D-SHIELD: DISTRIBUTED SPACECRAFT WITH HEURISTIC INTELLIGENCE TO ENABLE LOGISTICAL DECISIONS

Sreeja Nag[1], *Mahta Moghaddam*[2], *Daniel Selva*[3], *Jeremy Frank*[4]

[1]NASA Ames Research Center and BAER Institute, Moffet Field, CA 94035
[2]University of Southern California, Los Angeles, CA 90089
[3]Texas A&M University, College Station, TX 77843
[4]NASA Ames Research Center, Moffet Field, CA 94035

ABSTRACT

D-SHIELD is a suite of scalable software tools that helps schedule payload operations of a large constellation, with multiple payloads per and across spacecraft, such that the collection of observational data and their downlink, constrained by the constellation constraints (orbital mechanics), resources (e.g., power) and subsystems (e.g., attitude control), results in maximum science value for a selected use case. Constellation topology, spacecraft and ground network characteristics can be imported from design tools or existing constellations and can serve as elements of an operations design tool. D-SHIELD will include a science simulator to inform the scheduler of the predictive value of observations or operational decisions. Autonomous, real-time re-scheduling based on past observations needs improved data assimilation methods within the simulator.

Index Terms— Agile Autonomy, Spacecraft Constellations, Satellite Remote Sensing, Onboard Planning

1. INTRODUCTION

Earth-science processes are intrinsically dynamic, complex, and interactive. To achieve an all-embracing understanding of the emergence and evolution of these processes requires the collection and assimilation of enormous amounts of data, using complementary measurements in space and time. Command and control of satellites (or other observing assets) can be informed by a highly simplified, real-time version of an Observing System Simulation Experiment (OSSE). In the traditional sense, an OSSE is a data analysis *experiment* used to evaluate the impact of new *observing systems* on operational forecasts, by *simulating* the natural phenomena of interest when actual observational data are not fully available. An OSSE comprises a free-running model ('nature run'), whose fields are then used to compute 'synthetic observations' for all current and new observing systems, with added random errors representative of measurement uncertainty. Synthetic observations represent a small, noisy subset of the ground truth. They are then processed by a data assimilation system, whose forecasts are compared to the nature run. The traditional goal of OSSEs is to validate science or operational return for proposed instruments and justify their designs by showing little disparity between the nature run of a chosen scenario and the instrument-derived forecasts, as seen in missions such as SMAP, CLARREO and HyspIRI. While OSSEs have never been used to inform mission re-planning, since they are computationally inefficient to run real-time, they may be used to train machine/deep learning (ML/DL) architectures that can run in soft real-time. For example, [1] describes an OSSE-based simulator to select high-level parameters, e.g., number of satellites and orbits, for a formation flight mission to estimate bi-directional reflectance distribution function. Value-driven, adaptable schedulers have been operationally supported by NASA over many years (e.g. Space Technology-5); however, physics-based OSSEs have never been used to train such autonomous planners.

Small spacecraft and the proliferation of launch service providers have lowered the cost of access to space, allowing distributed space missions (DSMs) to complement unmanned aerial systems (UAS) and vast ground-based networks in building an agile Sensor Web for Earth observation (EO). Smaller spacecraft enable larger numbers of them; therefore, NASA is supporting the miniaturization of high resolution imagers, radars, and other instruments to fly on small spacecraft (defined as <180 kg) and even CubeSats spanning 6U/14 kg to 27U/40 kg. Small spacecraft can now re-orient in three degrees of freedom to re-point their instruments within short notice (e.g. CHRIS on Proba), have onboard processing to interpret collected science data and potentially improve their observing plans (e.g. IPEX CubeSat as a HyspIRI pathfinder), and are supported by inter-sat communication links to transmit data or metadata (e.g. NODeS demonstration on Edison-like CubeSat). Whether the spacecraft in a DSM are small with agile, re-orienting bodies, or large with agile, gimbaled instruments, software tools that support the evaluation (e.g. using OSSEs) and scheduling operations of DSMs can exploit the full potential of such hardware advancements.

Scheduling algorithms for agile EO have been successfully developed for *single, large satellite* missions, examples being ASPEN for EO-1, the ASTER Radiometer on Terra, Ikonos commercial satellite, the geostationary yet-to-fly GEO-CAPE satellite[2], or image strips over Taiwan by ROCSAT-II. The problem of tasking multiple, diverse sensors was preliminarily addressed for aerial flight paths on

NASA's INTEX-B flight data. Coordinated planners can handle a continuous stream of image requests, and agent-based schedules have been implemented for static execution on NASA's Deep Space 1. However, scheduling tools for single spacecraft or non-orbital aircraft are not necessarily scalable to large constellations. For example, stochastic algorithms are accurate, but at the unacceptable cost of initial condition dependence, exponential time to converge or large training sets. Scheduling observations for *constellations of large satellites with payload re-pointing* has been formulated for the French Pléiades constellation and COSMO-SkyMed constellation of synthetic aperture radars. However, these schedulers are specific to large spacecraft. Small satellites need to re-orient their full body to point to evolving targets, thus need special consideration of steering dynamics. Schedulers for *CubeSat constellations*, such as the 200+ Dove spacecraft fleet operated by Planet Lab, Inc. assume static orientation of the sensor in orbit and only schedule duty cycles for payload power. Accounting for *full re-orientation in multi-spacecraft missions* imposes computationally expensive constraints on scheduling spacecraft slews between payload operations. It is only recently that scheduling with slew-time variations[3] has shown reasonable convergence using hierarchical division of assignment and step-and-stare approaches using matrix imagers. Planet Labs has published a preliminary scheduler used to operate their agile Skybox spacecraft fleet. However, these agile observation schedulers cannot re-compute science value in real-time nor autonomously re-schedule. The increasing number of operational small sats with limited downlink bandwidth has spurred literature on *scheduling CubeSat data download from/to a network*; e.g. optimization of single CubeSat downlink to a network of ground stations (GS) and multiple payloads' downlink to existing stations within available storage, energy and access time constraints. While they use crosslinks to propagate planning information through the constellation, the tools are optimized for data downlink, and *not* for commanding better observations and science. Since they are agnostic to the data content (only size matters), payload type and concepts of operations, they are not particularly appropriate for custom Earth Science applications. Moreover, planners that negotiate task assignment but without realistic orbital constraints or simulate the space environment but not realistic inter-satellite communication, are far more common than those that consider both factors.

A ground-based, autonomous scheduling algorithm that optimizes the operations of small spacecraft attitude control systems (ACS) to maximize any computable black-box value function for a known constellation has been demonstrated[4]. A modified version of the algorithmic framework can be used onboard in real-time, by leveraging inter-satellite links implementing the delay/disruption tolerant network (DTN) paradigm - a standard for routing in

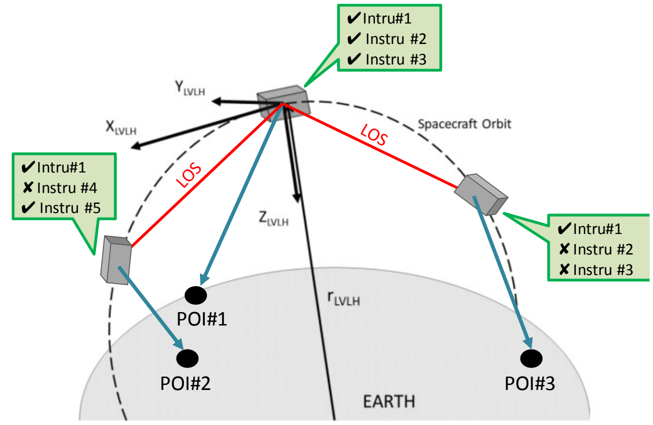

Figure 1: Cartoon of a 3-sat DSM, with different instrument sets, using D-SHIELD to make coordinated decisions of pointing and instrument usage. Communication is intermittent since spacecraft may not always have line of sight (LOS).

dynamic, intermittent operations environment. Improved coordination among multiple spacecraft generated more valuable measurements of fast changing precipitation and urban floods[5]. D-SHIELD will improve fidelity and extend the said framework to optimize the re-orientation and operations of multiple payloads for soil moisture applications with *high variability* in spatio-temporal requirements using a hybrid approach of DTN-enabled onboard and ground-based scheduling. The payloads will have different characteristics and inter-dependencies, and be heterogeneously distributed in a constellation. NASA has been investing in constellations; e.g., CYGNSS measures wind speed with 8 satellites, and TROPICS will increase temperature/pressure profiles of tropical cyclones with 6 spacecraft. D-SHIELD would make such constellations more responsive to evolving observables.

2. PROPOSED METHODOLODY

D-SHIELD comprises three components, as seen in Figure 2: [A] *an intelligent scheduler* that can be run on the ground in a centralized manner or onboard spacecraft in a distributed manner, to help operate agile DSMs for reactive remote sensing, i.e., we support the development of a new observing strategy. Scheduled tasks include heterogeneous payload operations, spacecraft re-orientation for payload re-pointing, payload coalitions across spacecraft for coordinated observation, inter-sat crosslink and downlink. The scheduler is informed by [B] *an observable science simulator* that not only enables its operational choices, but also enables the comparison of intelligent, agile DSMs complementing traditional approaches, against traditional missions alone. The scheduler system is packaged into [C] *an operations tradespace analyzer*, that will evaluate the performance of the system, and inform various trade-offs such as running it onboard or on ground, for a given relevancy scenario (e.g. an urban flood vs. monitoring

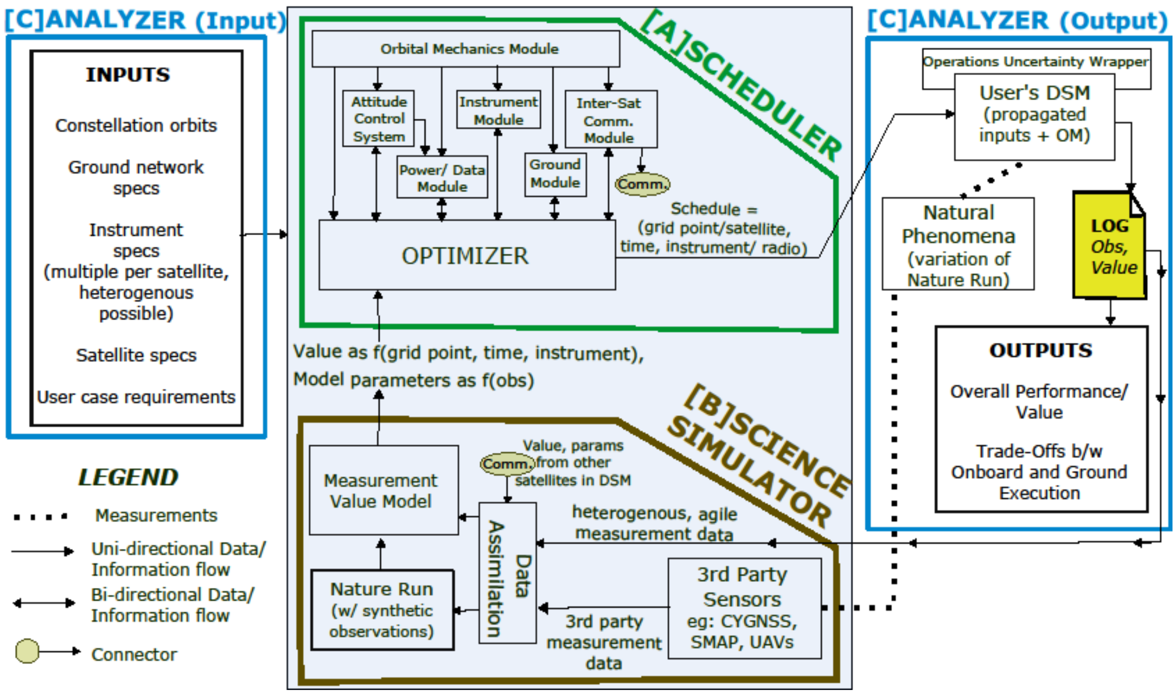

Figure 2: Information Flow chart of the D-SHIELD Technology (Scheduler + Science Simulator) and Assessment plan (using Analyzer, which also serves as the User Interface)

melting snow). The analyzer may also serve as a plug-in into an external constellation/spacecraft/instrument design, to inform how operations affect decisions on the number of spacecraft, instrument characteristics, etc. The constellation architecture can be an output from a design tool (e.g. TAT-C[6]) or a current fleet (e.g. Digital Globe). The Analyzer also informs the hybridization ratio (when will the scheduler run on the ground optimizing for all sats using downloaded data in a centralized manner and uploading schedules during overpasses, and when will it run onboard spacecraft in a distributed across the constellation using cross-linked data as it comes through the DTN?) as a function of science requirements, inter-sat comm, onboard processing capabilities, etc. The framework will be scalable to dozens of spacecraft. While it is agnostic to size, we will ensure that it is implementable on small spacecraft because they have tighter resource constraints, and are more likely to be deployed in large numbers. In the future, the scheduler can be integrated with open-source flight software such as SpaceCubeX or Core Flight Software (cFS), or ground software such as Univ. of Hawaii's COSMOS.

3. INITIAL RESULTS

The D-SHIELD Scheduler's output is a series of (*GP/S, tStep, Instru*) tuples that informs the satellite that it should orient its instrument *instru* toward grid point *GP* or satellite *S* at time *tStep*, and turn it on. The instruments include the scientific payload and radio(s). The relative value of observing a grid point (GP) with a science *Instru* at some *tStep,* and parameters that affect operational decisions (e.g. solar viewing angle, off-nadir pointing angle, time or spatial distance between observations) are iteratively received from the Science Simulator, which continuously re-evaluates value based on past observations per the executed schedule. The feedback loop allows D-SHIELD to make up for missed/incorrect observations, and changing targets.

Three of six modules that inform Scheduler's Optimizer have been prototyped- The *orbital mechanics (OM)* module propagates the given constellation orbits, discretizes the regions of interest into grid points and computes possible coverage per the given instrument specs, and access opportunities between the constellation and a given Ground Station (GS) network. The OM module also calculates LOS between satellites (contact map), inter-sat distances in the constellation at any time, and priority of bundle delivery (based on the ordering of satellites expected to access a region from the time of packet generation) for inter-sat exchange. The *ACS module* uses the OM's outputs, with known satellite and subsystem (hardware) characteristics to compute: the time required by any satellite at a given time to slew from one gridpoint to another (also considering satellite movement), resultant power consumption, momentum and stabilization profiles. In a highly dynamic environment with moving sensors and limited resources on a small sat, the accuracy and speed of this computation is paramount. If the (typically large) spacecraft is static with gimbaled instruments or beam steering, instead of full body re-orientation (typically small sat), the modular ACS can be modified, or replaced by parametric equations or fixed payload re-orientation time. The *Inter-Spacecraft Communications (comm.) module* computes the link budget for a known set of specifications and protocols. It uses the resultant data rate, with contact opportunities and bundle priority, to simulate DTN and compute bundle drop rates and latency to deliver any known bundle between any given pair of satellites. This module is essential for efficiency of the onboard scheduler, because each spacecraft will compute schedules individually and needs knowledge from others to minimize replication of planned observations. Comm. packet latency affects the level of consensus among the DSM assets (intent sharing), and recentness of observations in the data assimilation process (heuristics sharing). The ground-based scheduler computes tasks for all spacecraft in a centralized manner, therefore latency is more sensitive to up/downlink opportunity than DTN delays. Future work includes the *Ground Module* that simulates data and C2 exchange with centralized ground systems; the *Payload Module* that will model radars and radiometers (candidate soil moisture instruments) with varying fields of view and look angles as well as impose rule-based or quantitative constraints on the optimizer to ensure that the scheduler output takes into account if the simultaneous operation of multiple instruments on a given spacecraft is possible and if ACS orientation of the spacecraft required by those instruments is feasible; the *Power and Data Module* that will maintain an energy flow/data budget constraining the options of the scheduler because radars are very power hungry and small sats are very power constrained, allowing ops for only a fraction of an orbit.

We have built a prototype Optimizer based on greedy path selection using dynamic programming (DP). Agile ACS and DTN-enabled schedules on a 24-sat homogeneous constellation with a single imager and no resource constraints show improved value for urban flood monitoring, over ground-scheduled, agile or static approaches. The algorithm complexity is linear in planning horizon, number of GPs, but scales exponentially with number of conflicting spacecraft. An Integer Programming (IP) exact solver verified that our DP-optimizer could find solutions within 10% optimality for single satellites and 22% for constellations, within runtimes less than 2% of real time execution. The IP solver, however, took 4 orders of magnitude more time to improve DP's solutions. Future work will develop a modular, fast optimization approach that can handle the newly added complex aspects of payload operations, guarantee solutions in real time, for operational use in global missions scalable to scores of assets. We will also consider decentralized approaches based on consensus-based coalition forming algorithms[7], such that specific combinations of instruments can form opportunistic coalitions among satellites to observe target regions.

The D-SHIELD Science Simulator is based on an OSSE developed for a soil moisture relevancy scenario and expected to take into account data from third party sources – spaceborne (e.g. Sentinel-1, SMAP), airborne (e.g. P-band AirMOSS and L-band UAVSAR) or ground based sensors (e.g. SoilSCAPE wirless sensor networks in CA, AZ, AK) – as well as measurements made by the user's DSM at a previous time. The Data Assimilator is *not* a high-fidelity model, but currently aimed for short-time-horizon predictions. The DSM measurements comprise of those made by the spacecraft executing the scheduler, and inputs from other spacecraft received via DTN or via the ground. The current version uses in-house numerical radar scattering models and generates radar backscattering maps for an arbitrarily defined landscape based on static layers that include landcover, soil texture, and topography, and dynamic variables such as measured soil moisture[8]. It uses ML-based upscaling models for sparse in-situ measurements that can generate high resolution (100 m) soil moisture field estimates at spatial domains of tens of kilometers. Future work includes development of a passive microwave simulator, a hydrologic land-surface model/simulator, a spatio-temporal value model and the D-SHIELD Analyzer.

4. ACKNOWLEDGEMENT

The authors wish to thank NASA's Earth Science Technology Office (ESTO) for funding via the Advanced Information Systems Technology (AIST) program.

5. REFERENCES

[1] S. Nag, C. K. Gatebe, and O. de Weck, "Observing system simulations for small satellite formations estimating bidirectional reflectance," *Int. J. Appl. Earth Obs. Geoinformation*, vol. 43, pp. 102–118, 2015

[2] J. Frank, M. Do, and T. T. Tran, "Scheduling ocean color observations for a GEO-stationary satellite," in *International Conference on International Conference on Automated Planning and Scheduling*, 2016, pp. 376–384.

[3] L. He, X.-L. Liu, Y.-W. Chen, L.-N. Xing, and K. Liu, "Hierarchical scheduling for real-time agile satellite task scheduling in a dynamic environment," *Adv. Space Res.*, vol. 63, no. 2, pp. 897–912, Jan. 2019

[4] S. Nag, A. Li, and J. Merrick, "Scheduling Algorithms for Rapid Imaging using Agile Cubesat Constellations," *COSPAR Adv. Space Res.*, vol. 61, no. 3, pp. 891–913, 2018.

[5] S. Nag *et al.*, "Autonomous Scheduling of Agile Spacecraft Constellations with Delay Tolerant Networking for Reactive Imaging," in *International Conference on Planning and Scheduling (ICAPS) SPARK Applications Workshop*, Berkeley, California, U.S.A., 2019.

[6] J. LeMoigne *et al.*, "Tradespace Analysis Tool for Designing Constellations (TAT-C)," in *IEEE International Geoscience and Remote Sensing Symposium*, Fort Worth, TX, 2017.

[7] X. Gallud and D. Selva, "Agent-based simulation framework and consensus algorithm for observing systems with adaptive modularity," *Syst. Eng.*, vol. 21, no. 5, pp. 432–454, 2018.

[8] A. Tabatabaeenejad, M. Burgin, X. Duan, and M. Moghaddam, "P-band radar retrieval of subsurface soil moisture profile as a second-order polynomial: First AirMOSS results," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 645–658, 2015.