per timestep. Although this will often result in initially adding many low-quality observations to the solution, the ability to later removing these low-quality observations results in a final solution of similar quality to the Greedy approach for much lower computation effort. Simulated Annealing has the additional capability of escaping local maxima to produce an improved solution, however as the quality of the final solution is similar to that of the Greedy approach that is likely not an important factor for the specific problem setup in these experiments.

The addition of replanning for this experiment resulted in a small increase in the utility of the final taken observations. The expected improvements from replanning come from the differences in the taken observations versus the requested observations. In this experiment, the planning constraints match the assets observation capabilities well, so most requested observations are fulfilled by the assets. This results in the small improvement to the final results. The addition of contention with other external requests or additional asset observational constraints that are not well represented with the federated planning constraints would likely result in a more significant improvement from replanning.

Future Work

This work provides an initial proof of concept for adaptive sampling using the federated planning paradigm. The asset observation models used here were fairly simplistic. Real-world assets would have additional constraints that would be difficult to model in the federated planner, including wind-velocity, AAV deployment capabilities, LEO observation contention, and data limitations. We would like to investigate these additional constraints and determine how that affects the performance of the federated planning algorithm. Additional, the performance of other constraint optimization algorithms should be evaluated. The federated planning paradigm should also be evaluated using a more diverse variety of possible utility functions and potential assets. Ideally these next steps would be completed with a more concrete set of assets to target, including operational asset planners, as that would drive the experiment towards realistic constraints and optimization objectives.

Conclusion

In this paper we presented work on developing and demonstrating an adaptive sensing system for Earth Science using a heterogeneous set of observation assets. We utilized a federated planning paradigm for requesting observations from those assets. We developed two algorithms, a greedy hill climbing approach and Simulated Annealing to select observation requests. These developments were tested against a baseline non-adaptive observation planning method in a simulation experiment for planning hurricane observations. The experiment results showed significant improvement of both algorithms over the baseline method. This work has demonstrated that the adaptive sensing concept and federated planning paradigm can be used to improve science returns over non-adaptive sampling methods.

Acknowledgements

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004).

References

Aghighi, M.; and Bäckström, C. 2015. Cost-optimal and net-benefit planning: a parameterised complexity view. In *24th International Joint Conference on Artificial Intelligence (IJCAI-15), Buenos Aires, Argentina, Jul 25-31, 2015*, 1487–1493. IJCAI-INT JOINT CONF ARTIF INTELL, ALBERT-LUDWIGS UNIV FREIBURG GEORGES ....

Ancell, B.; and Hakim, G. J. 2007. Comparing adjoint-and ensemble-sensitivity analysis with applications to observation targeting. *Monthly Weather Review*, 135(12): 4117–4134.

Chien, S.; Barrett, A.; Estlin, T.; and Rabideau, G. 2000. A comparison of coordinated planning methods for cooperating rovers. In *Proceedings of the fourth international conference on Autonomous agents*, 100–101.

Chien, S.; Mclaren, D.; Doubleday, J.; Tran, D.; Tanpipat, V.; and Chitradon, R. 2019. Using taskable remote sensing in a sensor web for Thailand flood monitoring. *Journal of Aerospace Information Systems*, 16(3): 107–119.

Chien, S. A.; Davies, A. G.; Doubleday, J.; Tran, D. Q.; Mclaren, D.; Chi, W.; and Maillard, A. 2020. Automated volcano monitoring using multiple space and ground sensors. *Journal of Aerospace Information Systems*, 17(4): 214–228.

Cione, J. J.; Kalina, E.; Uhlhorn, E.; Farber, A.; and Damiano, B. 2016. Coyote unmanned aircraft system observations in Hurricane Edouard (2014). *Earth and Space Science*, 3(9): 370–380.

Clement, B. J.; and Barrett, A. C. 2003. Continual coordination through shared activities. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 57–64.

Darko, J.; Folsom, L.; Park, H.; Minamide, M.; Ono, M.; and Su, H. 2022. A Sampling-Based Path Planning Algorithm for Improving Observations in Tropical Cyclones. *Earth and Space Science*, 9(1): e2020EA001498.

Fioretto, F.; Pontelli, E.; and Yeoh, W. 2018. Distributed constraint optimization problems and applications: A survey. *Journal of Artificial Intelligence Research*, 61: 623–698.

Kirkpatrick, S.; Gelatt Jr, C. D.; and Vecchi, M. P. 1983. Optimization by simulated annealing. *science*, 220(4598): 671–680.

Kuzmin, V. E.; Polishchuk, P. G.; Artemenko, A. G.; and Andronati, S. A. 2011. Interpretation of QSAR models based on random forest methods. *Molecular informatics*, 30(6-7): 593–603.

Le Moigne, J.; Dabney, P.; de Weck, O.; Foreman, V.; Grogan, P.; Holland, M.; Hughes, S.; and Nag, S. 2017. Tradespace analysis tool for designing constellations (TAT-C). In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 1181–1184. IEEE.

Levinson, R.; Niemoeller, S.; Nag, S.; and Ravindra, V. 2022. Planning Satellite Swarm Measurements for Earth Science Models: Comparing Constraint Processing and MILP Methods. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 32, 471–479.

Maillard, A.; Chien, S. A.; and Wells, C. 2021. Planning the Coverage of Planets under Geometrical Constraints. *Journal of Aerospace Information Systems*, 18:5: 289–306.

Maillard, A.; Wells, C.; Oveisgharan, S.; Rosen, P.; and Chien, S. 2023. Where is my coverage? Using explainable automated scheduling to inform mission design of an Earth-observing constellation. In *2023 International Workshop on Planning & Scheduling for Space (IWPSS 2021)*.

Nag, S.; Moghaddam, M.; Selva, D.; Frank, J.; Ravindra, V.; Levinson, R.; Azemati, A.; Aguilar, A.; Li, A.; and Akbar, R. 2020. D-shield: Distributed spacecraft with heuristic intelligence to enable logistical decisions. In *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, 3841–3844. IEEE.

Parjan, S.; and Chien, S. 2023. Decentralized Observation Allocation for a Large-Scale Constellation (in press). *Journal of Aerospace Information Systems (JAIS)*.

Perron, L.; and Furnon, V. 2021. OR-Tools.

Robinson, E.; Balakrishnan, H.; Abramson, M.; and Kolitz, S. 2017. Optimized stochastic coordinated planning of asynchronous air and space assets. *Journal of Aerospace Information Systems*, 14(1): 10–25.

Schaffer, S.; Chien, S.; Branch, A.; and Hernandez, S. 2018. Automatic orbit selection for a radio interferometric spacecraft constellation. *Journal of Aerospace Information Systems*, 15(11): 627–639.

Squillaci, S.; Roussel, S.; and Pralet, C. 2021. Managing complex requests for a constellation of Earth observing satellites. In *2021 International Workshop on Planning & Scheduling for Space (IWPSS 2021)*.

Tavallali, P.; Chien, S.; Mandrake, L.; Marchetti, Y.; Su, H.; Wu, L.; Smith, B.; Branch, A.; Mason, J.; and Swope, J. 2020. Adaptive Model-driven Observation for Earth Science. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation for Space*, i-SAIRAS'2020. Noordwijk, NL: European Space Agency.

Torn, R. D.; and Hakim, G. J. 2008. Ensemble-based sensitivity analysis. *Monthly Weather Review*, 136(2): 663–677.

Xie, B.; Zhang, F.; Zhang, Q.; Poterjoy, J.; and Weng, Y. 2013. Observing strategy and observation targeting for tropical cyclones using ensemble-based sensitivity analysis and data assimilation. *Monthly weather review*, 141(5): 1437–1453.

Zhang, F.; Minamide, M.; and Clothiaux, E. E. 2016. Potential impacts of assimilating all-sky infrared satellite radiances from GOES-R on convection-permitting analysis and prediction of tropical cyclones. *Geophysical Research Letters*, 43(6): 2954–2963.