Toward Automated Operational Assist in Satellite Fleet Management via Organizational MARL

Julien Soulé

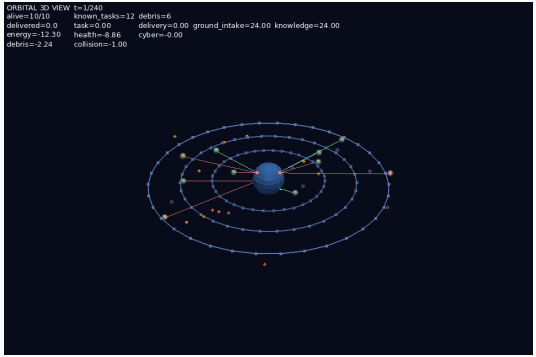
MASSpace @ AAMAS 2026
May 26, 2026

SEDAN Research Group – SnT
University of Luxembourg

An increasingly difficult satellite-fleet monitoring problem

- Dynamic observation demand
- Short communication windows
- Limited energy and buffer capacity
- Cyber and jamming events
- Debris and collision-risk pressure

Automated support with controllability and explainability for partial assistance?



Illustrative view of cooperative satellites operating under task, relay, and safety pressure.

Related works and gaps

Family	Strengths	Gaps
(A) Satellite planning, scheduling, digital twins	operational constraints; optimized plans; domain realism	few reusable decentralized-learning and audit benchmarks
(B) DCOP and planning	explicit constraints; coordination; guarantees	adaptation remains difficult under partial observability and disturbances
(C) Dec-POMDP, MARL	decentralized learning; adaptation	controllability and XRL are often left to external methods
(D) **Constrained/Guided MARL**	norms, convergence, stability, XRL/XMARL	**promising fit, but no satellite-fleet simulator and ready-to-use models**

A: (Ferrari et al., 2025; Rocha et al., 2025; Govoni et al., 2026). B: (Krigman et al., 2024; Hoang et al., 2022). C: (Oliehoek and Amato, 2016; Rashid et al., 2018; Yu et al., 2022). (D) **(Ferber et al., 2003; Hübner et al., 2002; Hubner, Jomi F et. al., 2007; Puiutta and Veith, 2020; Soulé et al., 2025; Garcia and Fernandez, 2015; Alshiekh et al., 2018)**.

Hypotheses and contributions

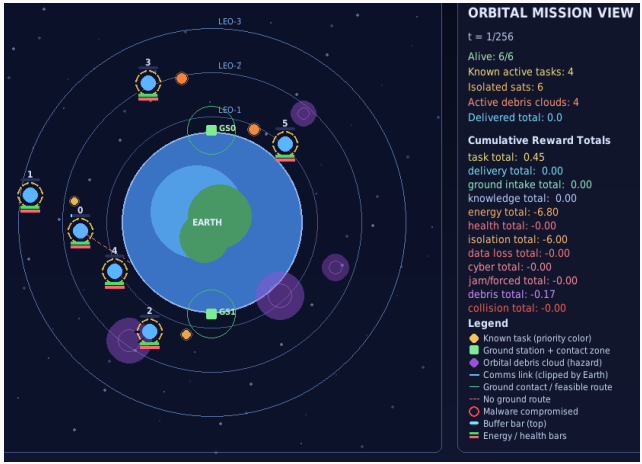
Organizational MARL approach for Earth-observation satellite-fleet management

- agents automatically coordinate under environmental constraints.
- inject prior knowledge into MARL $\rightleftarrows$ extract posterior organizational insights

We propose:

- A controlled benchmark for fleet-level operational-assist experiments.
- An adapted organizational MARL framework with roles and goals for prior knowledge injection and post-hoc analysis.

Orbital Resilient Benchmark for Interactive Task-aware Autonomous Learning



View of ORBITAL: each satellite is an agent, receives observations and makes decisions to support acquisition (go toward *observation tasks*); delivery (relay to *ground stations*); and stabilization (handle *health*, *energy*, *cyber*, *isolation*). Health decreases due to debris collisions or when satellites are randomly compromised by cyber events. Energy is consumed by actions and can be recovered by low-power mode. Communication is intermittent and can be jammed.

Observation space. 20-dimensional local state.

Family	Features
State and orbit	energy, health, θ, radius, ϕ, sunlight
Communication	direct ground contact, network route, local degree, jamming
Data and tasks	buffer load, remaining capacity, task and priority
Safety and cyber	debris density, collision risk, compromised state, recent scan
Fleet context	compromised-neighbor ratio and alive-satellite fraction

Action space. Scalar action between 0 and 7.

Action	Main effect
`observe`	services a local task and fills the buffer (energy ↓)
`relay_ground`	delivers data and syncs task knowledge via ground (energy ↓)
`relay_sat`	shares knowledge or transfers data to a neighbor (energy ↓)
`orbit_down/up`	changes radius, connectivity, and debris exposure (energy ↓)
`lowpower`	reduces consumption and can recover energy
`cyberscan`	mitigates compromise risk (energy ↓)
`idle`	waits while orbital dynamics continue

Acquire, deliver, stabilize

Operational tension

- Acquire
  - useful only if the data can later be delivered;
  - limited by task knowledge, position, energy, and buffer.
- Deliver
  - high value through ground contact;
  - depends on relay paths and timing.
- Stabilize
  - protects energy, health, cyber state, and debris safety;
  - often delays acquisition or delivery.

Reward rule	Component	Reward effect
Ground relay with buffered data and valid contact	delivery	$+1000\times$ delivered data
Observe active, known, reachable task	task	$+100\times$ observed data
Task knowledge discovered or shared	knowledge	$+10\times$ knowledge gain
Ground contact imports catalog tasks	intake	$+1\times$ new tasks
Any action consumes energy	energy	$-0.05\times$ energy spent
Observation exceeds buffer capacity	overflow	$-0.4\times$ lost data
Destroyed satellite loses buffered data	data loss	$-0.8\times$ lost data
Malware, drag, or collision reduces health	health	$-0.8\times$ health loss
Living satellite has no communication neighbor	isolation	$-0.3\times$ isolated satellites
Dead satellites remain in the fleet	failure	$-1.0\times$ dead satellites
Compromise, jamming, forced action, drag, debris, collision	safety costs	-0.4 to $-2.5\times$ event intensity

$\mathcal{M}OISE^+$ and MOISE+MARL

$\mathcal{M}OISE^+$ (Hübner et al., 2002; Hubner, Jomi F et. al., 2007) is a highly formalized organizational model

- structural (roles),
- functional (goals);
- and deontic (permissions/obligations)

Decentralized Partially Observable Markov Decision Process (Dec-POMDP) (Oliehoek and Amato, 2016)

- considers multiple agents in a MAS-like setting
- stochastic processes for uncertainty in environmental changes including observations;
- $(S, \{A_i\}, T, R, \{\Omega_i\}, O, \gamma)$

MOISE+MARL (Soulé et al., 2025) binds $\mathcal{M}OISE^+$ and MARL

- **Roles provide shielding**
  - input: current observation;
  - output: actions allowed at this step;
  - effect: replace forbidden policy actions by allowed ones.
- **Goals provide reward shaping**
  - input: history, current observation, selected action;
  - output: bonus or malus added to the reward;
  - effect: guide learning toward intermediate objectives.
- **Trajectory-based Evaluation in MOISE+MARL (TEMM) provides organizational analysis**
  - input: trajectories;
  - output: structural, functional, and organizational fit scores;
  - effect: post-hoc evidence of organizational traces (org. fit).

ORBITAL-oriented MOISE+MARL

Element	Rule sketch	Effect
`acquirer` role	use `relay_ground` to obtain the task catalog; use `observe` when a known nearby task and buffer space exist	turns ground catalog access into useful task acquisition
`deliverer` role	use `relay_ground` for buffered data; use `relay_sat` when a route or neighbor can help	moves buffered mission data and task knowledge toward delivery
`stabilizer` role	use `scan`, `lowpower`, or orbit actions under cyber, energy, jamming, or debris stress	preserves future action capacity and reduces mission collapse
`acquirer_goal`	bonus actions that match the acquirer logic; small malus when a relevant acquisition action is ignored	makes catalog intake and observation easier to learn
`deliverer_goal`	bonus actions that match the deliverer logic; small malus when a relevant delivery action is ignored	links local relay decisions to final mission value
`stabilizer_goal`	bonus actions that match the stabilizer logic; small malus when a relevant safety action is ignored	encourages recovery from cyber, energy, and debris stress

Partial roles impose their handcrafted action with constraint hardness 0.3; full roles impose it with hardness 1.0.

Baselines

Family	Condition	What it tests
Rule-Based	`handcrafted`	A fully scripted acquire–deliver–stabilize loop; interpretable operational control without learning.
Rule-Based	`rb_deliverer`	Delivery continuity first: full `deliverer`, with softer acquisition and stabilization.
Rule-Based	`rb_dcop_like`	DCOP-inspired structure: full `acquirer` and `deliverer`, soft `stabilizer`.
Rule-Based	`rb_acquirer`	Acquisition-first control: full `acquirer`, with softer delivery and stabilization.
Learning-Based	`lb_unconstrained`	Same MARL backbone with the native ORBITAL reward only; tests adaptation without explicit organization.
Learning-Based	`lb_reward_only`	Goal-reward guidance only; isolates whether shaping can induce mission discipline without shielding.
Learning-Based	`lb_action_only`	Role-action shielding only; isolates structural control without extra mission rewards.

The comparison separates interpretable rule-based control from adaptive learning-based ablations.

Experimental setup

Software

- ORBITAL[a] is a PettingZoo environment(Terry et al., 2021)
- The MOISE+MARL framework and baselines[b] are implemented within BenchMARL(Bettini et al., 2024)

Hardware (NVIDIA DGX Spark Version 7.4.0)

CPU	20 physical/logical cores at 3354 MHz
RAM	119.69 GB total
GPU	NVIDIA GB10, 48 SMs, 119.69 GB VRAM

[a] https://github.com/julien6/ORBITAL
[b] https://github.com/julien6/BenchMARL

Hyperparameters

Selection	HPO with BenchMARL sweeps on validation seeds • MAPPO (Yu et al., 2022), QMIX (Rashid et al., 2018), MASSAC (Pu et al., 2021)
Budget	same episode horizon and optimization budget across learning conditions
Seeds	disjoint training, selection, and final-evaluation seeds

Metrics

Mission	return, delivery volume, and mission success
Robustness	energy stress, isolation, failures, cyber impact
Control	role-violation rate and consistency indicators
Audit	structural fit, functional fit, and organizational fit

Results overview

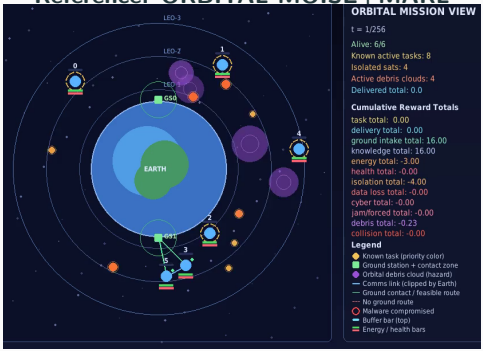
Metric values normalized over experimental values across all baselines for MAPPO(Yu et al., 2022)

Condition	Final return	Std. dev.	Robustness	Convergence	Rule violation	OF score
`handcrafted`	0.82	0.04	0.47	n/a	n/a	n/a
`rb_deliverer`	0.67	0.04	0.50	n/a	n/a	n/a
`rb_dcop_like`	0.60	0.05	0.53	n/a	n/a	n/a
`rb_acquirer`	0.64	0.04	0.45	n/a	n/a	n/a
`lb_unconstrained`	**0.95**	0.17	**0.86**	0.50	0.74	0.52
`lb_reward_only`	0.90	0.15	0.85	0.58	0.63	0.67
`lb_action_only`	0.83	0.12	0.77	0.68	0.31	0.79
`lb_moise_marl`	0.85	**0.08**	0.80	**0.70**	**0.18**	**0.91**

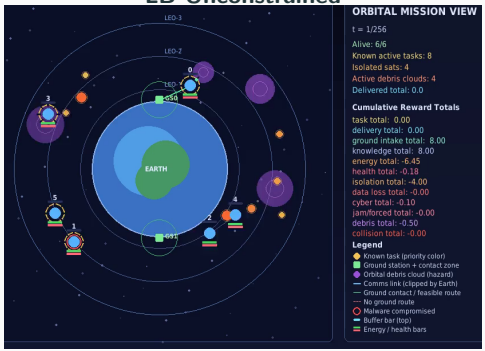
Finding a trade-off between constraining too much (rule-based) and constraining too little (unconstrained learning).

Full MOISE+MARL vs LB-Unconstrained

Reference: ORBITAL-MOISE+MARL



LB-Unconstrained



- LB-Unconstrained uses the same MARL backbone without organizational roles or goals.
- Full MOISE+MARL appears easier to audit when communication, energy, and cyber pressure interact.

Training, evaluation, and TEMM traces are summarized in Wandb:
https://wandb.ai/julien-soule-university-of-luxembourg/benchmarl?nw=nwuserjuliensoule.

Conclusion, limits, and perspectives

- Proposed the **ORBITAL benchmark** for satellite-fleet operational-assist research.
- Developed an **organizational MARL framework** for satellite-fleet guidance and analysis.
- Organizational priors can improve performance quickly with explainability and control.

Limits and future work

- Organizational priors can improve auditability, but overly rigid roles may reduce adaptation (trade-off).
- ORBITAL remains an abstraction of orbital mechanics, communication, and operational procedure.
- The framework has computational overhead because several baselines, sweeps, and trajectory analyses are needed.
- TEMM still requires careful audit, semantic interpretation, and manual intervention.

Thank You

Questions?

julien.soule@uni.lu

References

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. 2018. Safe Reinforcement Learning via Shielding. *Proceedings of the AAAI Conference on Artificial Intelligence* 32, 1 (Apr. 2018). https://doi.org/10.1609/aaai.v32i1.11797

Matteo Bettini, Amanda Prorok, and Vincent Moens. 2024. BenchMARL: Benchmarking Multi-Agent Reinforcement Learning. *Journal of Machine Learning Research* 25, 217 (2024), 1–10. http://jmlr.org/papers/v25/23-1612.html

Jacques Ferber, Olivier Gutknecht, and Fabien Michel. 2003. Agent/Group/Roles: Simulating with Organizations. In *ABS 2003 - 4th International Workshop on Agent-Based Simulation*, J.P. Muller (Ed.). Montpellier, France. https://hal-lirmm.ccsd.cnrs.fr/lirmm-00269714

Benedetta Ferrari, Jean-François Cordeau, Maxence Delorme, Manuel Iori, and Roberto Orosei. 2025. Satellite Scheduling Problems: A survey of applications in Earth and outer space observation. *Computers & Operations Research* 173 (2025), 106875. https://doi.org/10.1016/j.cor.2024.106875

Javier Garcia and Fernando Fernandez. 2015. A Comprehensive Survey on Safe Reinforcement Learning. *Journal of Machine Learning Research* 16, 42 (2015), 1437–1480. http://jmlr.org/papers/v16/garcia15a.html

References ii

Lorenzo Govoni, Corrado Chiatante, Bjørn Andreas Kristiansen, Tor Arne Johansen, and Andrea Cristofaro. 2026. Optimization-Based Task Allocation for Earth Observation in Multi-Satellite Systems. *Aerospace Science and Technology* 168 (2026), 111058. https://doi.org/10.1016/j.ast.2025.111058

Khoi D. Hoang, Ferdinando Fioretto, Ping Hou, William Yeoh, Makoto Yokoo, and Roie Zivan. 2022. Proactive Dynamic Distributed Constraint Optimization Problems. *Journal of Artificial Intelligence Research* 74 (2022), 179–225.

Jomi Fred Hübner, Jaime Simão Sichman, and Olivier Boissier. 2002. A Model for the Structural, Functional, and Deontic Specification of Organizations in Multiagent Systems. In *Advances in Artificial Intelligence*, Guilherme Bittencourt and Geber L. Ramalho (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 118–128.

Hubner, Jomi F et. al. 2007. Developing organised multiagent systems using the MOISE+ model: programming issues at the system and agent levels. *Int. Journal of Agent-Oriented Software Engineering* (2007), 370. https://doi.org/10.1504/ijaose.2007.016266

Shai Krigman, Tal Grinshpoun, and Lihi Dery. 2024. Scheduling of Earth Observing Satellites Using Distributed Constraint Optimization. *Journal of Scheduling* 27, 5 (2024), 507–524. https://doi.org/10.1007/s10951-024-00816-x

References iii

Frans A. Oliehoek and Christopher Amato. 2016. *A Concise Introduction to Decentralized POMDPs*. Springer. https://link.springer.com/book/10.1007/978-3-319-28929-8

Yuan Pu, Shaochen Wang, Rui Yang, Xin Yao, and Bin Li. 2021. Decomposed Soft Actor-Critic Method for Cooperative Multi-Agent Reinforcement Learning. arXiv:2104.06655 [cs.AI] https://arxiv.org/abs/2104.06655

Erika Puiutta and Eric M. S. P. Veith. 2020. Explainable Reinforcement Learning: A Survey. Springer-Verlag, Berlin, Heidelberg, 77–95. https://doi.org/10.1007/978-3-030-57321-8_5

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 4295–4304. https://proceedings.mlr.press/v80/rashid18a.html

Yure Rocha, Guilherme O. Chagas, Leandro C. Coelho, and Anand Subramanian. 2025. The integrated agile Earth observation satellite scheduling problem. *Computers & Operations Research* 184 (2025), 107212. https://doi.org/10.1016/j.cor.2025.107212

References iv

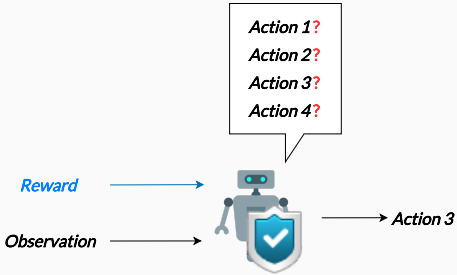
Julien Soulé, Jean-Paul Jamont, Michel Occello, Louis-Marie Traonouez, and Paul Théron. 2025. An Organizationally-Oriented Approach to Enhancing Explainability and Control in Multi-Agent Reinforcement Learning. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems (***AAMAS** *2025)* (Detroit, MI, USA). International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1968–1976.

J K Terry, Benjamin Black, Nathaniel Grammel, Mario Jayakumar, Ananth Hari, Ryan Sullivan, Luis S Santos, Clemens Dieffendahl, Caroline Horsch, Rodrigo Perez-Vicente, Niall Williams, Yashas Lokesh, and Praveen Ravi. 2021. PettingZoo: Gym for Multi-Agent Reinforcement Learning. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 15032–15043. `https://proceedings.neurips.cc/paper_files/paper/2021/file/803f7c4c3ff61b71be53a0c803bfb57f-Paper.pdf`

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and YI WU. 2022. The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 24611–24624. `https://proceedings.neurips.cc/paper_files/paper/2022/file/9c1535a02f0ce079433344e14d910597-Paper-Datasets_and_Benchmarks.pdf`

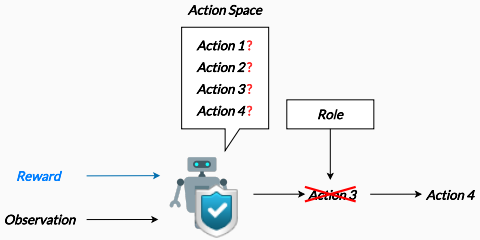
Training

- Select the best actions to maximize cumulative reward



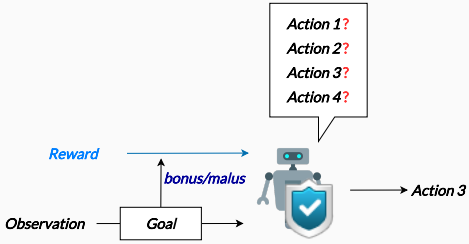
MARL + organization

- Role: enforce / forbid actions → safety guarantee



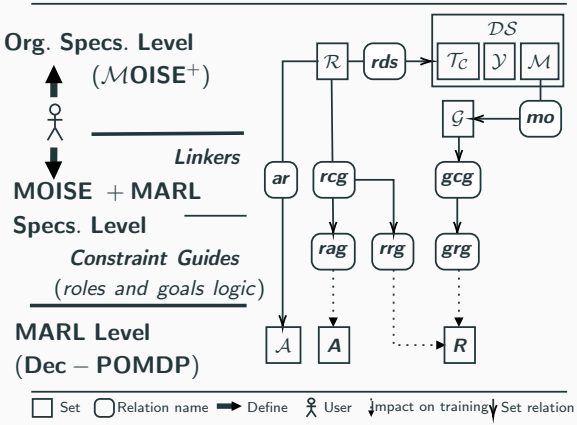
MARL + organization

- Goal: encourage the achievement of an intermediate objective



Training

- Combine Dec-POMDP with $\mathcal{MOISE}^+$.
- Agents $\rightarrow$ role and mission $\rightarrow$ goals.
- Constraint guides $\sim$ "role / goal implementation":
  - **Actions** via *RoleActionGuides* (RAG)
  - **Rewards** via *RoleRewardGuides* and *GoalRewardGuides*



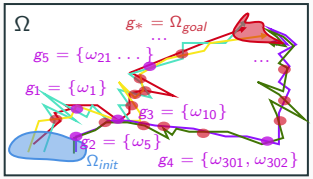
Analysis

Trajectory-based Evaluation in MOISE+MARL (TEMM)

- **Objective**: provide a post-hoc interpretation of agent behavior at the organizational level.
- **Hypothesis**: trajectories are "noisy" variants of a limited number of strategies.

Underlying hypotheses:

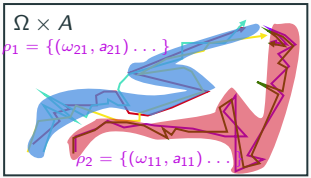
- **Roles** correspond to frequent *(observation, action)* transition patterns in agent trajectories.
- **Goals** correspond to observations frequently received within agent trajectories.



An abstract visualization of observations in the trajectories

Operationalization...

- *Trajectories as vectors;*
- *Distances: Smith-Waterman, LCS, Euclidean...;*
- *Clustering + centroids* $\rightarrow$ *roles / goals*



An abstract visualization of transitions in the trajectories

ORBITAL Encodes the Operational Couplings

Modeled pressures

- non-stationary prioritized tasks;
- finite energy and heterogeneous action costs;
- stochastic communication degradation;
- stochastic compromise events;
- drifting debris-risk zones.

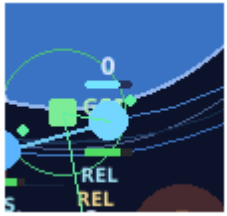
Core doctrine

$$\text{Acquire} \rightarrow \text{Deliver} \rightarrow \text{Stabilize}$$

Experimental value

- local task service can conflict with fleet delivery;
- safety-preserving actions can reduce short-term reward;
- robust behavior requires switching between mission phases;
- these switches can be evaluated in trajectories.

Visual Grammar



Satellite marker

Number = satellite id. Cyan top bar = buffered data. Green bottom bar = energy. Text labels show role and last action.

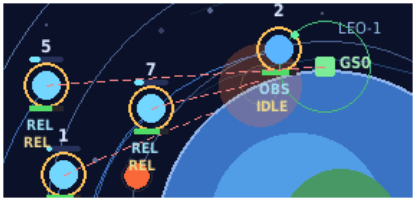
Observation task

Warm dots are active tasks. Brighter/larger means higher priority. A successful OBS action removes the task and fills a buffer.

Mission panel

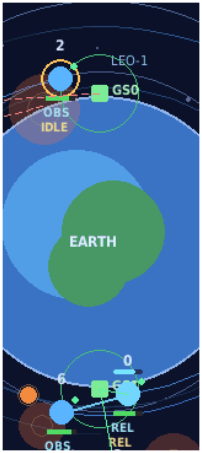
Watch alive satellites, isolated satellites, delivered value, and reward components to understand the global effect.

Connectivity



Blue communication links

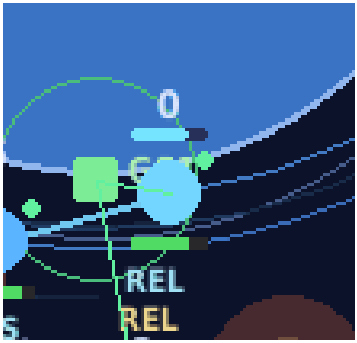
Blue lines mean satellites can communicate at this step. They are not deliveries by themselves; they are possible relay paths.



Green ground/downlink cue

A green square is a ground station. A green downlink line means buffered data currently has a feasible path to ground.

Labels



Role labels

OBS	Observer role: prioritize acquiring task data.
REL	Relay role: prioritize delivery and communication utility.
SAFE	Safety role: preserve energy, cyber health, or debris safety.
FREE	No organizational role constraint.
SOFT	Reward-shaped guidance, but no hard role enforcement.

Action labels

OBS	Observe task.
REL_GRN	Relay to ground. REL_SAT: relay to satellite.
PWR	Low-power safe mode.
SCAN	Cyber scan. UP/DN: orbit maneuver.
IDLE	No active mission operation this step.

Action Transition – Observe

Before

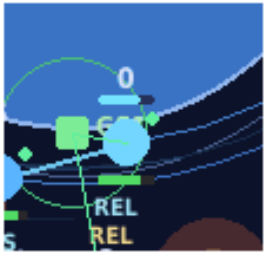
A nearby active task is visible as a warm dot. The satellite must be close enough to sense it.

Action

OBS: the satellite spends energy to service one local task.

Immediate effect

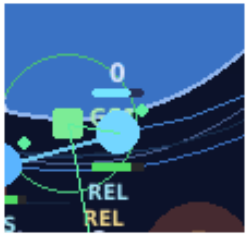
The task becomes inactive and the satellite's cyan buffer increases.



After

The data is only stored. Delivery still requires relay plus a path to ground.

Action Transition – Relay



Before

A satellite has buffered data: cyan bar above the marker.

Actions

REL_GRN: deliver to ground.

REL_SAT: relay to another satellite.

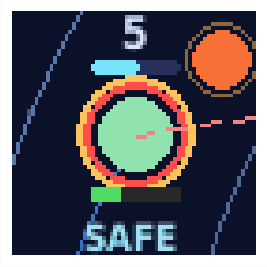
Requirement

Delivery succeeds only with direct ground contact or a communication path through other satellites.

After

If the path is feasible, the buffer decreases and delivered total increases.

Action Transitions – Safety and Waiting



Stress cues

Yellow ring = isolated; red ring = compromised; orange halos = debris.

Before → action → after

PWR	recover or preserve energy; buffer remains stored.
SCAN	reduce or clear compromise risk.
UP/DN	maneuver to reduce local debris conjunction risk.
IDLE	wait without observing, relaying, scanning, or maneuvering.

Why IDLE after Observe?

The satellite may have no task nearby, no delivery path yet, a role constraint, or a reason to save energy.