RÉPUBLIQUE
FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

Multi-agent techniques for resource allocation and planning

Application to Earth observation by satellite constellations

Gauthier Picard
gauthier.picard@onera.fr

ENS Rennes – 07/12/2023

ONERA/DTIS, Université de Toulouse

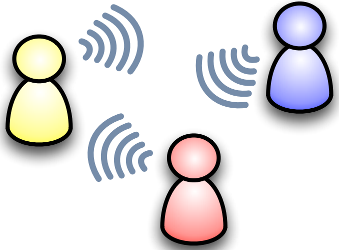
Introduction

Multi-Agent Systems and Distributed Artificial Intelligence

- Agent: An entity that behaves autonomously in the pursuit of goals
- Multi-agent system: A system of multiple interacting agents

An agent is...

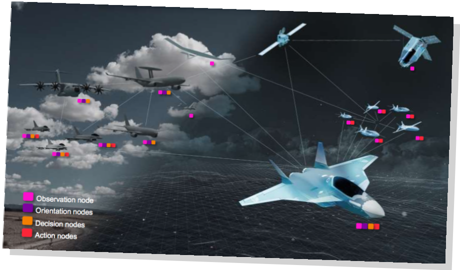
- Autonomous: Is of full control of itself
- Interactive: May communicate with other agents
- Reactive: Responds to changes in the environment or requests by other agents
- Proactive:Takes initiatives to achieve its goals



Introduction

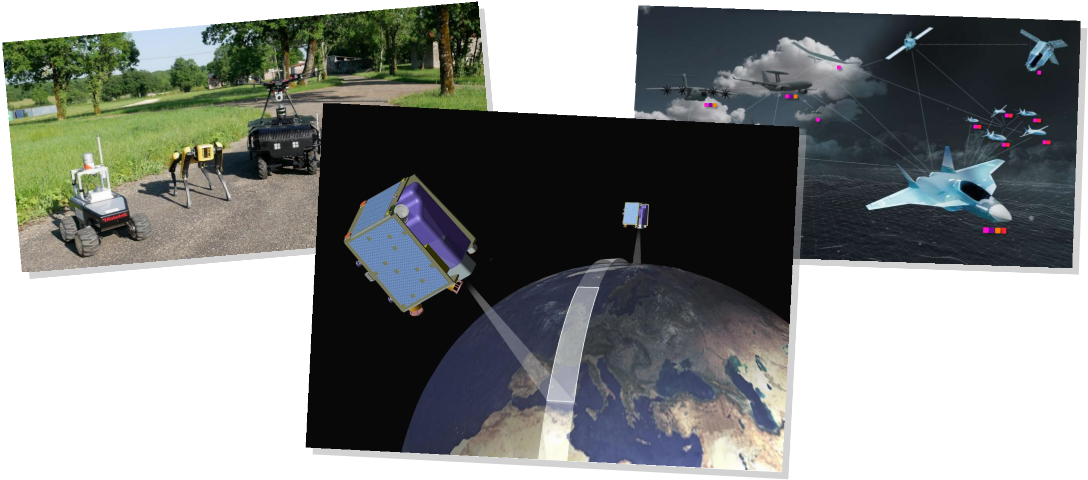
Sample multi-agent systems

Introduction
Sample multi-agent systems



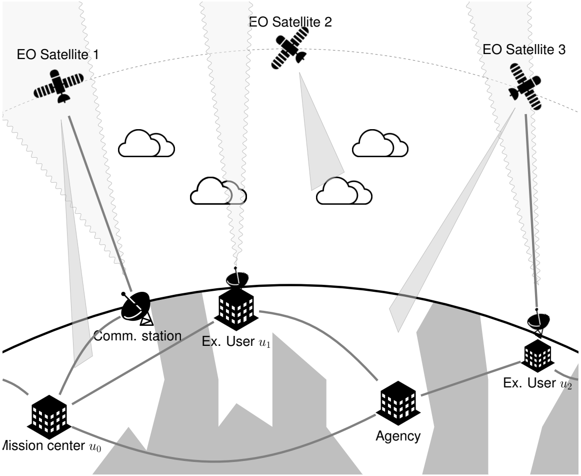
Introduction

Sample multi-agent systems



Introduction

Multi-Agent Decisions for Earth Observation



- Constellation Design
  - How to compose the constellation?
  - How to dimension the constellation?
  - Where to position assets?
- Offline Operations
  - How to allocate resources?
  - How to share resources?
  - How to schedule in a multi-party/multi-mission context?
- Online Operations
  - How to adapt activities facing unpredictable events?
  - Which coordination protocols to use?

Outline

Outline

Challenges in Earth Observation Constellation Operations

[PICARD et al., 2021]

- Recent years have shown a large increase in the development of satellite constellations
- Increasing the size allows to capture any point on Earth at higher frequency, e.g. the Planet Dove constellation
- But, operating numerous Earth observation satellites (**EOS**) requires to *cooperate*, *collectively* solve and schedule, *self*-adapt and *interact*

Many AAMAS-related and Open Research and Technology Questions

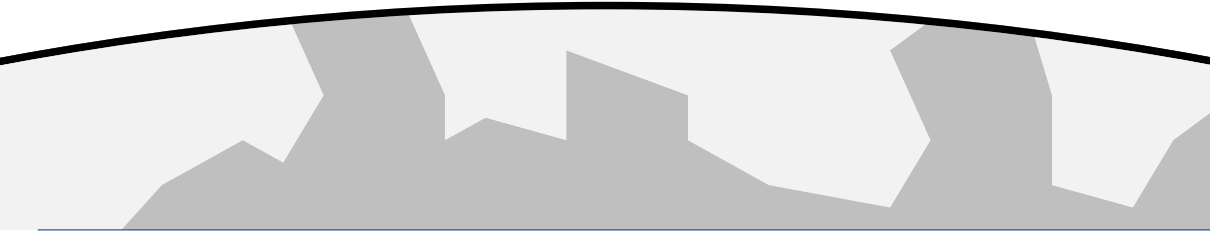
Categories of Challenges

Constellation Design

Offline Operations

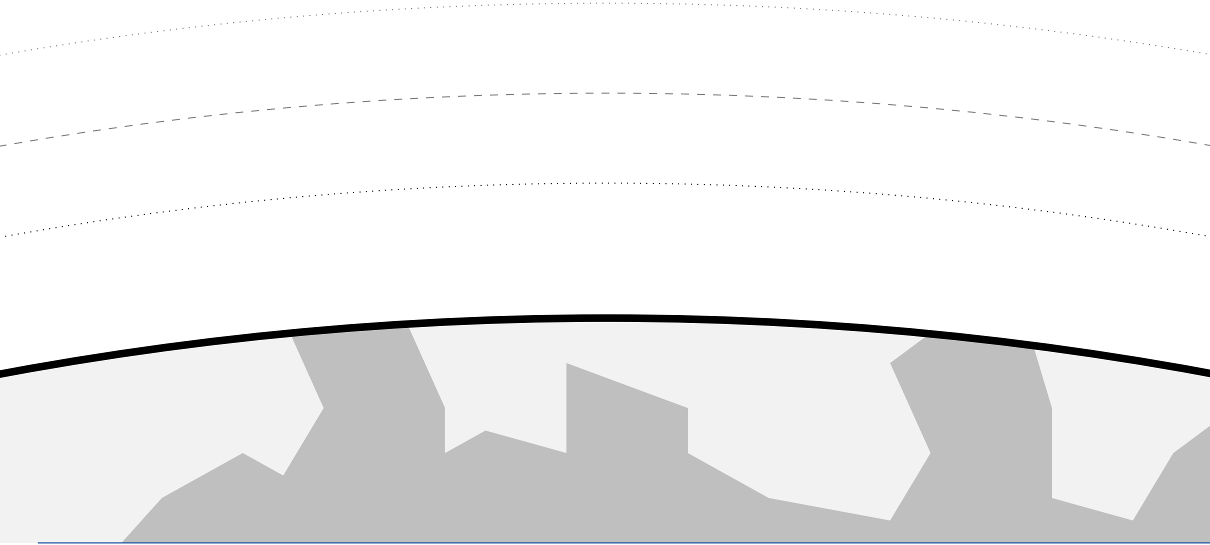
Online Operations

How to Design an EOS Constellation?



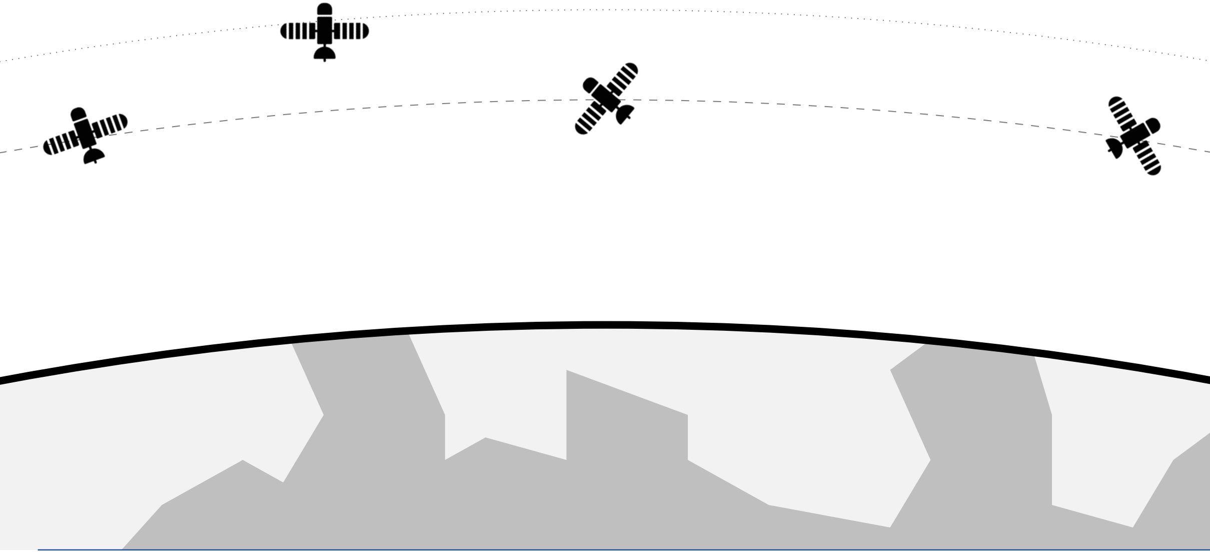
How to Design an EOS Constellation?

Orbits



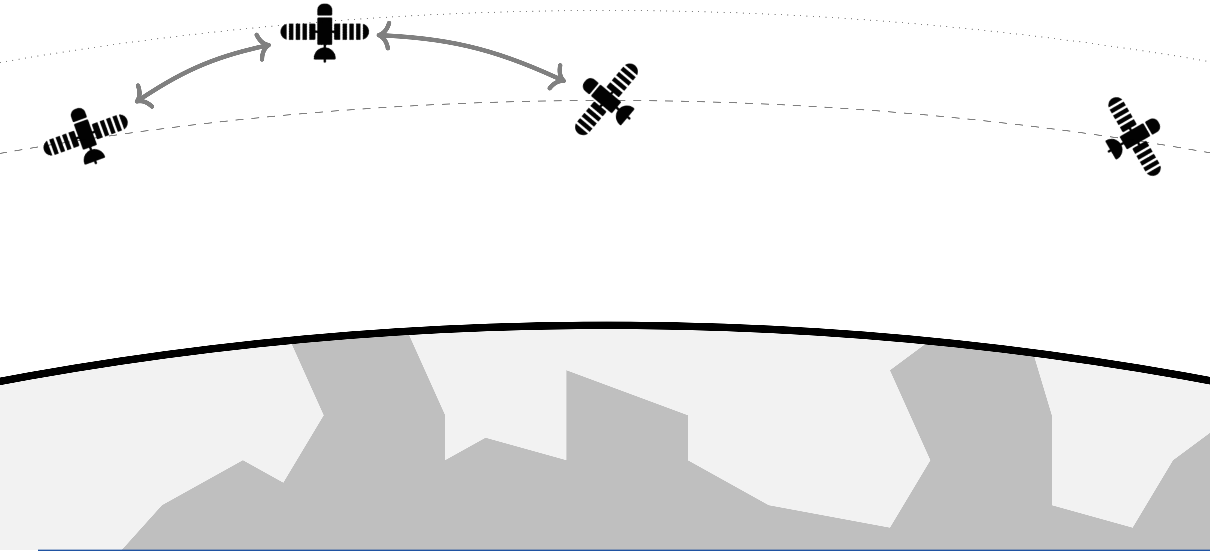
How to Design an EOS Constellation?

Constellation composition



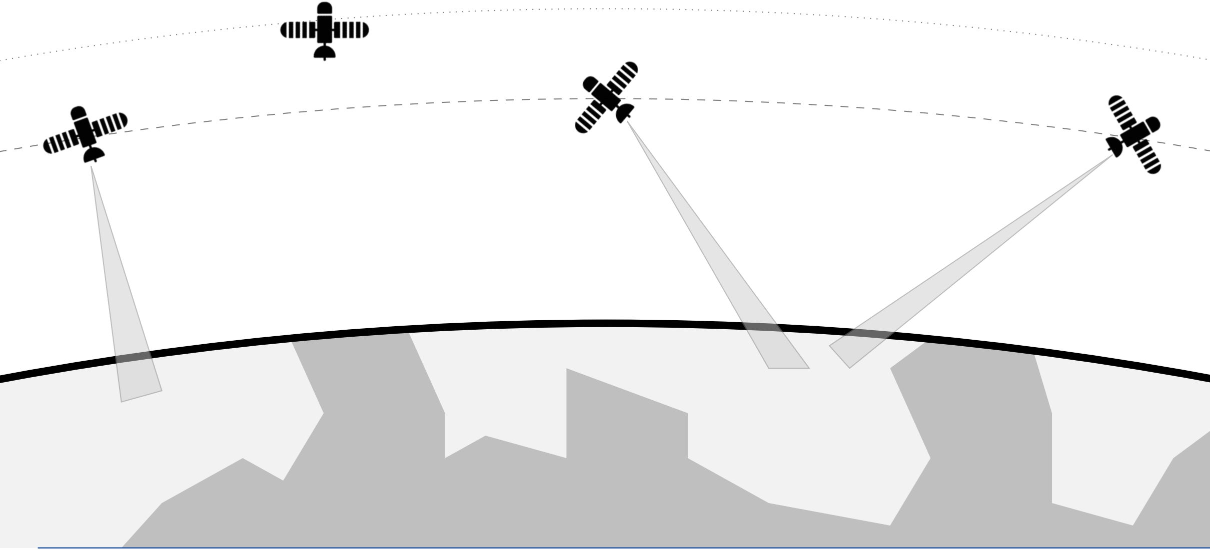
How to Design an EOS Constellation?

Constellation composition



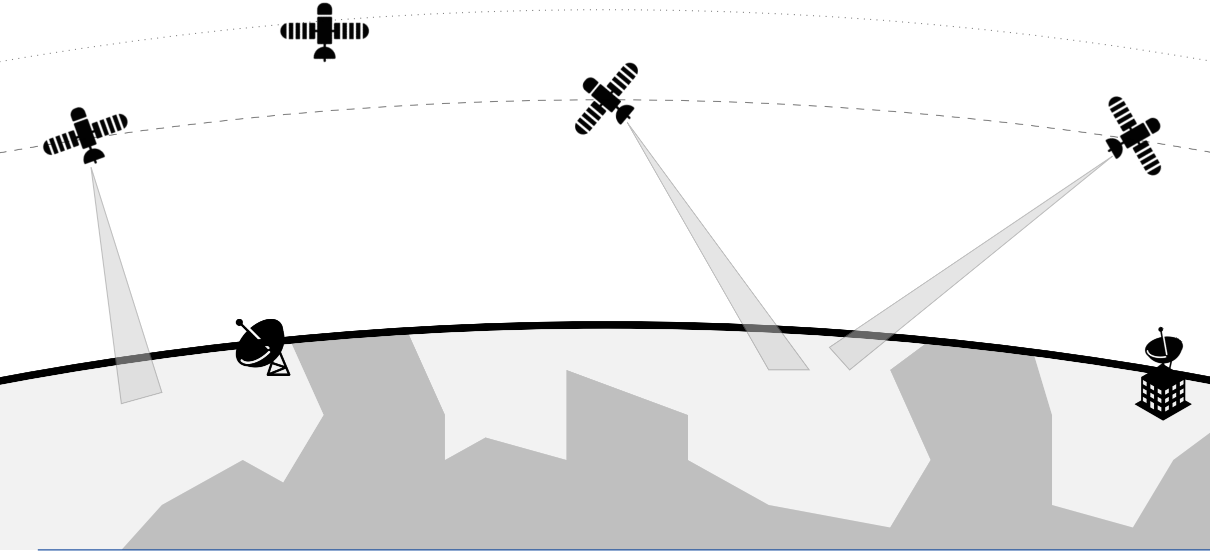
How to Design an EOS Constellation?

Points of interest



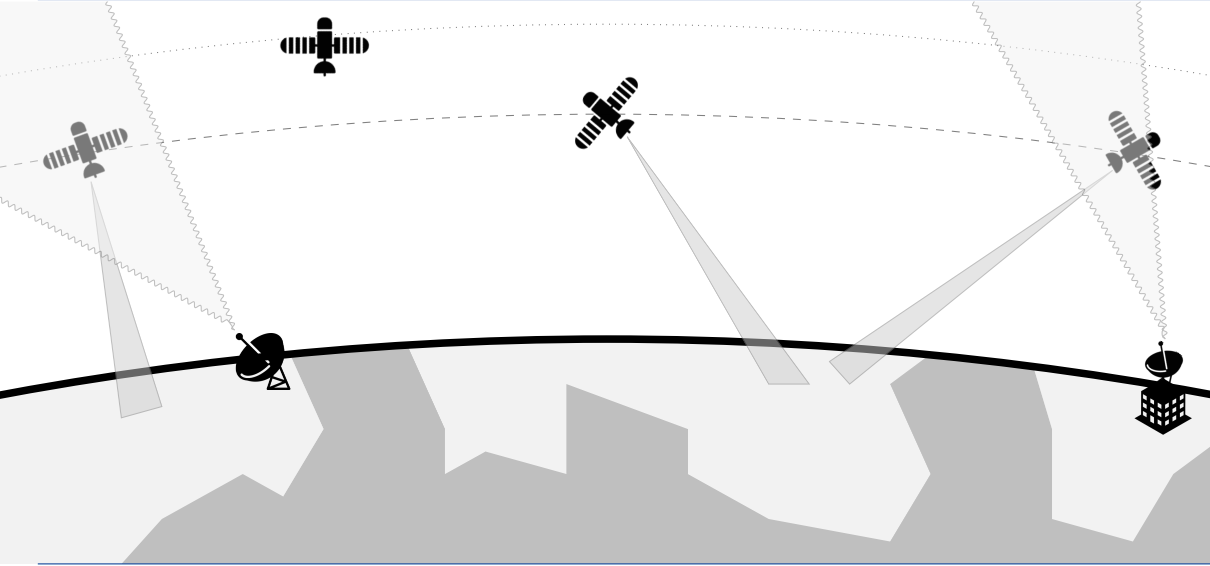
How to Design an EOS Constellation?

On-ground communication stations



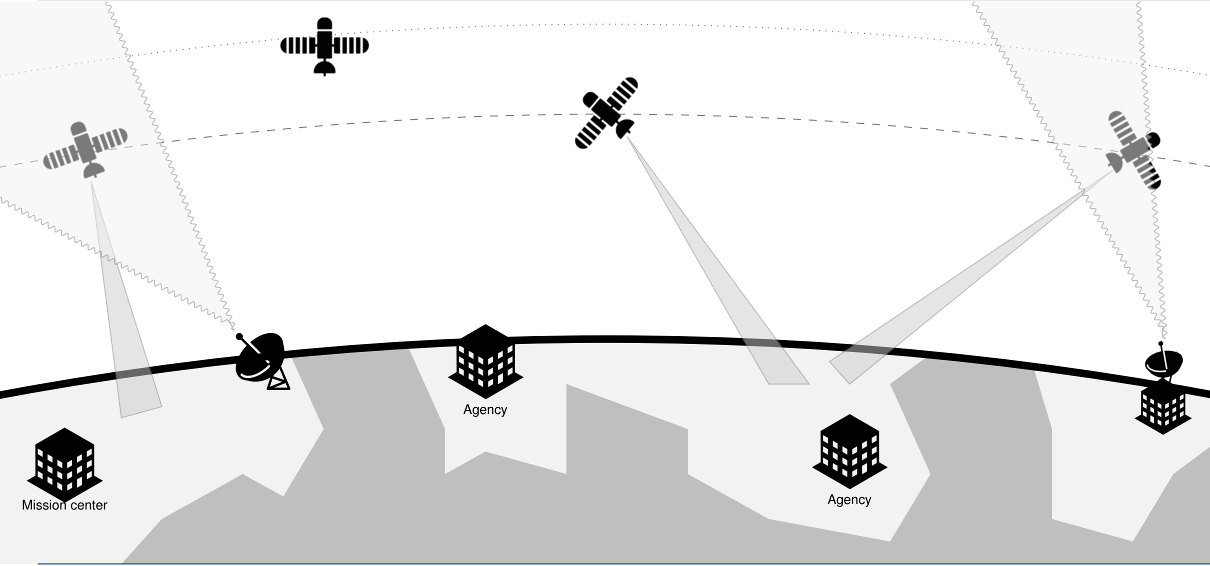
How to Design an EOS Constellation?

Visibility windows



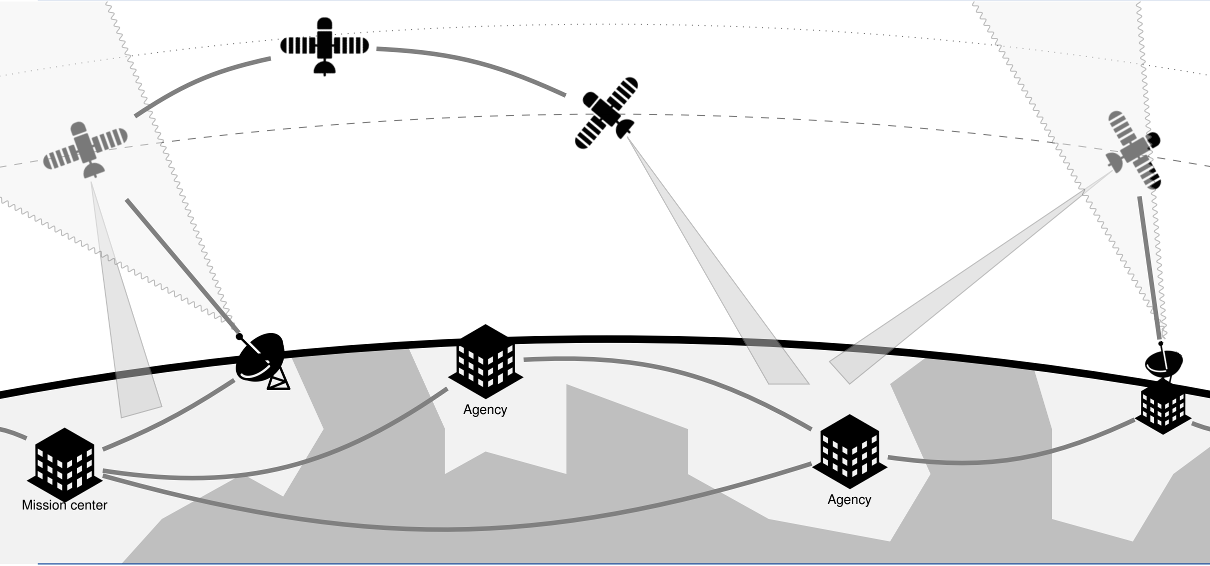
How to Design an EOS Constellation?

Other actors and stakeholders

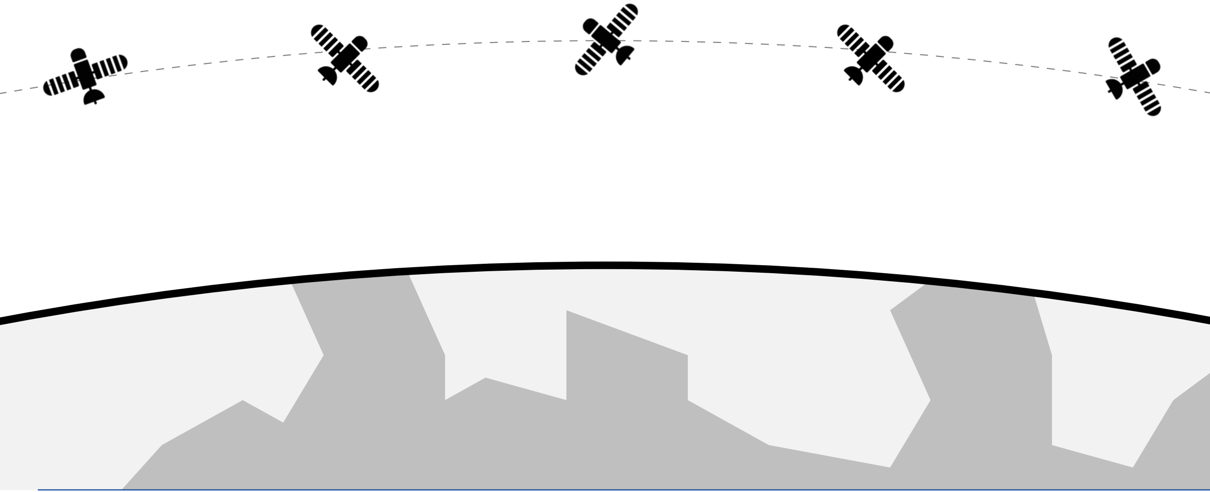


How to Design an EOS Constellation?

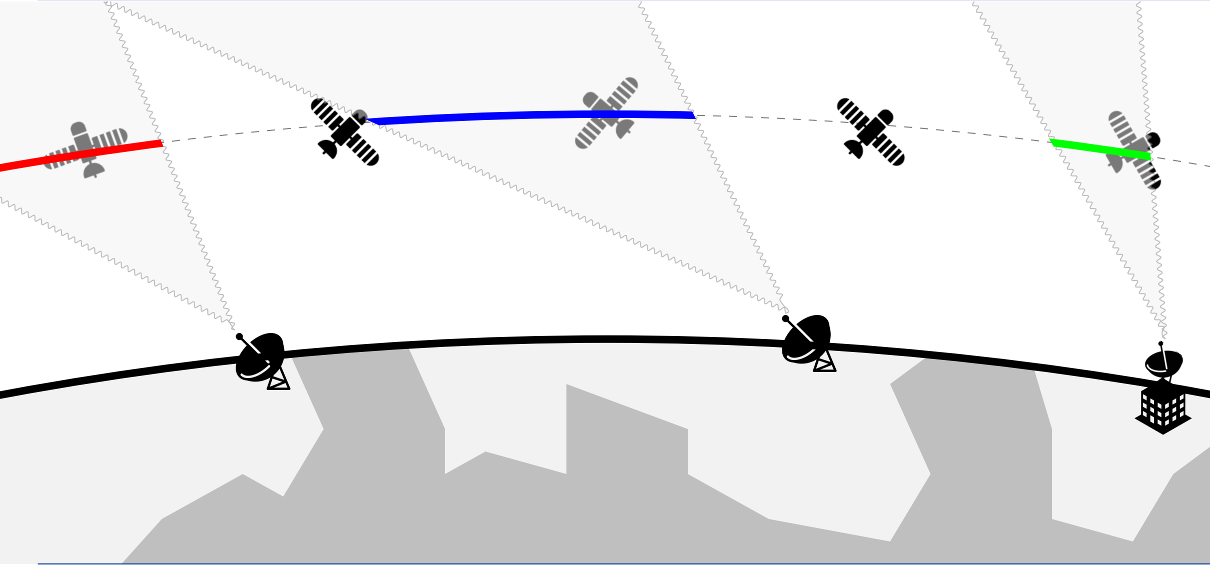
System organization



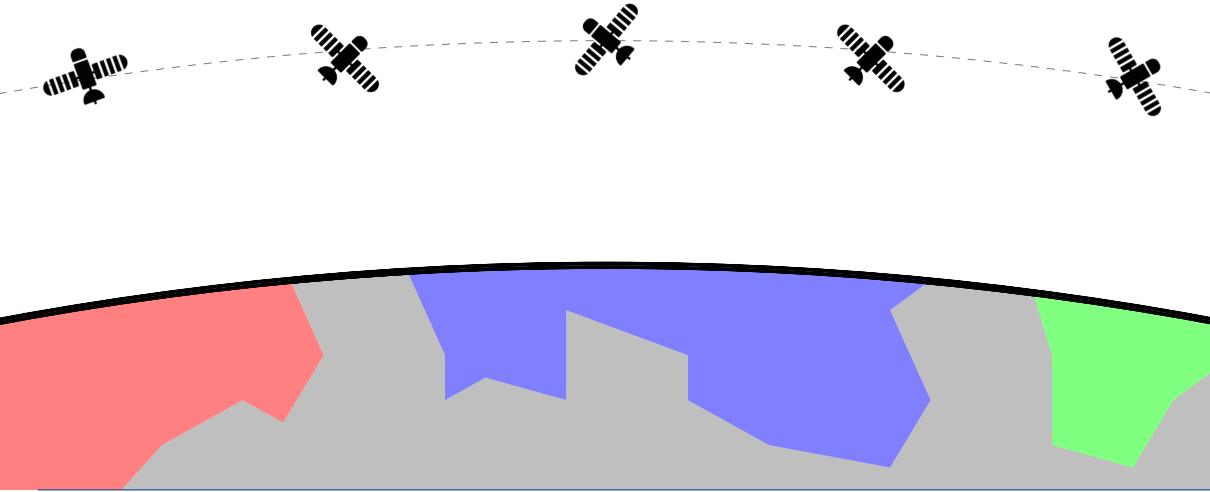
How to Allocate Resources?



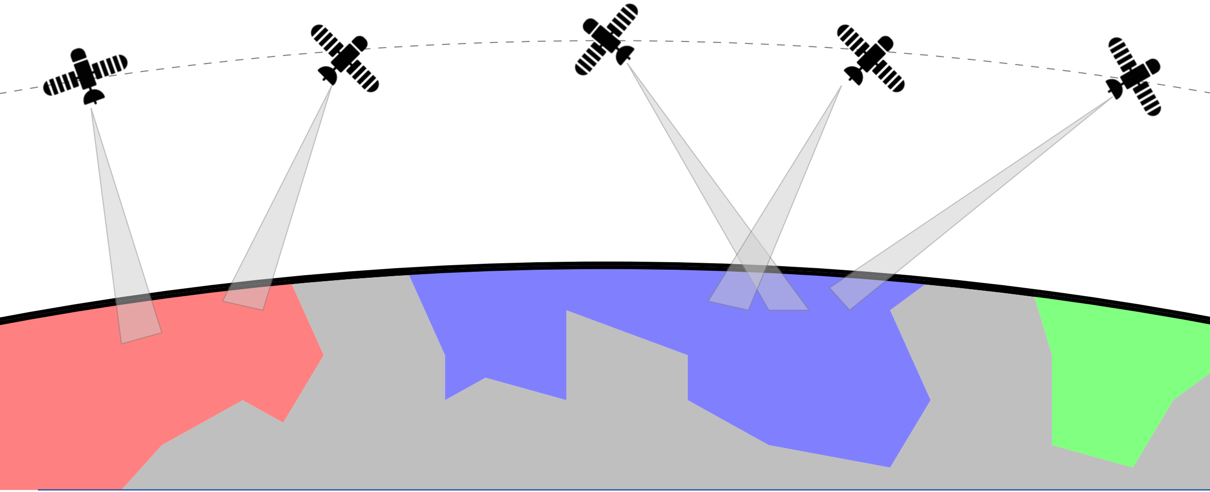
How to Allocate Resources?



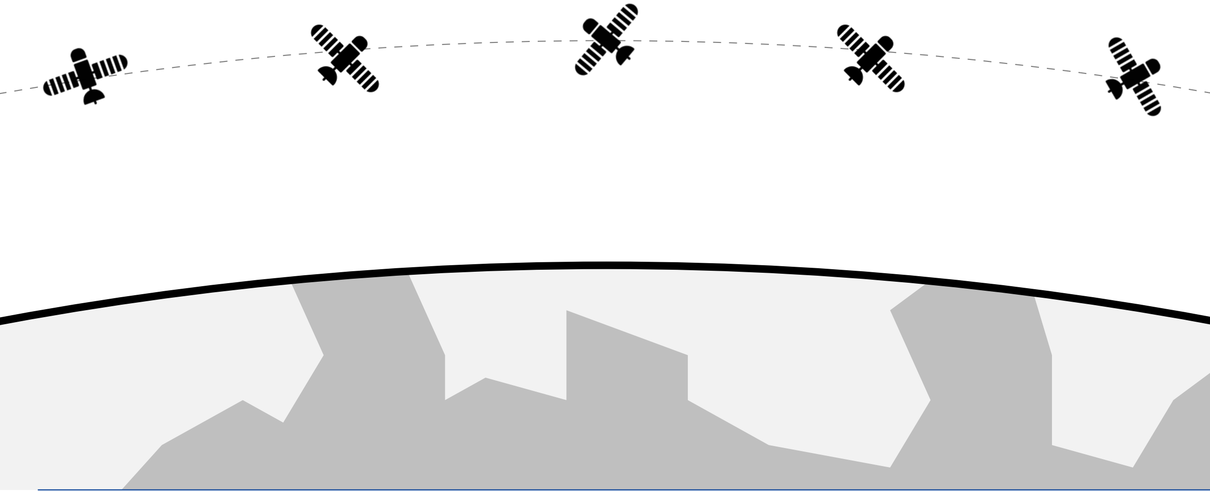
How to Share Resources?



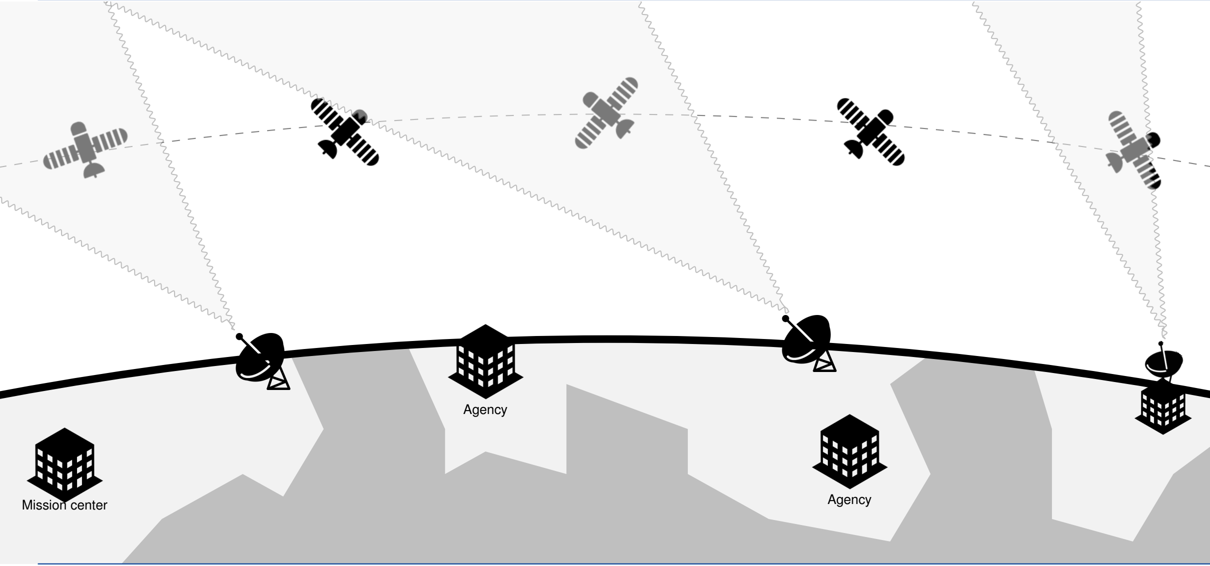
How to Share Resources?



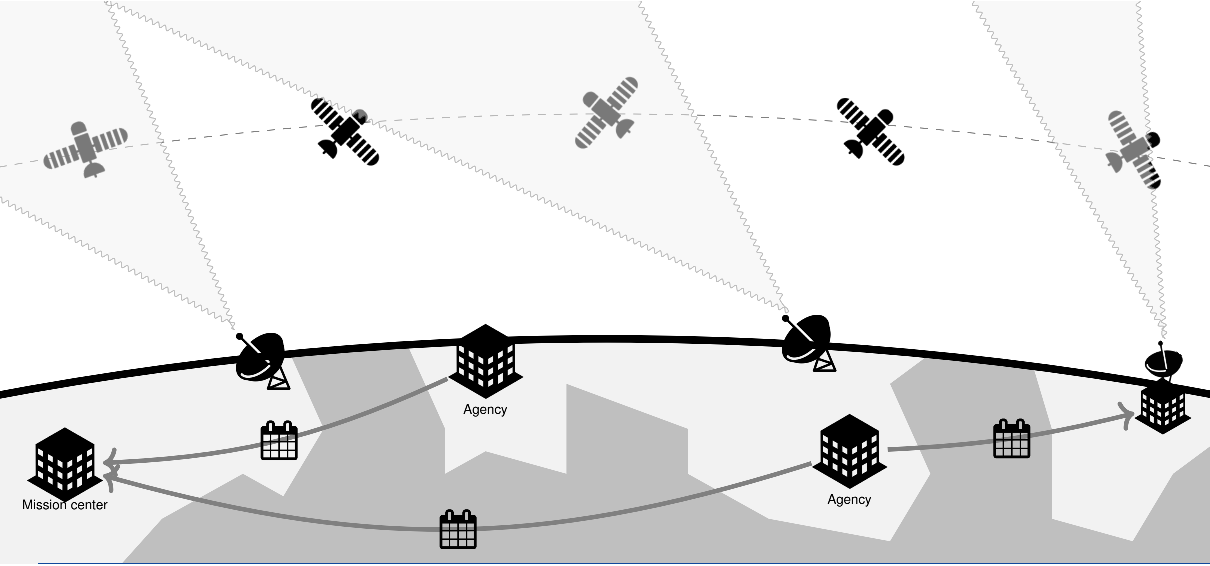
How to Schedule in a Multi-satellite and Multi-user Setting?



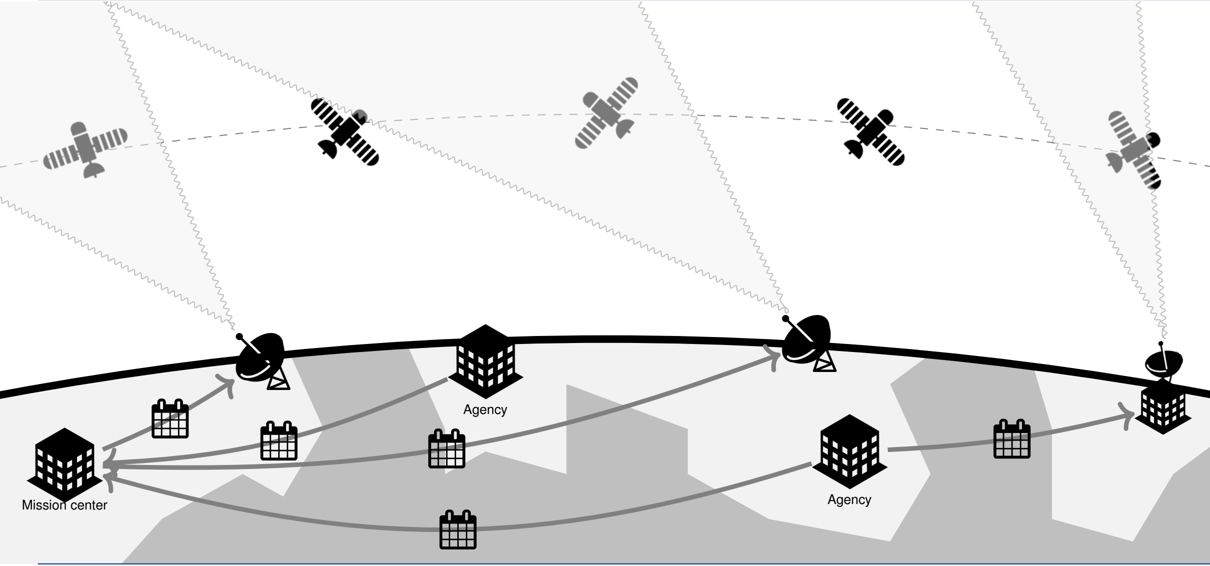
How to Schedule in a Multi-satellite and Multi-user Setting?



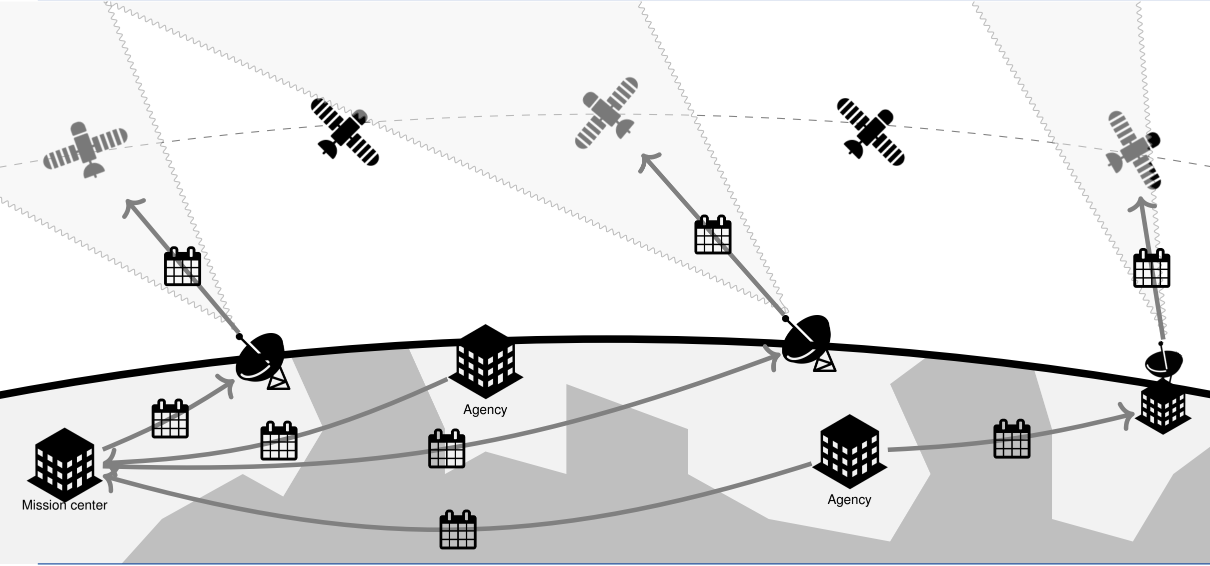
How to Schedule in a Multi-satellite and Multi-user Setting?



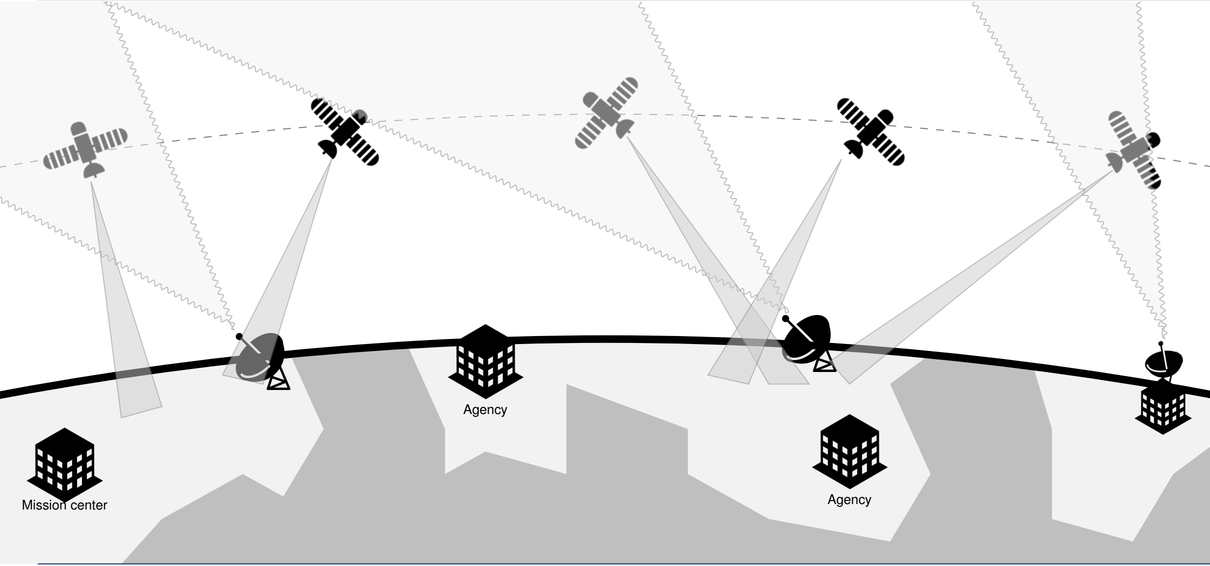
How to Schedule in a Multi-satellite and Multi-user Setting?



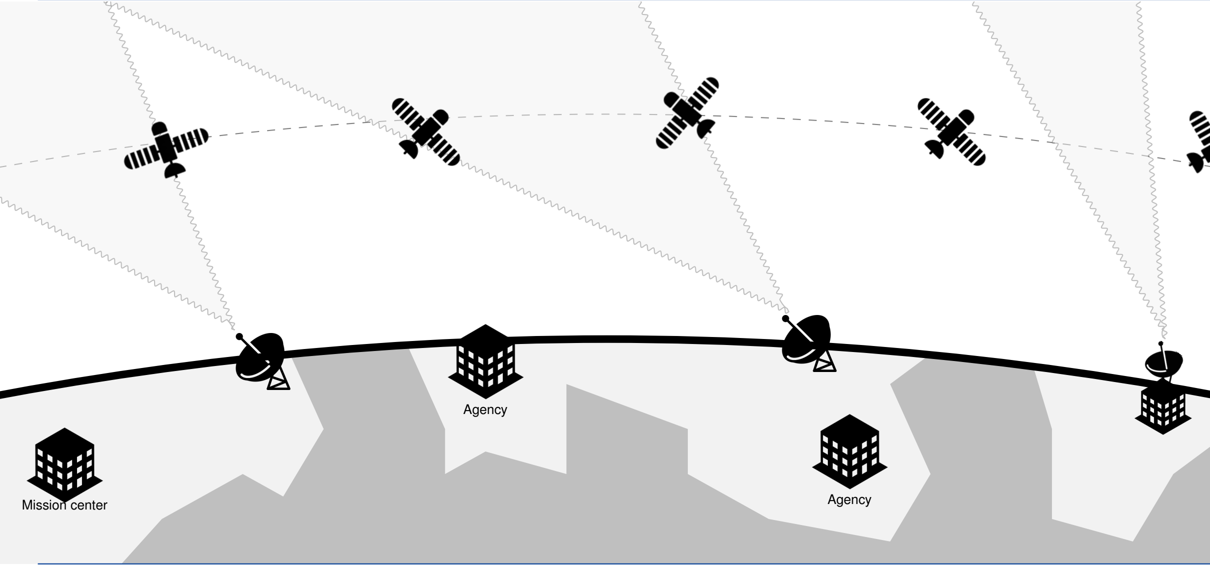
How to Schedule in a Multi-satellite and Multi-user Setting?



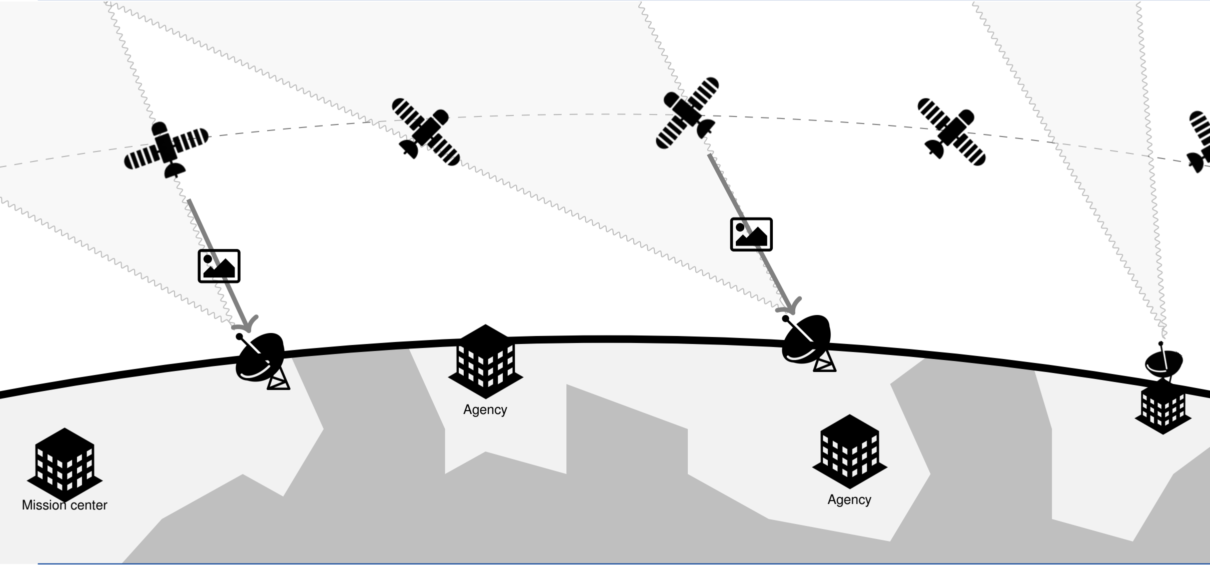
How to Schedule in a Multi-satellite and Multi-user Setting?



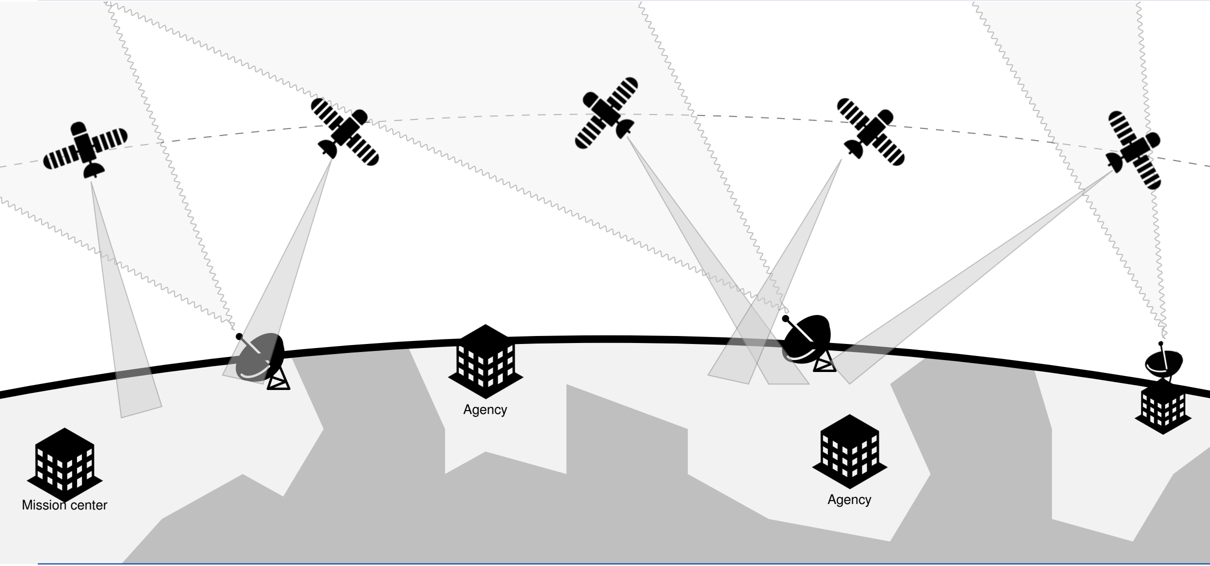
How to Schedule in a Multi-satellite and Multi-user Setting?



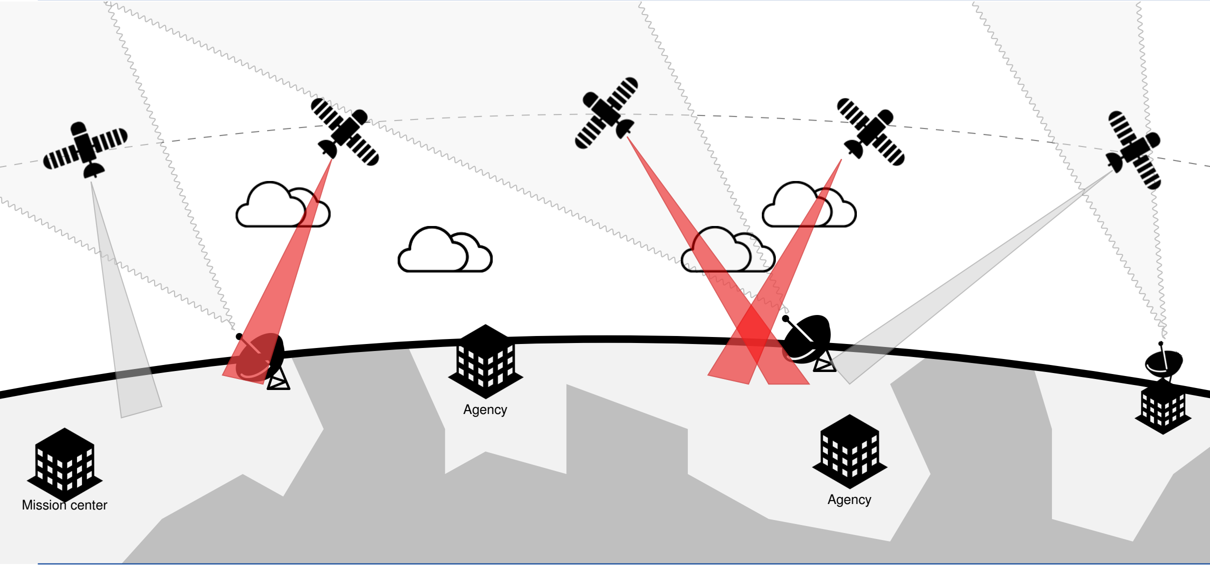
How to Schedule in a Multi-satellite and Multi-user Setting?



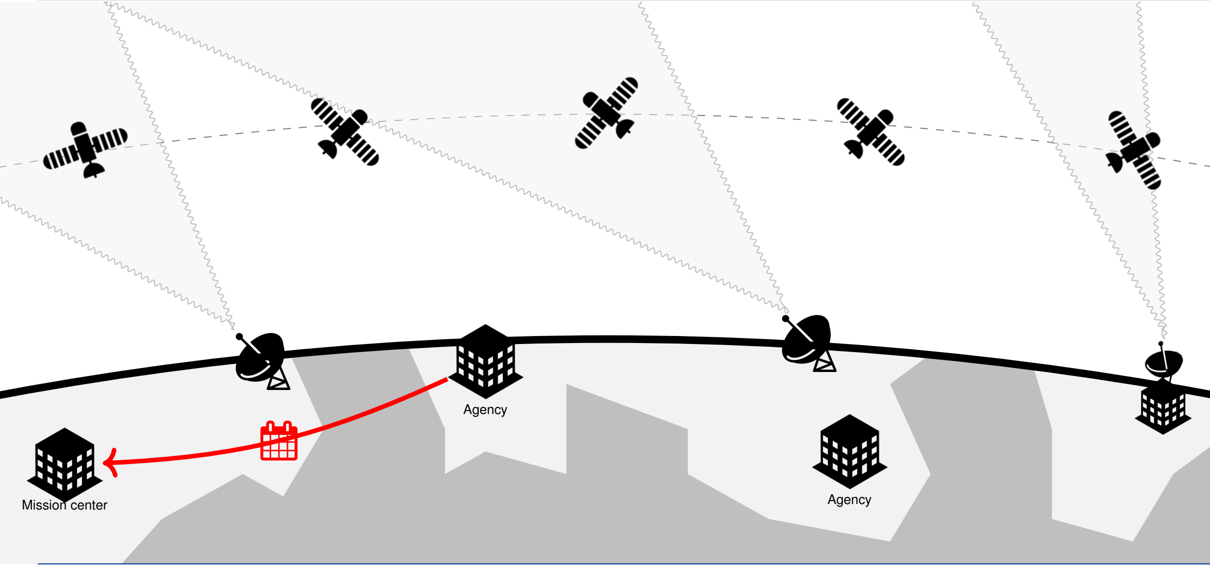
How to Adapt Activities when Facing Unpredictable Events?



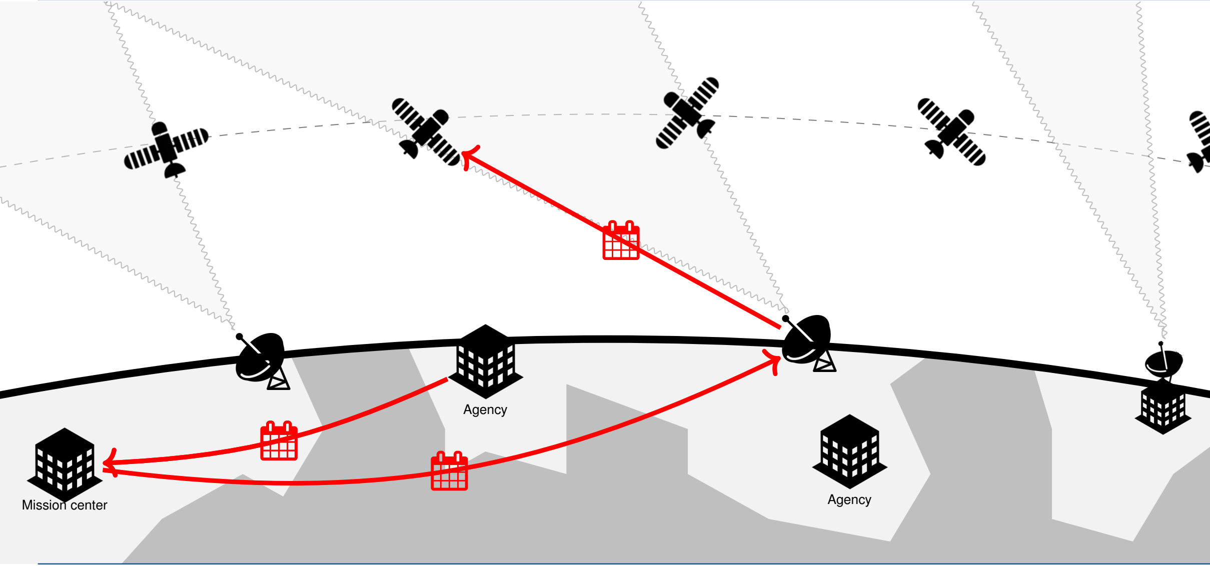
How to Adapt Activities when Facing Unpredictable Events?



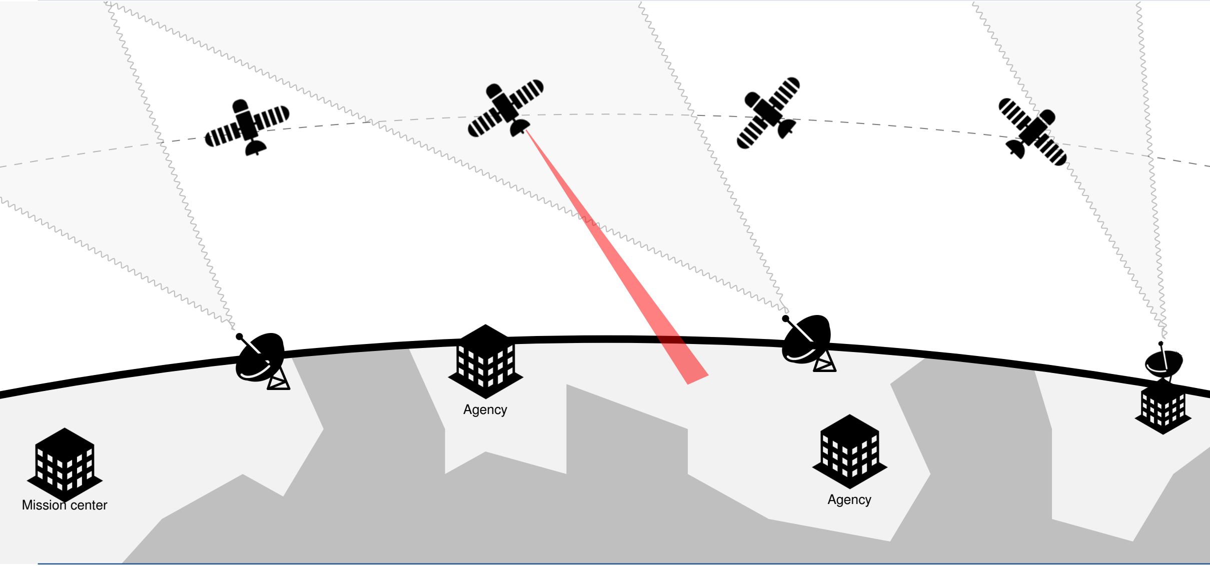
How to Adapt Activities when Facing Unpredictable Events?



How to Adapt Activities when Facing Unpredictable Events?



How to Adapt Activities when Facing Unpredictable Events?



Outline

Sharing Space Assets

Example: Earth observation satellite constellations

- **Problem :** exploitation of the same constellation/mission by several stakeholders

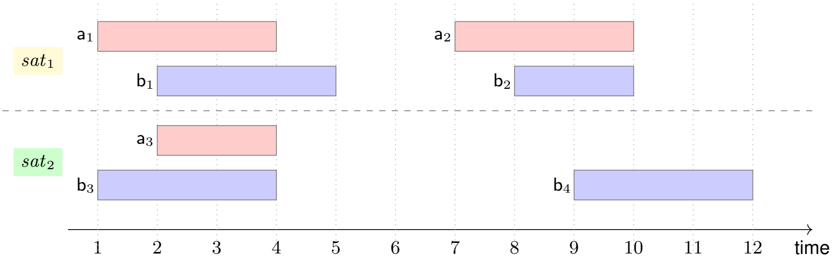
  Offline reservation systematic orbit slots — Exclusivity periods → **Online planning** Image acquisition

- **Current allocation scheme:** first come, first served
- **Objective**

  Preferences (POI, dates tolerance, ...) → **Candidate slots** (Objects to allocate) → **Utilitarian** or **fair allocation**

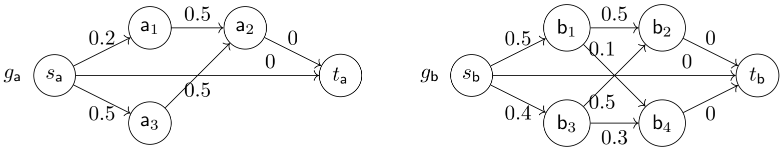
Orbit Slot Allocation
An example

- 2 agents (a in red, b in blue) requesting acquisitions:
  - of points of interest (POI) around the same region
  - around 2 time points (3 and 9) every day
- 1 satellite giving access to 2 orbit slots for each time point $(a_1, \ldots, a_3, b_1, \ldots, b_4)$



Orbit Slot Allocation Problem

Graph representation



Paths for graph g_a:

$\pi_{a,0} = [s_a, t_a]$

$\pi_{a,1} = [s_a, a_1, a_2, t_a]$

$\pi_{a,2} = [s_a, a_3, a_2, t_a]$

Paths for graph g_b:

$\pi_{b,0} = [s_b, t_b]$

$\pi_{b,1} = [s_b, b_1, b_2, t_b]$

$\pi_{b,2} = [s_b, b_1, b_4, t_b]$

$\pi_{b,3} = [s_b, b_3, b_2, t_b]$

$\pi_{b,4} = [s_b, b_3, b_4, t_b]$

Forbidden combinations:

$(\pi_{a,1}, \pi_{b,1})$

$(\pi_{a,1}, \pi_{b,3})$

$(\pi_{a,2}, \pi_{b,1})$

$(\pi_{a,2}, \pi_{b,3})$

$(\pi_{a,2}, \pi_{b,4})$

Problem Model
The non-compact case

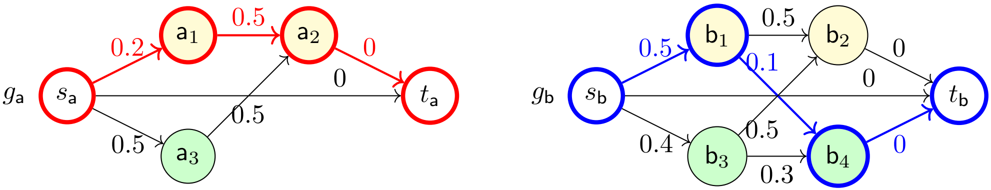
Definition

A *Directed Path Allocation Problem* (DPAP) is a tuple $\langle \mathcal{A}, \mathcal{G}, \mu, \phi \rangle$, where

- $\mathcal{A} = \{1, \ldots, n\}$ is a set of agents
- $\mathcal{G} = \{g_1, \ldots, g_m\}$ is a set of single-source single-sink edge-weighted DAGs
- $\mu : \mathcal{G} \to \mathcal{A}$ maps each graph g in $\mathcal{G}$ to its owner a in $\mathcal{A}$; we also denote by $\mathcal{G}_a = \mu^{-1}(a)$ the set of graphs owned by agent a
- $\phi : \Pi_{g_1} \times \ldots \times \Pi_{g_m} \to \{0, 1\}$ is a *path compatibility function* that indicates whether a combination of paths $(p_1, \ldots, p_m)$ (one path per graph) is feasible (value 1) or not (value 0)

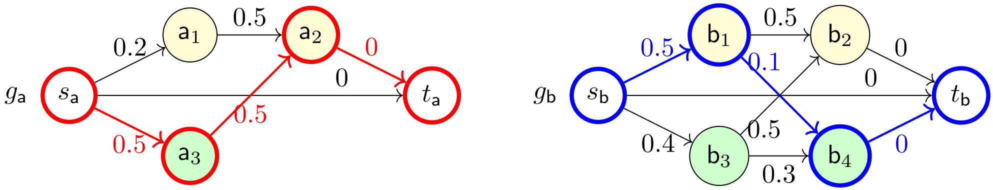
DPAP Solutions

Selecting non conflicting path in each graph



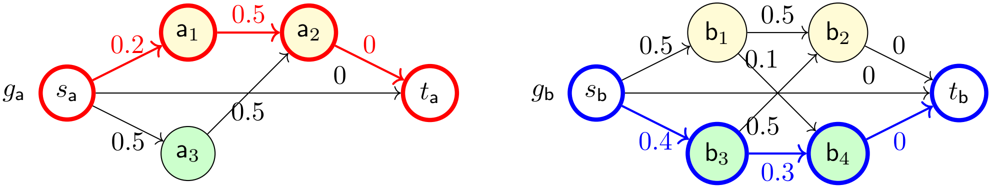
DPAP Solutions

Selecting non conflicting path in each graph, **maximizing global utility**



DPAP Solutions

Selecting non conflicting path in each graph, maximizing fairness

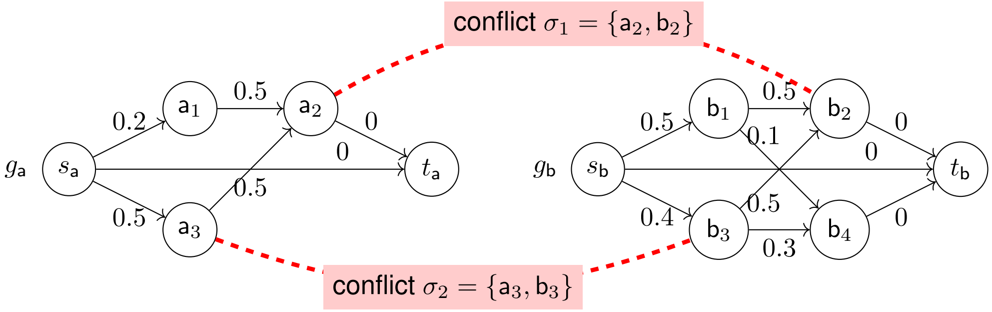


DPAP Conflict Formulations

More compact ways to represent conflicts

V-DPAP: Vertex-constrained Directed Path Allocation Problems

- ϕ is defined by a set of conflicts $\mathcal{C}$ between vertices of the graph
- each conflict $\sigma \in \mathcal{C}$ is a non-empty set of vertices V_σ that cannot be all selected by an allocation

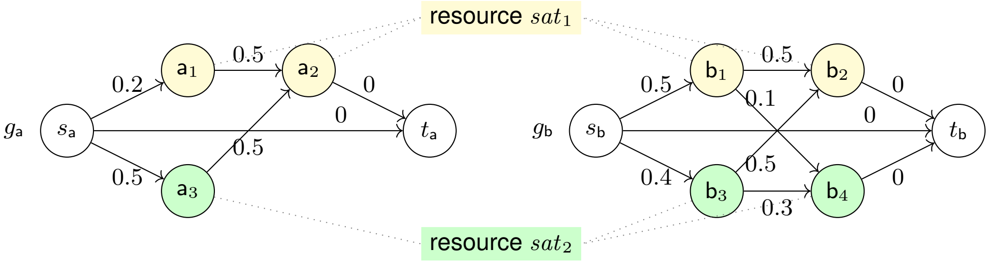


DPAP Conflict Formulations (cont.)

More compact ways to represent conflicts

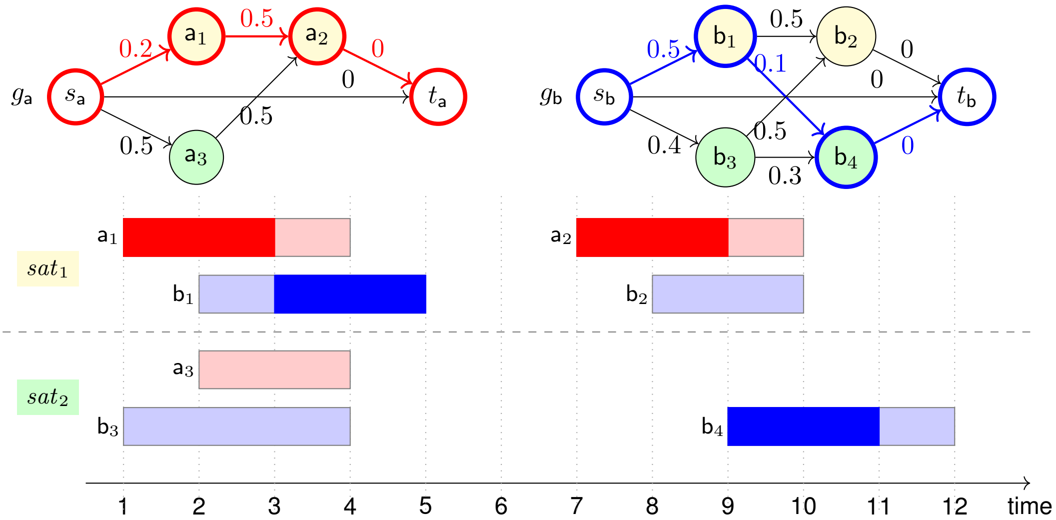
R-DPAP: Resource-constrained Directed Path Allocation Problems

- ϕ considers a set of disjunctive resources $\mathcal{R} = \{r_1, \ldots, r_p\}$
- each vertex in the graph has start date, an end date, a duration, and a required resource
- there is a conflict if at least two time windows overlap on the same resource when scheduling without any interruption (non preemptive consumption)



DPAP Conflict Formulations (cont.)

More compact ways to represent conflicts



Properties

- V-DPAP is *NP-complete* (via reduction of 3-SAT)
- R-DPAP is *NP-complete* (via reduction of 1-machine scheduling problem)
- There exists an equivalent V-DPAP to any R-DPAP
    - by generating a set of item selection conflicts that is equivalent to the set of selections forbidden by the scheduling problem

Properties

- V-DPAP is *NP-complete* (via reduction of 3-SAT)
- R-DPAP is *NP-complete* (via reduction of 1-machine scheduling problem)
- There exists an equivalent V-DPAP to any R-DPAP
  - by generating a set of item selection conflicts that is equivalent to the set of selections forbidden by the scheduling problem

→ We focus on the definition of algorithms for solving V-DPAP (because limited number of requests)

How to solve V-DPAP?

Sorry, no detail here... see [Maqrot et al., 2022; Roussel et al., 2023b]

1. Optimal utilitarian allocation (util)	MILP-based
2. Optimal leximin allocation (lex)	MILP-based iterated w/ revision
3. Approximate leximin allocation (a-lex)	MILP-based iterated wo/ revision
4. Greddy allocation (greedy)	adhoc
5. *round-robin* path allocation (p-rr)	adhoc
6. *round-robin* node allocation (n-rr)	adhoc

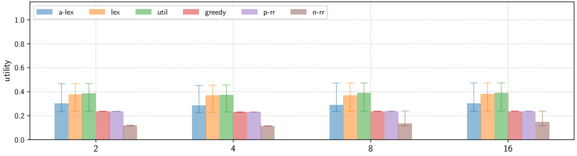
Experimental Evaluation

Generation Parameters		Values
Constellation	Altitude	500 km
	Number of orbital planes n_p	2, 4, 8, 16
	Number of satellites/plane	2
	Inclination	40°
Scheduling horizon	Start	01-01-2020
	Duration	180 days
Problems	Number of users	4
	Type	V-DPAP, R-DPAP
Requests	Number of requests/user	2
	Requested Observation Times	3 RTs/request
	Maximum random time shift δ_r	1 hour
	Tolerance Δ	1 hour
	Minimum slot duration $minD$	120 seconds
	Satisfaction mode	full, partial
Algorithms	Type	util, lex, a-lex, greedy, p-rr, n-rr
	CPLEX Time Limit	120 seconds

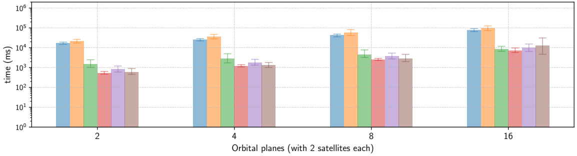
Experimental Evaluation (cont.)

Problem	Properties	n_p			
		2	4	8	16
V-DPAP	Conflicts	37715.34	74009.12	146657.94	291831.52
	Conflict size	2.0	2.0	2.0	2.0
	Slots per RT	1.94	3.81	7.54	15.01
	Slot duration (s)	618.10	616.44	616.91	616.66
R-DPAP	Conflicts	1715.38	3527.42	6981.19	13929.55
	Conflict size	3.28	3.17	3.21	3.19
	Slots per RT	1.94	3.81	7.54	15.01
	Slot duration (s)	618.10	616.44	616.91	616.66

Results for full request satisfaction mode

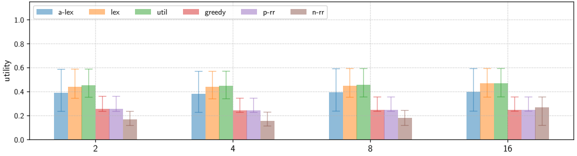


(a) Normalized utility

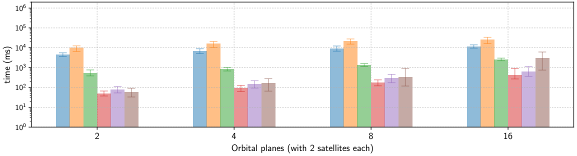


(b) Computation time

Figure: V-DPAP



(a) Normalized utility



(b) Computation time

Figure: R-DPAP

Results for full request satisfaction mode (cont.)

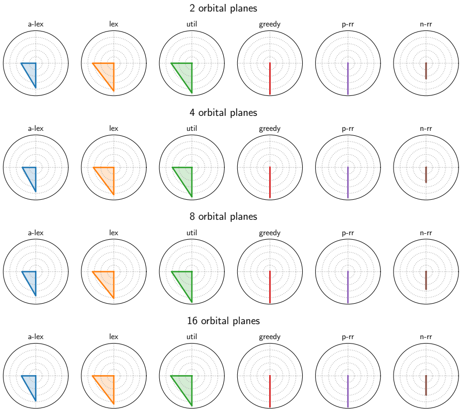


Figure: V-DPAP

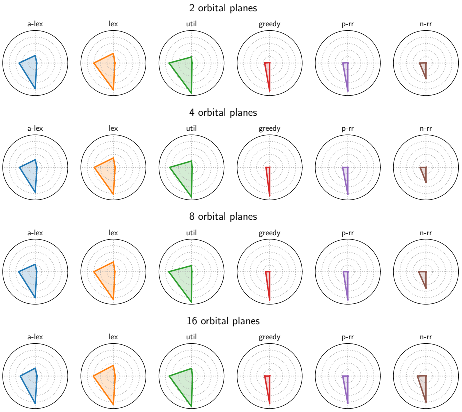
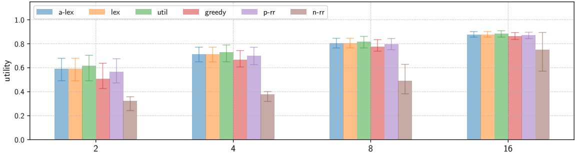
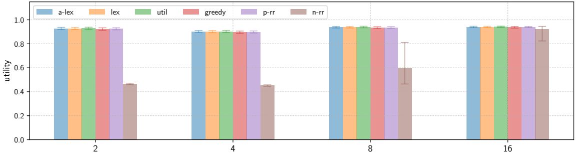


Figure: R-DPAP

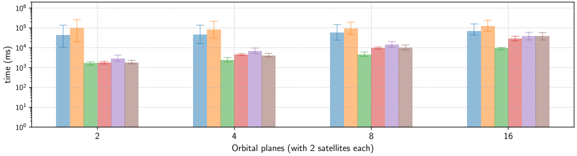
Results for flexible request satisfaction mode



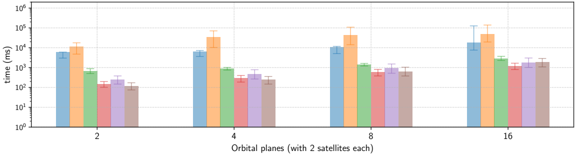
(a) Normalized utility



(a) Normalized utility



(b) Computation time



(b) Computation time

Figure: V-DPAP

Figure: R-DPAP

Results for flexible request satisfaction mode (cont.)

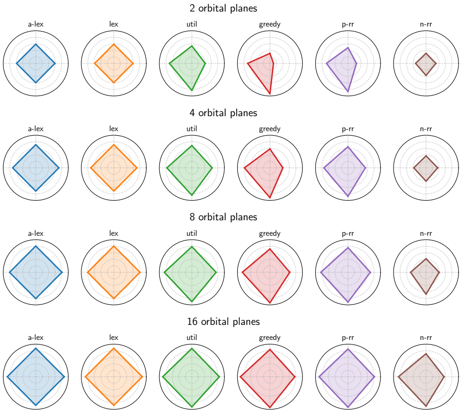


Figure: V-DPAP

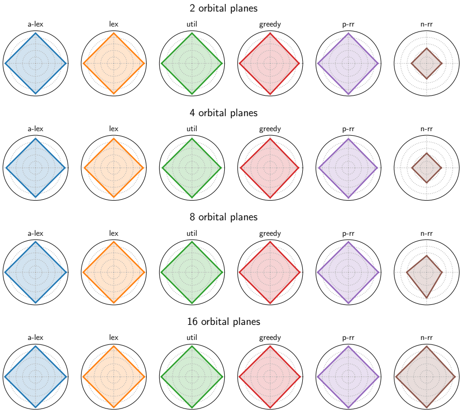


Figure: R-DPAP

Results for flexible request satisfaction mode (cont.)

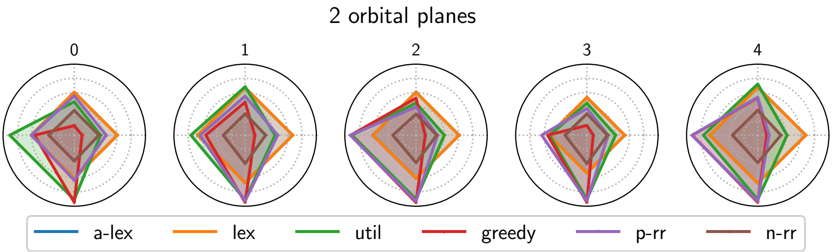
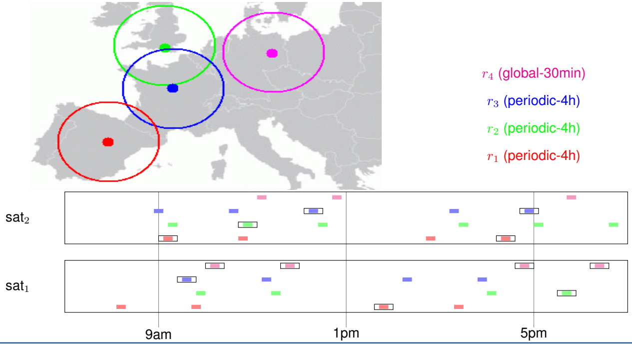


Figure: Utility profiles (in leximin order) for the first 5 instances for a constellation with 2 orbital plans (4 satellites) and each algorithm (south: best utility over all agents; west: second best utility; north: third best utility; east: worst utility), for flexible requests encoded as V-DPAP.

Where to find detailed info?

- Path allocation [MAQROT et al., 2022]
- DPAP and related methods [ROUSSEL et al., 2023b]
- More complex requests and CP-based methods [MAQROT et al., 2022]
- Some data [ROUSSEL et al., 2023a]

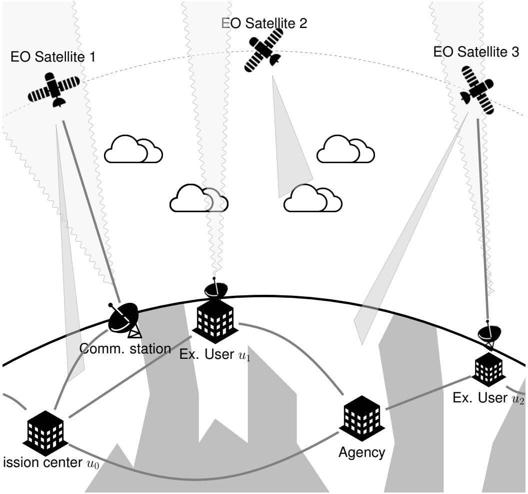


Outline

Inter-Exclusive Coordinated Scheduling

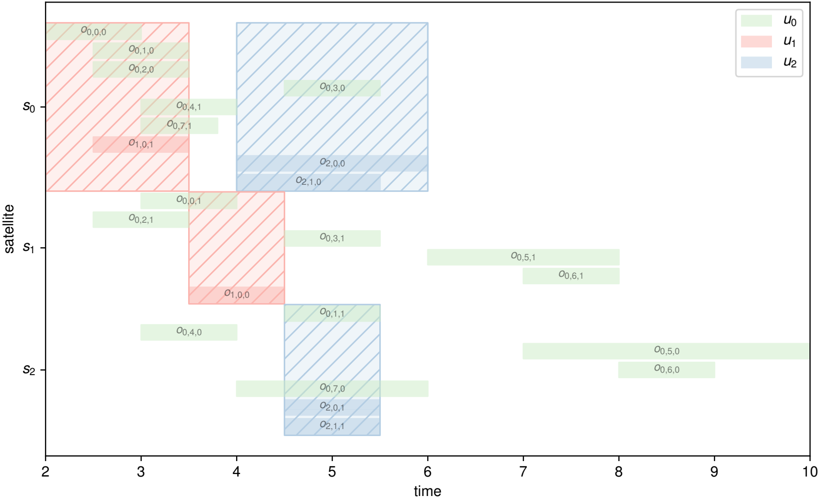
- We focus here on collective observation scheduling on a constellation where some users have exclusive access to some orbit portions

⇒ Answer to strong user expectations to benefit both from a shared system (to reduce costs) and a proprietary system (total control and confidentiality)



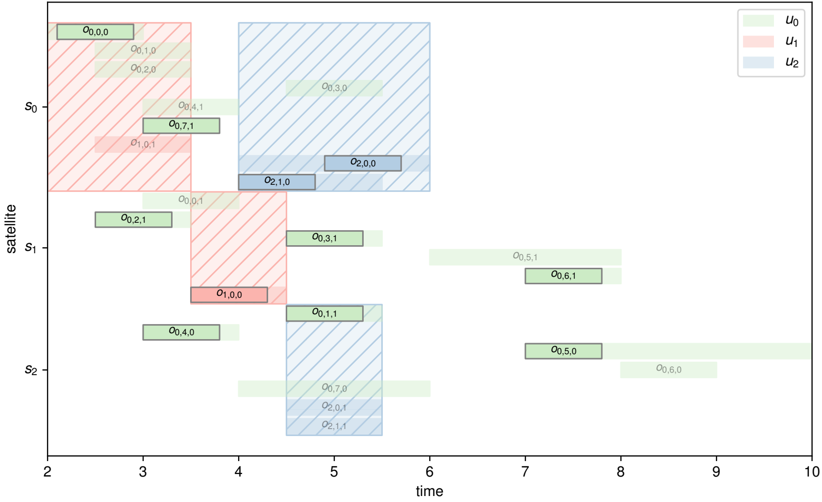
Scheduling Observations on an EOS Constellation

Illustrative Example



Scheduling Observations on an EOS Constellation

Illustrative Example



The Problems Behind

- How to **coordinate** exclusive user plans, **without disclosing private plans**, whilst meeting system constraints (memory, energy, etc.)
- How to couple private and non-private observations as to **maximize the system cost-efficiency**?

EOSCSP Model [Picard, 2022]

Earth Observation Satellite Constellation Scheduling with Exclusives Problem is a tuple

$$P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$$

- $\mathcal{S} = \{s = \langle t_s^{\text{start}}, t_s^{\text{end}}, \kappa_s, \tau_s \rangle\}$ is a set of satellites
- $\mathcal{U} = \{u = \langle e_u, p_u \rangle\}$ is a set of users
- $\mathcal{R} = \{r = \langle t_r^{\text{start}}, t_r^{\text{end}}, \Delta_r, \rho_r, p_r, u_r, \theta_r \rangle\}$ is a set of requests
- $\mathcal{O} = \{o = \langle t_o^{\text{start}}, t_o^{\text{end}}, \Delta_o, r_o, \rho_o, s_o, u_o, p_o \rangle\}$ is a set of observation opportunities

A *solution* to an EOSCSP is a mapping $\mathcal{M} = \{(o, t) \mid o \in \mathcal{O}, t \in [t_o^{\text{start}}, t_o^{\text{end}}]\}$
s.t. the overall reward is maximized (sum of the rewards of the scheduled observations):
argmax$_{\mathcal{M}} \sum_{(o,t) \in \mathcal{M}} \rho_o$

How to Solve EOSCSPs?

How to Solve EOSCSPs?

- Centralized allocation

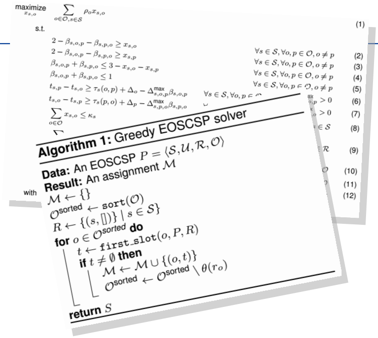
How to Solve EOSCSPs?

$$
\begin{aligned}
\underset{x_{s,o}}{\text{maximize}} \quad & \sum_{o \in \mathcal{O}, s \in \mathcal{S}} \rho_o x_{s,o} && && (1)\\
\text{s.t.} \quad & && && \\
& 2 - \beta_{s,o,p} - \beta_{s,p,o} \geq x_{s,o} && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p && (2)\\
& 2 - \beta_{s,o,p} - \beta_{s,p,o} \geq x_{s,p} && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p && (3)\\
& \beta_{s,o,p} + \beta_{s,p,o} \leq 3 - x_{s,o} - x_{s,p} && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p && (4)\\
& \beta_{s,o,p} + \beta_{s,p,o} \leq 1 && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p && (5)\\
& t_{s,p} - t_{s,o} \geq \tau_s(o,p) + \Delta_o - \Delta^{\text{max}}_{s,o,p} \beta_{s,o,p} && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p, \text{s.t.} \Delta^{\text{max}}_{s,o,p} > 0 && (6)\\
& t_{s,o} - t_{s,p} \geq \tau_s(p,o) + \Delta_p - \Delta^{\text{max}}_{s,p,o} \beta_{s,p,o} && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p, \text{s.t.} \Delta^{\text{max}}_{s,p,o} > 0 && (7)\\
& \sum_{o \in \mathcal{O}} x_{s,o} \leq \kappa_s && \forall s \in \mathcal{S} && (8)\\
& \sum_{o \in \theta(r)} x_{s,o} \leq 1 && \forall r \in \mathcal{R} && (9)\\
& x_{s,o} \in \{0,1\} && \forall s \in \mathcal{S}, \forall o \in \mathcal{O} && (10)\\
& t_{s,o} \in [t^{\text{start}}_o, t^{\text{end}}_o] \subset \mathbb{R} && \forall s \in \mathcal{S}, \forall o \in \mathcal{O} && (11)\\
& \beta_{s,o,p} \in \{0,1\} && \forall s \in \mathcal{S}, \forall o,p \in \mathcal{O}, o \neq p && (12)\\
\text{with} \quad & \Delta^{\text{max}}_{s,o,p} = t^{\text{end}}_o - t^{\text{start}}_p + \Delta_o + \tau^s(o,p) && &&
\end{aligned}
$$

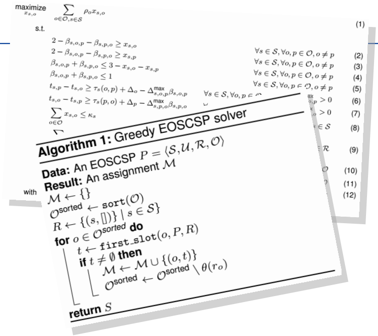
- Centralized allocation
    - Exact solving (e.g. MILP), but won’t scale-up

How to Solve EOSCSPs?

- Centralized allocation
  - Exact solving (e.g. MILP), but won't scale-up
  - Heuristic solving (e.g. greedy)

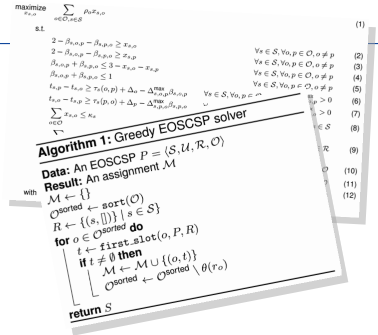


How to Solve EOSCSPs?



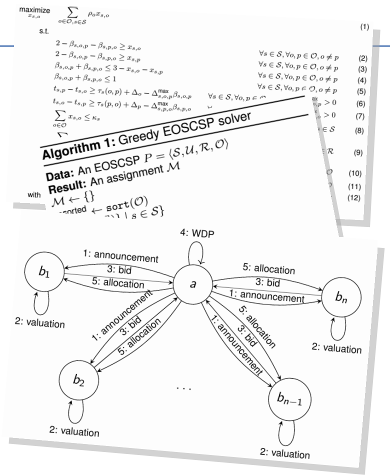
- Centralized allocation
  - Exact solving (e.g. MILP), but won’t scale-up
  - Heuristic solving (e.g. greedy)
  - ✗ private plan disclosure

How to Solve EOSCSPs?



- Centralized allocation
  - Exact solving (e.g. MILP), but won't scale-up
  - Heuristic solving (e.g. greedy)
  - ✗ private plan disclosure
- Distributed allocation

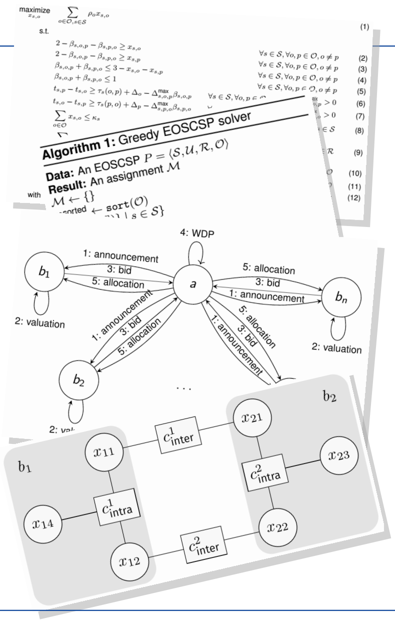
How to Solve EOSCSPs?



- Centralized allocation
  - Exact solving (e.g. MILP), but won't scale-up
  - Heuristic solving (e.g. greedy)
  - ✗ private plan disclosure
- Distributed allocation
  - Auctions (e.g. PSI, SSI, CBBA)

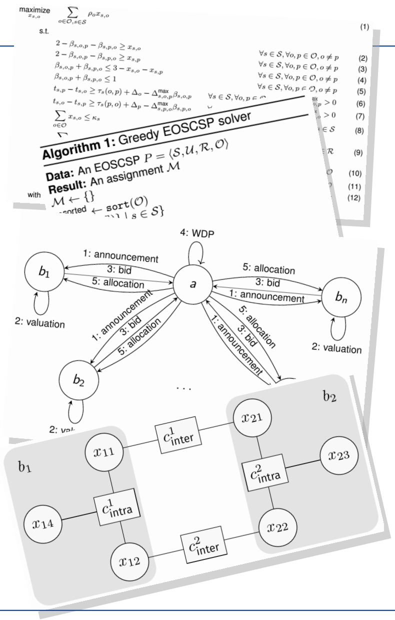
How to Solve EOSCSPs?

- Centralized allocation
  - Exact solving (e.g. MILP), but won't scale-up
  - Heuristic solving (e.g. greedy)
  - ✗ private plan disclosure
- Distributed allocation
  - Auctions (e.g. PSI, SSI, CBBA)
  - Distributed optimization (e.g. DCOPs)

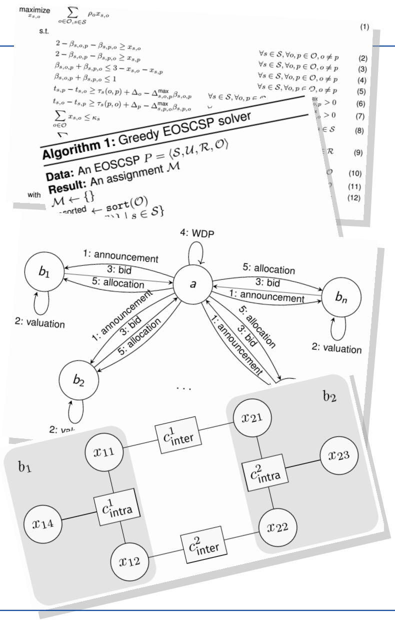


How to Solve EOSCSPs?

- Centralized allocation
  - Exact solving (e.g. MILP), but won't scale-up
  - Heuristic solving (e.g. greedy)
  - ✗ private plan disclosure
- Distributed allocation
  - Auctions (e.g. PSI, SSI, CBBA)
  - Distributed optimization (e.g. DCOPs)
  - ✓ plans remain private



How to Solve EOSCSPs?



- Centralized allocation
  - Exact solving (e.g. MILP), but won't scale-up
  - Heuristic solving (e.g. greedy)
  - ✗ private plan disclosure
- Distributed allocation
  - Auctions (e.g. PSI, SSI, CBBA)
  - Distributed optimization (e.g. DCOPs)
  - ✓ plans remain private
  - ⚠ requires some coordination/communication

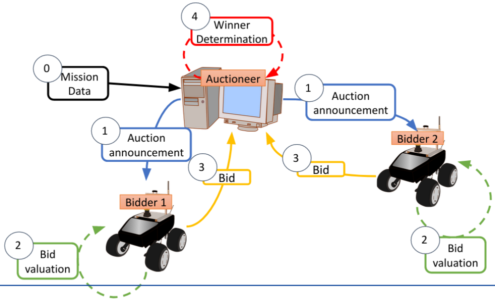
Auction-based Coordination for EOSCSP

Focus on Resource/Task Allocation

Many application fields, as Collective Robotics, make use of market-based approach to allocate tasks/resources to robots

- A set of **resources** (robots, satellites, etc.), $R = \{r_1, \ldots, r_{|R|}\}$
- A set of **tasks**, $T = \{t_1, \ldots, t_{|T|}\}$, each having a time-related and operation constraints
- Find an allocation of tasks to resources, wrt. some consistency constraints

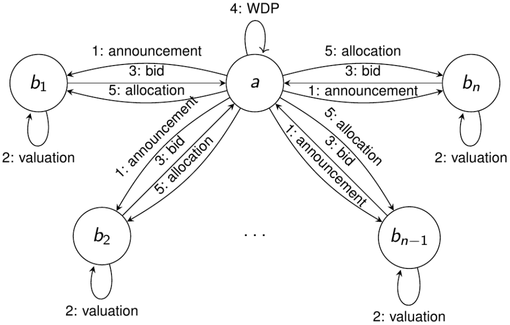
≈ **multi-item allocation**: each resource is allocated several tasks (bundle)



Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

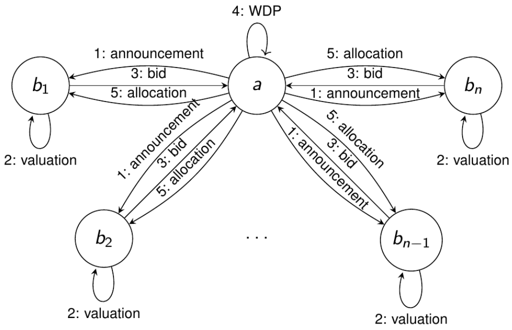


Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

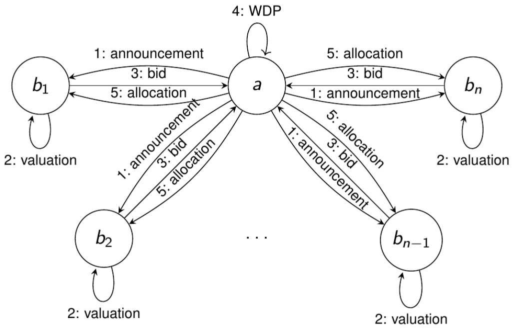
- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]



Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

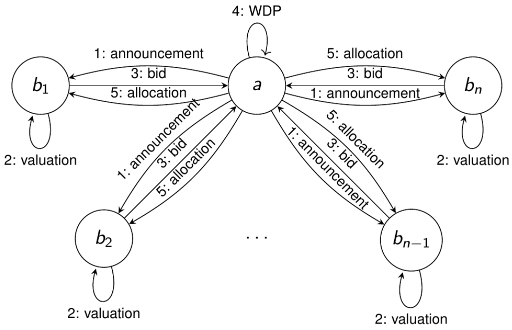


- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]
- **Parallel Single Item Auctions** (PSI) [KOENIG et al., 2006]

Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

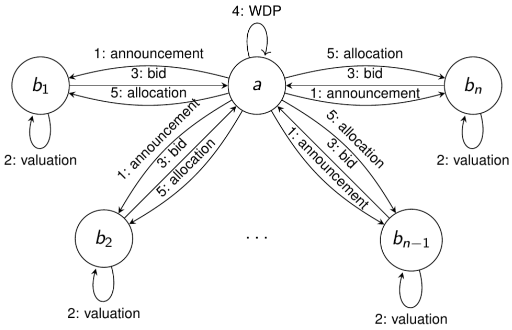


- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]
- **Parallel Single Item Auctions** (PSI) [KOENIG et al., 2006]
  - Each agent bids on the *whole set of tasks* in parallel

Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

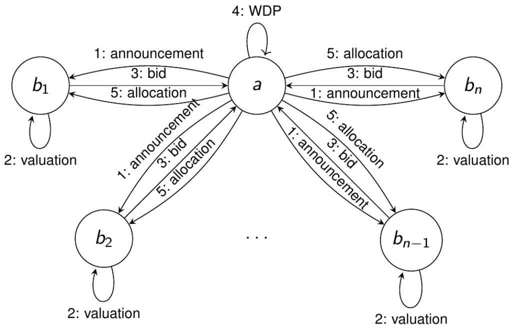


- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]
- **Parallel Single Item Auctions** (PSI) [KOENIG et al., 2006]
  - Each agent bids on the *whole set of tasks* in parallel
- **Sequential Single Item Auctions** (SSI) [LAGOUDAKIS et al., 2005]

Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

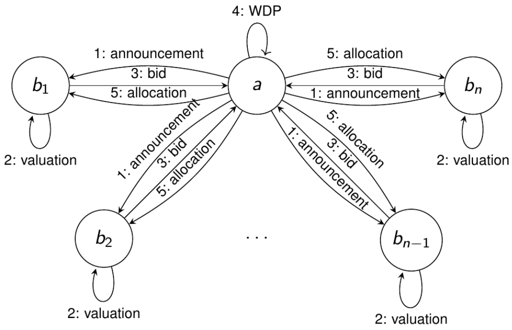


- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]
- **Parallel Single Item Auctions** (PSI) [KOENIG et al., 2006]
  - Each agent bids on the *whole set of tasks* in parallel
- **Sequential Single Item Auctions** (SSI) [LAGOUDAKIS et al., 2005]
  - Each agent *sequentially* bids on a *single task* wrt to the already allocated tasks

Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

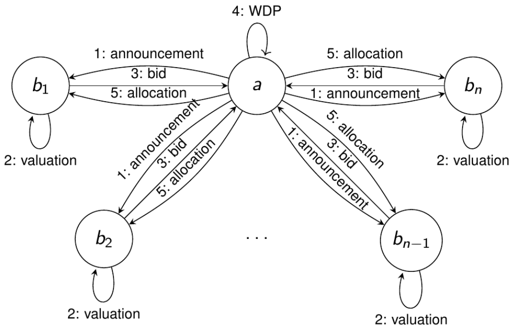


- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]
- **Parallel Single Item Auctions** (PSI) [KOENIG et al., 2006]
  - Each agent bids on the *whole set of tasks* in parallel
- **Sequential Single Item Auctions** (SSI) [LAGOUDAKIS et al., 2005]
  - Each agent *sequentially* bids on a *single task* wrt to the already allocated tasks
- **Consensus-based Bundle Auction** (CBBA) [CHOI et al., 2009]

Auction-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

Auction-based approches are relevant for satellite task allocation [PHILLIPS and PARRA, 2021]

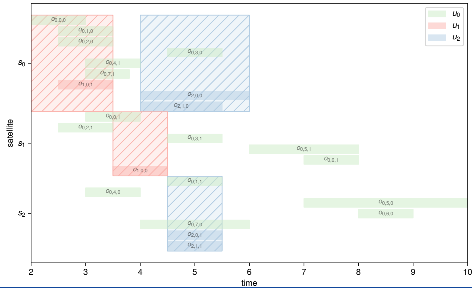


- **Combinatorial Auctions** (CA) [CRAMTON et al., 2010]
- **Parallel Single Item Auctions** (PSI) [KOENIG et al., 2006]
  - Each agent bids on the *whole set of tasks* in parallel
- **Sequential Single Item Auctions** (SSI) [LAGOUDAKIS et al., 2005]
  - Each agent *sequentially* bids on a *single task* wrt to the already allocated tasks
- **Consensus-based Bundle Auction** (CBBA) [CHOI et al., 2009]
  - Each agent bids on some *bundle of tasks* and *converge to a consensus* with other agents

Applying Auction-based Allocation to EOSCSP

General Scheme

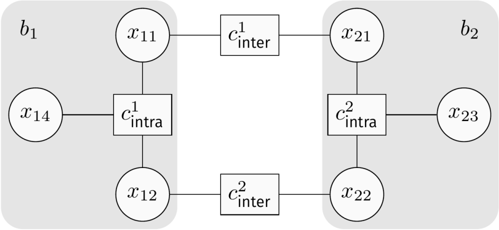
1. Identify non exclusive requests possibly fulfilled in exclusive portions
2. Send identified requests to exclusive users
3. Solve the allocation problem using PSI, SSI or CBBA
    - Bids are computed as the **best marginal costs** of integrating requests in their current plans (which amounts to solve scheduling problems...)
4. Allocate as many remaining requests outside exclusive windows



DCOP-based Coordination for EOSCSP

Allocating non exclusive observations to best exclusive portions

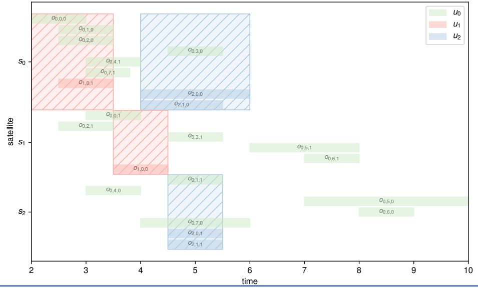
- Consider the **collective decision** for allocating non exclusive tasks to exclusive windows
- Collective decision to coordinate exclusive users' decisions modeled as a **distributed constraint optimization problem** (DCOP)
- As for auctions, exclusive users aim to **minimizing the marginal cost** of integrating non exclusive tasks in their schedule, while meeting some operational constraints



DCOP-based Coordination for EOSCSP

General Scheme

1. Identify non exclusive requests possibly fulfilled in exclusive windows
2. Send each identified request r to exclusives users, one by one
3. Solve the problem of r using a DCOP solution method (e.g. DPOP [Petcu2005])
   - Costs are computed as the **best marginal cost** of integrating requests in their current plan (which amounts to solve a scheduling problem...)
4. Allocate as many remaining requests outside exclusive windows



DCOP-based Coordination for EOSCSP

DCOP Model

A DCOP $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu \rangle$ is defined for a given request r, and a current scheduling

DCOP-based Coordination for EOSCSP

DCOP Model

A DCOP $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu \rangle$ is defined for a given request r, and a current scheduling

- The agents are the exclusive users which can potentially schedule r:

$$\mathcal{A} = \{u \in \mathcal{U}^{\text{ex}} | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \exists o \in \theta_r \text{ s.t. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\} \quad (1)$$

DCOP-based Coordination for EOSCSP

DCOP Model

A DCOP $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu \rangle$ is defined for a given request r, and a current scheduling

- The agents are the exclusive users which can potentially schedule r:

$$\mathcal{A} = \{u \in \mathcal{U}^{\text{ex}} | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \exists o \in \theta_r \text{ s.t. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\} \quad (1)$$

- Each agent u owns binary decision variables, one for each observation $o \in \mathcal{O}[u]^r$ and exclusive e in its exclusives e_u, stating whether it schedules o in e or not:

$$\mathcal{X} = \{x_{e,o} | e \in \textstyle\bigcup_{u \in \mathcal{A}} e_u, o \in \mathcal{O}[u]^r\} \quad (2)$$

$$\mathcal{D} = \{\mathcal{D}_{x_{e,o}} = \{0, 1\} | x_{e,o} \in \mathcal{X}\} \quad (3)$$

DCOP-based Coordination for EOSCSP

DCOP Model

A DCOP $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu \rangle$ is defined for a given request r, and a current scheduling

- The agents are the exclusive users which can potentially schedule r:

$$\mathcal{A} = \{u \in \mathcal{U}^{\text{ex}} | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \exists o \in \theta_r \text{ s.t. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\} \quad (1)$$

- Each agent u owns binary decision variables, one for each observation $o \in \mathcal{O}[u]^r$ and exclusive e in its exclusives e_u, stating whether it schedules o in e or not:

$$\mathcal{X} = \{x_{e,o} | e \in \textstyle\bigcup_{u \in \mathcal{A}} e_u, o \in \mathcal{O}[u]^r\} \quad (2)$$

$$\mathcal{D} = \{\mathcal{D}_{x_{e,o}} = \{0, 1\} | x_{e,o} \in \mathcal{X}\} \quad (3)$$

with $\mathcal{O}[u]^r = \{o \in \theta r | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \text{ s.t. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\}$ are observations related to request r that can be scheduled on u's exclusives

A DCOP $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu \rangle$ is defined for a given request r, and a current scheduling

- The agents are the exclusive users which can potentially schedule r:

$$\mathcal{A} = \{u \in \mathcal{U}^{\text{ex}} | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \exists o \in \theta_r \text{ s.t. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\} \quad (1)$$

- Each agent u owns binary decision variables, one for each observation $o \in \mathcal{O}[u]^r$ and exclusive e in its exclusives e_u, stating whether it schedules o in e or not:

$$\mathcal{X} = \{x_{e,o} | e \in \bigcup_{u \in \mathcal{A}} e_u, o \in \mathcal{O}[u]^r\} \quad (2)$$

$$\mathcal{D} = \{\mathcal{D}_{x_{e,o}} = \{0, 1\} | x_{e,o} \in \mathcal{X}\} \quad (3)$$

with $\mathcal{O}[u]^r = \{o \in \theta r | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \text{ s.t. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\}$ are observations related to request r that can be scheduled on u's exclusives

- μ associates each variable $x_{e,o}$ to e's owner

DCOP-based Coordination for EOSCSP (cont.)

DCOP Model

- Constraints should check that at most one observation is scheduled per request (4), that satellites are not overloaded (5), that at most one agent serves the same observation (6)

$$\sum_{e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq 1, \quad \forall u \in \mathcal{X}, \forall o \in \mathcal{O}[u]^r \tag{4}$$

$$\sum_{o \in \{o \in \mathcal{O}[u]^r | u \in \mathcal{A}, s_o = s\}, e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq \kappa_s^*, \; \forall s \in \mathcal{S} \tag{5}$$

$$\sum_{e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq 1, \quad \forall o \in \mathcal{O} \tag{6}$$

DCOP-based Coordination for EOSCSP (cont.)

DCOP Model

- Constraints should check that at most one observation is scheduled per request (4), that satellites are not overloaded (5), that at most one agent serves the same observation (6)

$$\sum_{e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq 1, \quad \forall u \in \mathcal{X}, \forall o \in \mathcal{O}[u]^r \tag{4}$$

$$\sum_{o \in \{o \in \mathcal{O}[u]^r | u \in \mathcal{A}, s_o = s\}, e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq \kappa_s^*, \; \forall s \in \mathcal{S} \tag{5}$$

$$\sum_{e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq 1, \quad \forall o \in \mathcal{O} \tag{6}$$

- The cost to integrate an observation in the current user's schedule should be assessed to guide the optimization process

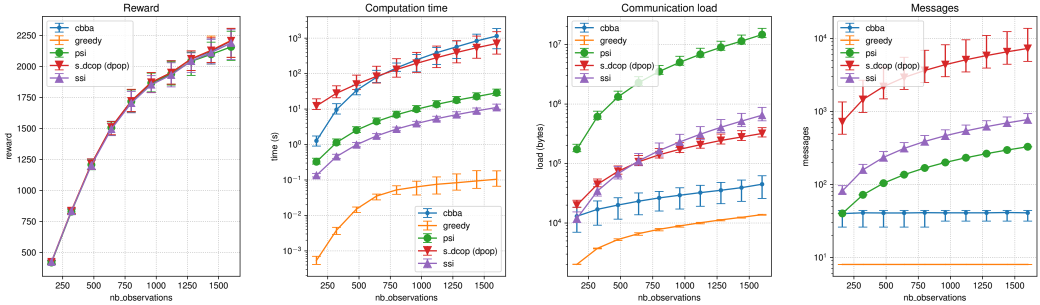
$$c(x_{e,o}) = \pi(o, \mathcal{M}_{u_o}), \quad \forall x_{e,o} \in \mathcal{X} \tag{7}$$

where π evaluates the best cost obtained when scheduling o and any combination of observations from $\mathcal{M}_{u_o}$, as to consider all possible revisions of u_o's current schedule

$$\mathcal{C} = \{(4), (5), (6), (7)\} \tag{8}$$

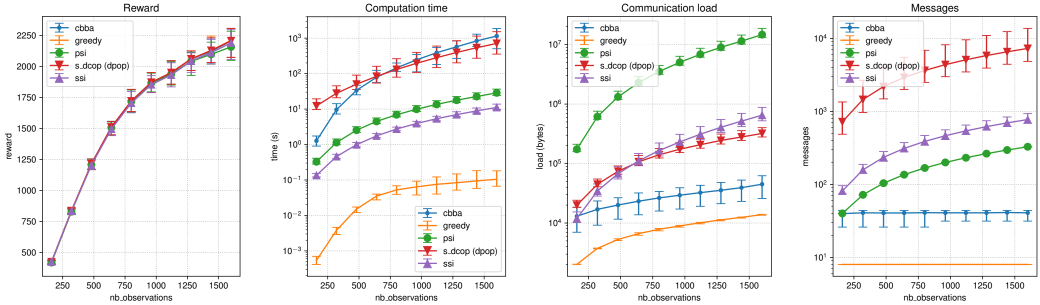
Highly conflicting randomly generated problems

5-min horizon with overlapping requests and limited capacity



Highly conflicting randomly generated problems

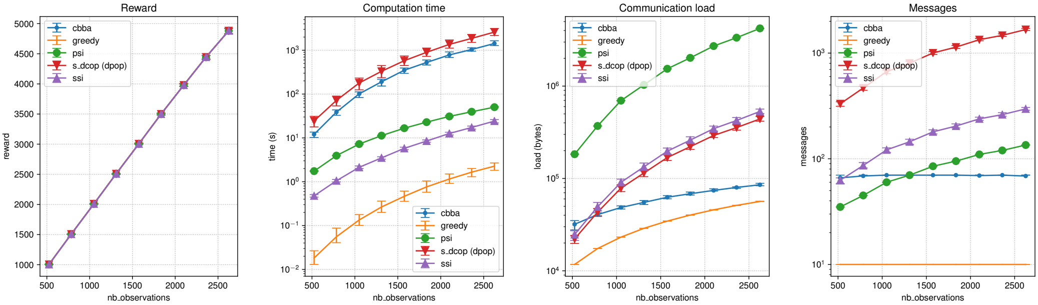
5-min horizon with overlapping requests and limited capacity



- ✗ cbba and s_dcop requires extra-computation time (≈ 1000s)
- ✓ cbba and s_dcop provide the best solutions wrt. reward
- ✓ cbba exchanges fewer messages of small size
- ✓ ssi remains the best compromise wrt. solution quality, computation time and communication load

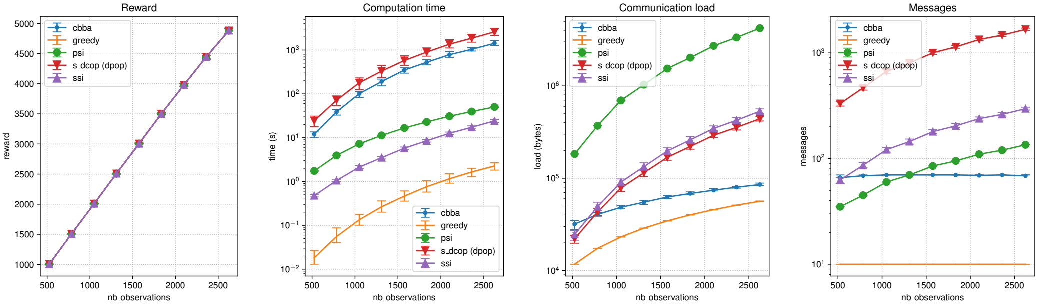
Realistic randomly generated problems

6-hour horizon with numerous requests and large capacity



Realistic randomly generated problems

6-hour horizon with numerous requests and large capacity



- ✓ cbba does require less time to compute than s_dcop
- ✓ s_dcop and cbba can perform many computation concurrently
- ⇒ There is room for computation speedup in real distributed settings

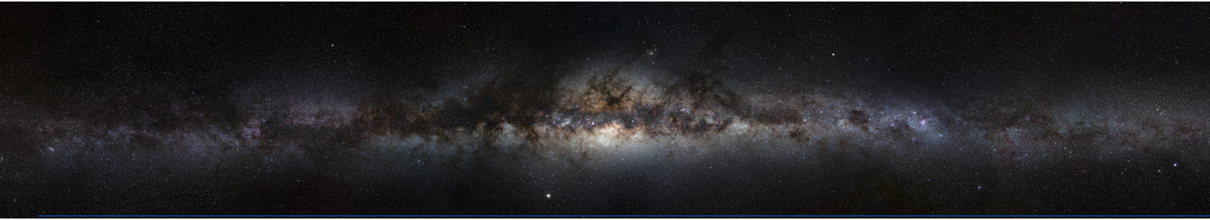
Where to find detailed info?

- Initial model definition [Picard, 2022]
- Auction-based and DCOP-based solution methods [ibid.]
- More complex requests and decentralized auctions [Picard, 2023a]
- Some data [Picard, 2023b]

Outline

Wrap-up

- Key terms for NewSpace: multi-asset, multi-user, multi-system...
- Asset sharing means **cost-efficiency**, but requires **automated coordination** and **privacy/sovereignity** preservation

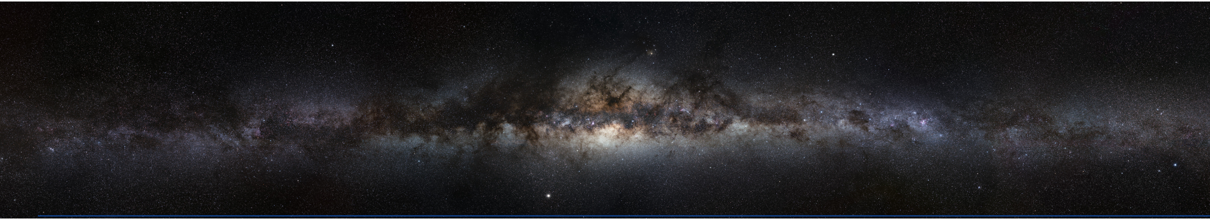


RÉPUBLIQUE FRANÇAISE
Liberté Égalité Fraternité

ONERA
THE FRENCH AEROSPACE LAB

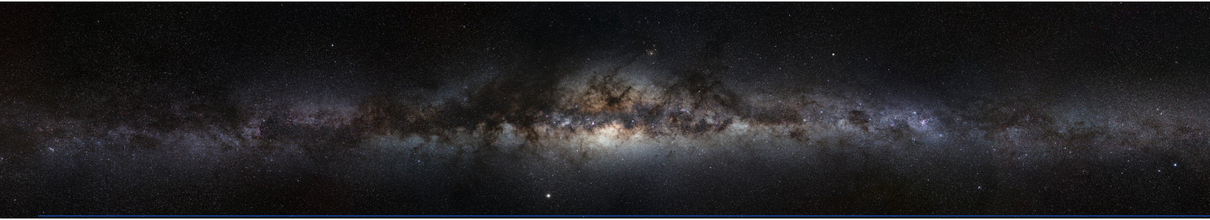
Wrap-up

- How to coordinate such composite systems?
  - Efficiency
  - Fairness
  - Explainability



Wrap-up

- How to coordinate such composite systems?
  - Efficiency
  - Fairness
  - Explainability
- Multi-agent Systems
  - Resource allocation and combinatorial auctions
  - Distributed optimization
  - Federated and multi-agent learning
  - ...



RÉPUBLIQUE FRANÇAISE
Liberté Égalité Fraternité

ONERA
THE FRENCH AEROSPACE LAB

Our Next Steps

- Even more complex requests
  - Periodic intra-/inter-day, short-/long-term
  - Large area and responsiveness
- Even more complex systems
  - Weather uncertainties
  - Constellation federations

Acknowledgements

Jean-Loup Farges, Cédric Pralet, Stéphanie Roussel, and Sara Maqrot
(at ONERA)

Clément Caron and Jonathan Guerra
(at ADS)

Acknowledgements

This work has been performed with the support of the French government in the context of the “Programme d’Invertissements d’Avenir”, namely by the BPI PSPC project “LiChIE”

Thank you for your attention!

Any question?

www.onera.fr

References

CHOI, Han-Lim, Luc BRUNET, and Jonathan P. HOW (2009). "Consensus-Based Decentralized Auctions for Robust Task Allocation". In: *IEEE Trans. Robotics* 25.4, pp. 912–926. DOI: 10.1109/TRO.2009.2022423.

CRAMTON, Peter, Yoav SHOHAM, and Richard STEINBERG, eds. (2010). *"Combinatorial Auctions"*. MIT Press.

KOENIG, Sven, Craig A. TOVEY, Michail G. LAGOUDAKIS, Evangelos MARKAKIS, David KEMPE, Pinar KESKINOCAK, Anton J. KLEYWEGT, Adam MEYERSON, and Sonal JAIN (2006). "The Power of Sequential Single-Item Auctions for Agent Coordination". In: *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*. AAAI Press, pp. 1625–1629. URL: http://www.aaai.org/Library/AAAI/2006/aaai06-266.php.

LAGOUDAKIS, Michail G., Evangelos MARKAKIS, David KEMPE, Pinar KESKINOCAK, Anton J. KLEYWEGT, Sven KOENIG, Craig A. TOVEY, Adam MEYERSON, and Sonal JAIN (2005). "Auction-Based Multi-Robot Routing". In: *Robotics: Science and Systems I, June 8-11, 2005, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA*. Ed. by Sebastian THRUN, Gaurav S. SUKHATME, and Stefan SCHAAL. The MIT Press, pp. 343–350. DOI: 10.15607/RSS.2005.I.045. URL: http://www.roboticsproceedings.org/rss01/p45.html.

MAQROT, S., S. ROUSSEL, G. PICARD, and C. PRALET (2022). "Bundle Allocation with Conflicting Preferences Represented as Weighted Directed Acyclic Graphs – Application to Orbit Slot Ownership". In: *Advances in Practical Applications of Agents, Multi-Agent Systems, and Complex Systems Simulation, The PAAMS Collection*. Ed. by Frank DIGNUM, Philippe MATHIEU, Juan Manuel CORCHADO, and Fernando De la PRIETA. Vol. 13616. LNAI. Springer, pp. 280–293. DOI: 10.1007/978-3-031-18192-4_23.

References (cont.)

PHILLIPS, Sean and Fernando PARRA (2021). "A Case Study on Auction-Based Task Allocation Algorithms in Multi-Satellite Systems". In: *AIAA Scitech 2021 Forum*. DOI: 10.2514/6.2021-0185. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2021-0185. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2021-0185.

PICARD, Gauthier (2022). "Auction-based and Distributed Optimization Approaches for Scheduling Observations in Satellite Constellations with Exclusive Orbit Portions". In: *International Conference on Autonomous Agents and Multiagent Systems (AAMAS-22)*. IFAAMAS, pp. 1056–1064. DOI: https://dl.acm.org/doi/10.5555/3535850.3535968.

— (2023a). "Multi-Agent Consensus-based Bundle Allocation for Multi-Mode Composite Tasks". In: *International Conference on Autonomous Agents and Multiagent Systems (AAMAS-23)*. in press. IFAAMAS.

— (2023b). *Multi-agent multi-mode composite task allocation problem (MACTA) instances*. Version 1.0. Dataset. DOI: 10.5281/zenodo.7550677. URL: https://doi.org/10.5281/zenodo.7550677.

PICARD, Gauthier, Clément CARON, Jean-Loup FARGES, Jonathan GUERRA, Cédric PRALET, and Stéphanie ROUSSEL (2021). "Autonomous Agents and Multiagent Systems Challenges in Earth Observation Satellite Constellations". In: *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*. Ed. by U. ENDRISS, A. NOWÉ, F. DIGNUM, and A. LOMUSCIO. AAMAS '21. Virtual Event, United Kingdom: International Foundation for Autonomous Agents and Multiagent Systems, 39–44. ISBN: 9781450383073. DOI: https://dl.acm.org/doi/10.5555/3463952.3463961. URL: https://dl.acm.org/doi/10.5555/3463952.3463961.

ROUSSEL, Stéphanie, Gauthier PICARD, and Cédric PRALET (2023a). *Instances of Orbit Slot Allocation Problems modeled as Directed Path Allocation Problems (DPAP)*. Version 1.0. Dataset. DOI: 10.5281/zenodo.7669379. URL: https://doi.org/10.5281/zenodo.7669379.

References (cont.)

Roussel, Stéphanie, Gauthier Picard, Cédric Pralet, and Sara Maqrot (2023b). "Conflicting Bundle Allocation with Preferences in Weighted Directed Acyclic Graphs: Application to Orbit Slot Allocation Problems". In: *Systems* 11.6. issn: 2079-8954. doi: 10.3390/systems11060297. url: https://www.mdpi.com/2079-8954/11/6/297.