Decentralized, Decomposition-Based Observation Scheduling for a Large-Scale Satellite Constellation

Itai Zilberstein, Ananya Rao, Matt Salis, Steve Chien

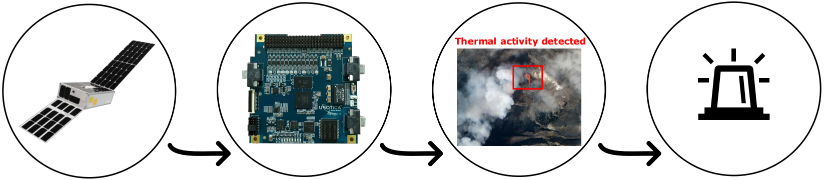
OptLearnMAS (Presentation Only)
Appears in Journal of Artificial Intelligence Research [Zilberstein et al., 2025]

May 20, 2025

Jet Propulsion Laboratory
California Institute of Technology

Background and Motivation

- Larger satellite constellations with 100s of spacecraft (e.g. Spire, Satellogic, Canon, SatRev, Planet Flock) [NewSpace, 2023]
- New satellites have increased onboard capabilities
  - AI acceleration hardware
  - ISL for persistent communications
- These enable new measurements and insights into Earth
  - Rapid responses: volcano monitoring, tracking floods and wildfires, etc.
- **Challenge:** How to coordinate actions of 100s of spacecraft onboard?



1. Overflight and data collection
2. Onboard data analysis
3. Anomaly or event detected
4. Issue alert for timely next actions

Related Work (1/2)

- Onboard data analysis [Chien et al., 2024, Zilberstein et al., 2024]
- Onboard single agent planning [Chien et al., 2025]

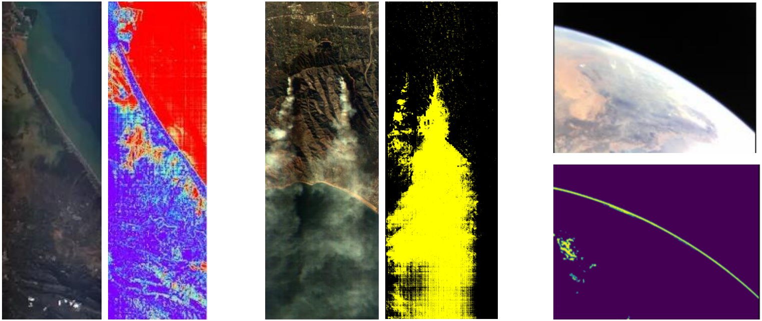


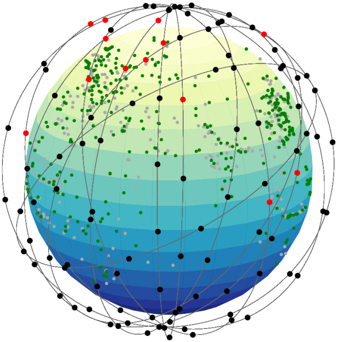
Figure: Onboard data analysis samples - flooding in Valencia (left), wildfire plumes in LA (center), off-limb anomaly (right). *Includes imagery from CogniSAT-6/HAMMER, 2025, Ubotica. All rights reserved.*

Related Work (2/2)

- Most work on centralized scheduling of satellite constellations [Augenstein et al., 2016, Boerkoel et al., 2021, Globus et al., 2004, Nag et al., 2018, Shah et al., 2019]
  - All operational systems are centralized
  - Lack scalability, adaptability, robustness, and may suffer from latencies
- Some work on decentralized scheduling of satellite constellations [Phillips and Parra, 2021, Picard, 2021, Parjan and Chien, 2023]
  - More on these approaches later!

Contributions

- **Goal**: increase scalability, adaptability, robustness and leverage distributed computation through decentralized scheduling solutions
- Contents of this presentation:
    1. Obstacles in applying existing techniques
    2. Present the *Neighborhood Stochastic Search Algorithm* (NSS)
    3. Evaluation of NSS on large-scale simulations



Problem Statement

Multi-Satellite Constellation Observation Scheduling Problem (COSP):

1. $H = [h_s, h_e]$: the scheduling horizon
2. $\mathcal{K}$: the set of orbital planes.
3. A: the set of agents, where each $a_i \in A$ is a satellite in the constellation.
4. R: the set of requests.

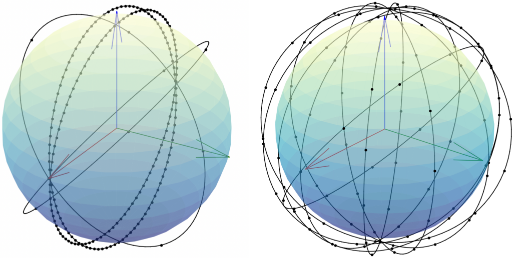


Figure: LEO constellations with hundreds of satellites. Left constellation called Planet, right called Walker.

Problem Statement (2/2)

For each agent, we have:

1. S_{a_i}: set of possible *request fulfillments*
2. $\mathcal{X}_{a_i}$: Boolean decision variables; $x_j = 1 \iff a_i$ schedules s_j.
3. D_{a_i}: downlinks (all mandatory)
4. C_{a_i}: constraints
    1. Memory constraint: can't exceed memory capacity and must downlink
    2. Non-overlapping schedule: tasks can't be scheduled to overlap

DCOP Formulation

Distributed constraint optimization problem (DCOP): agents coordinate parameters to maximize sum of reward functions

$$X^* = \operatorname{argmax}_X \sum_{f \in \mathcal{F}} f\left(X^{\mathcal{S}}\right)$$

We can formulate our problem with the following reward functions:

$$\mathcal{F}(X) = \bigcup_{a_i \in A} f_{a_i}(X_{a_i}) \cup \bigcup_{r_j \in R} f_{r_j}(X_{r_j})$$

where

$$f_{a_i}(X_{a_i}) = \begin{cases} 0 & C_{a_i} \text{ satisfied by } X_{a_i} \\ -\infty & \text{else} \end{cases}$$

and

$$f_{r_j}(X_{r_j}) = 1 - \prod_{x \in X_{r_j}} (1 - x)$$

Problem Features (1/2)

- Day long horizons with 1000s of requests, 100s of agents
- Most agents get one or more opportunities to service a request
  - Request observation intervals are long (e.g. image every 6 hours)
  - Satellite orbital period's are short (90 minutes)
- Constraint graph composed of cliques: $|X_{a_i}|=\Omega(|R|)$ and $|X_{r_j}|=\Omega(|A|)$

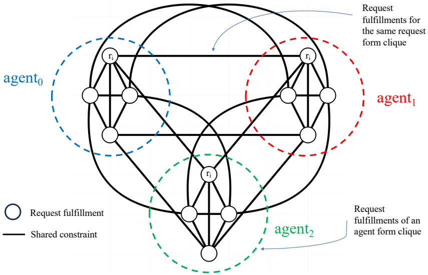


Figure: Example constraint graph with $|A| = 3$ and $|R| = 4$.

Problem Features (2/2)

- Cannot assume agents know other agent tasks and constraints
  - Satellites know existence of other satellites
  - Satellites do not know other agent resources or tasks

Problem Model	Variables Per Agent	Known Agents	Known Constraints
DCOP	1	Neighboring Agents	Neighboring Agents
COSP	Many	All Agents	Self Only

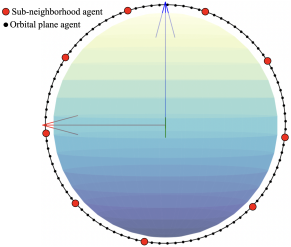
Table: Differences between DCOP assumptions and COSP assumptions.

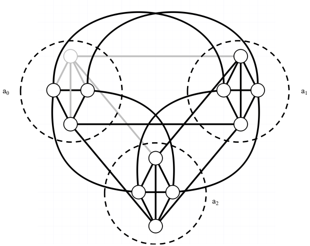
Previous Work

- Decentralized solutions:
  - Auction-based: centralized auctioneer and polynomial messages in agents and requests [Phillips and Parra, 2021, Picard, 2021]
  - Heuristic search: polynomial messages per iteration, no convergence guarantees [Parjan and Chien, 2023]
- DCOP algorithms [Fioretto et al., 2018]:
  - Scale and structure of problem make existing DCOP algorithms infeasible
  - Complete algorithms: computational complexity $O(2^{|A|\cdot|R|}) \rightarrow 10^{30102}$
  - Incomplete algorithms: time and message complexity $O(|A|\cdot|R|) \rightarrow 10^{7}$ per iteration
  - Many algorithms violate COSP assumption of not knowing other agent constraints

Heuristic Approach

- Decompose problem by removing nodes in constraint graph
- Each agent individually computes heuristic decomposition
- *Geometric Neighborhood Decomposition* (GND) heuristic:
  1. Supply computation: estimate number of satellite overflights based on view geometry
  2. Intra-neighborhood delegation: divide requests among fixed neighborhoods of agents based on supply and geometry
  3. Inter-neighborhood: further divide requests into small (parameterized) neighborhoods





Neighborhood Stochastic Search Algorithm (NSS)

- NSS is a DCOP algorithm that uses a decomposition heuristic to decompose and solve sub-problems; conforms to COSP assumptions
- NSS algorithms derived from DSA/BD algorithms [Parjan and Chien, 2023]; adaptations of incomplete DCOP algorithms [Fioretto et al., 2018]

Algorithm	**Computation**	**Communication**
NSS	$\|R\| + k\|R_{\mathcal{N}}\|\|A_{\mathcal{N}}\|$	$k\|R_{\mathcal{N}}\|\|A_{\mathcal{N}}\|$
DSA/BD	$k\|R\|\|A\|$	$k\|R\|\|A\|$

Table: $O(\cdot)$ complexity of decentralized algorithms. $R_{\mathcal{N}}, A_{\mathcal{N}}$ define largest sub-problem, k is number of iterations.

Results: Small Problem Instances

- Compute optimal solution using B&B for small problems
- Results confirm theoretical analysis of NSS and show near-optimal performance

Algorithm	Opt. Gap (%)	Time (ms)	Messages (KB)
Random	3.856	< 1	0
Greedy Start Time	5.152	< 1	0
Portfolio Gredy	4.544	< 1	0
NSS-Random	3.187	3.375	1,4303.712
NSS-GND(1)	3.072	< 1	106.718
NSS-GND(2)	1.713	< 1	181.392
NSS-GND(4)	1.784	< 1	207.886
NSS-GND(8)	1.784	< 1	207.886
BD	2.160	10.775	11,895.264
SWO	0.020	644.925	0.489
B&B	0.0	373,174.350	0.489

Algorithm	Opt. Gap (%)	Time (ms)	Messages (KB)
Random	0.188	< 1	0
Greedy Start Time	3.980	< 1	0
Portfolio Gredy	0.284	< 1	0
NSS-Random	0.632	3.02	604.730
NSS-GND(1)	1.192	< 1	122.336
NSS-GND(2)	0.308	< 1	234.564
NSS-GND(4)	0.236	1.08	385.093
NSS-GND(8)	0.272	1.46	411.798
BD	0.408	4.18	2,451.242
SWO	0.0	305.98	0.499
B&B	0.0	1,129.0	0.499

Table: Results of algorithms on 50 small problems ($<$ 500 requests) for the Planet constellation (left) and Walker constellation (right).

Results: Large Problem Instances

- NSS achieves comparable utility to centralized baselines and DSA, outperforms other baselines

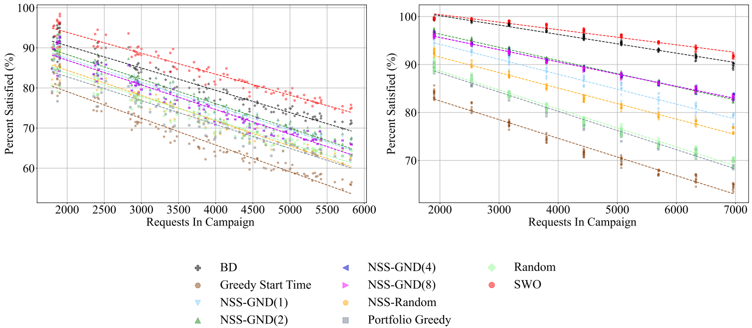


Figure: Request satisfaction of each algorithm across 100 large problem instances presented for Planet (left) and Walker (right).

Results: Large Problem Instances

- NSS uses orders of magnitude lower computation time per agent compared to DSA/BD and centralized solver

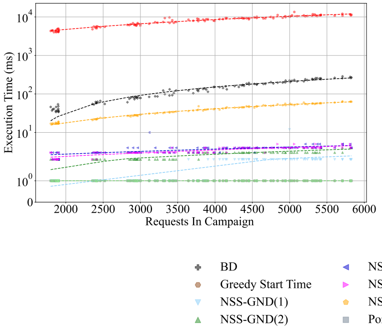


Figure: Execution time of each algorithm across 100 large problem instances presented for Planet (left) and Walker (right).

Results: Large Problem Instances

- NSS uses orders of magnitude lower total message volume compared to DSA/BD

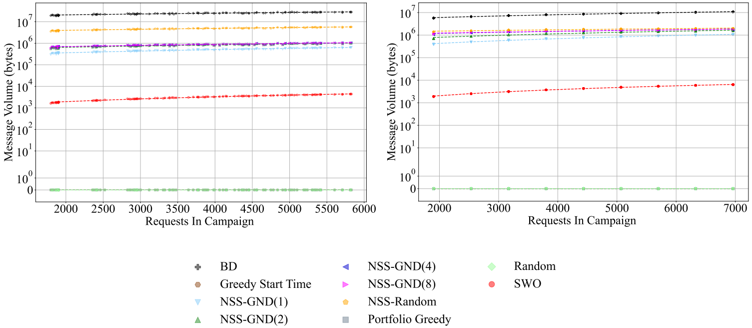


Figure: Message volume of each algorithm across 100 large problem instances presented for Planet (left) and Walker (right)

Conclusion

- NSS addresses drawbacks of existing DCOP methods for COSP
- Future work: dynamic problems, federated systems, flight demos

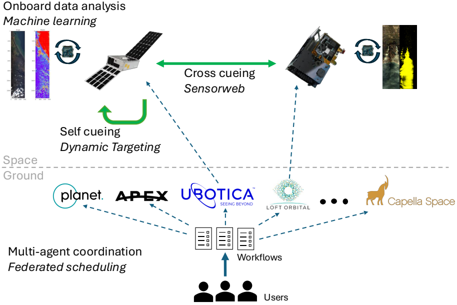


Figure: Overview of NASA FAME demonstration[1].

[1] *Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.*

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. Government sponsorship acknowledged.

References I

[Augenstein et al., 2016] Augenstein, S., Estanislao, A., Guere, E., and Blaes, S. (2016).
Optimal scheduling of a constellation of earth-imaging satellites, for maximal data throughput and efficient human management.
In *Proc. of ICAPS*, volume 26, pages 345–352.

[Boerkoel et al., 2021] Boerkoel, J., Mason, J., Wang, D., Chien, S., and Maillard, A. (2021).
An efficient approach for scheduling imaging tasks across a fleet of satellites.
In *Proc. of IWPSS*.

[Chien et al., 2024] Chien, S., Candela, A., Zilberstein, I., Rijlaarsdam, D., Hendrix, T., and Dunne, A. (2024).
Leveraging commerical assets, edge computing, and near real-time communications for an enhanced New Observing Strategies (NOS) flight demonstration.
In *Proc. of IGARSS*.

References II

[Chien et al., 2025] Chien, S., Zilberstein, I., Candela, A., Rijlaarsdam, D., Perrocheau, A., Dunne, A., Hendrix, T., Grauc, O. C., i Mestrec, A. G., Bovec, M. P., Aragon, O., and Miquel, J. P. (2025).
Flight of dynamic targeting on CogniSAT-6 - Update.
In *Proc. of the Intl. Conference on Space Operations*.

[Fioretto et al., 2018] Fioretto, F., Pontelli, E., and Yeoh, W. (2018).
Distributed constraint optimization problems and applications: A survey.
Journal of Artificial Intelligence Research, 61:623–698.

[Globus et al., 2004] Globus, A., Crawford, J., Lohn, J., and Pryor, A. (2004).
A comparison of techniques for scheduling earth-observing satellites.
In *Proc. of IAAI*.

[Nag et al., 2018] Nag, S., Li, A. S., and Merrick, J. H. (2018).
Scheduling algorithms for rapid imaging using agile cubesat constellations.
Advances in Space Research, 61(3):891–913.

References III

[NewSpace, 2023] NewSpace (2023).
Newspace constellations.
https://www.newspace.im.

[Parjan and Chien, 2023] Parjan, S. and Chien, S. A. (2023).
Decentralized observation allocation for a large-scale constellation.
Journal of Aerospace Information Systems, pages 1–15.

[Phillips and Parra, 2021] Phillips, S. and Parra, F. (2021).
A case study on auction-based task allocation algorithms in multi-satellite systems.
In *Proc. of AIAA Scitech*.

[Picard, 2021] Picard, G. (2021).
Auction-based and distributed optimization approaches for scheduling observations in satellite constellations with exclusive orbit portions.

References IV

[Shah et al., 2019] Shah, V., Vittaldev, V., Stepan, L., and Foster, C. (2019).
Scheduling the world's largest earth-observing fleet of medium-resolution imaging satellites.
In *Proc. of IWPSS*, pages 156–161.

[Zilberstein et al., 2024] Zilberstein, I., Candela, A., Chien, S., Rijlaarsdam, D., Hendrix, T., Buckley, L., and Dunne, A. (2024).
Demonstrating onboard inference for Earth science applications with spectral analysis algorithms and deep learning.
In *Proc. of ISAIRAS*.

[Zilberstein et al., 2025] Zilberstein, I., Rao, A., Salis, M., and Chien, S. (2025).
Decentralized, decomposition-based observation scheduling for a large-scale satellite constellation.
Journal of Artificial Intelligence Research, 82:169–208.