RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

Planification multi-utilisateurs et multi-satellites de tâches d'observation dans des constellations avec portions d'orbites exclusives

Gauthier Picard
`gauthier.picard@onera.fr`

ONERA, DTIS-SYD, Université de Toulouse

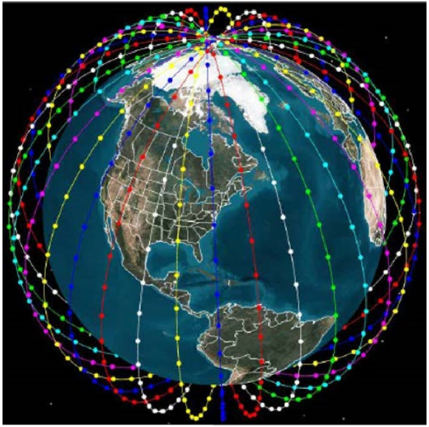
29 juin 2021

Contexte et motivations

- Forte croissance du déploiement de constellations EOS de grande taille

⇒ Observer tout point sur Terre à plus grande fréquence, e.g. Planet [SHAH et al., 2019]

- Mais, nécessite d'améliorer la coordination et la coopération entre les différentes ressources et acteurs



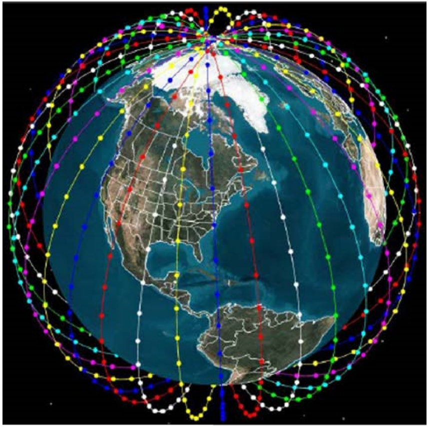
Contexte et motivations

- Forte croissance du déploiement de constellations EOS de grande taille

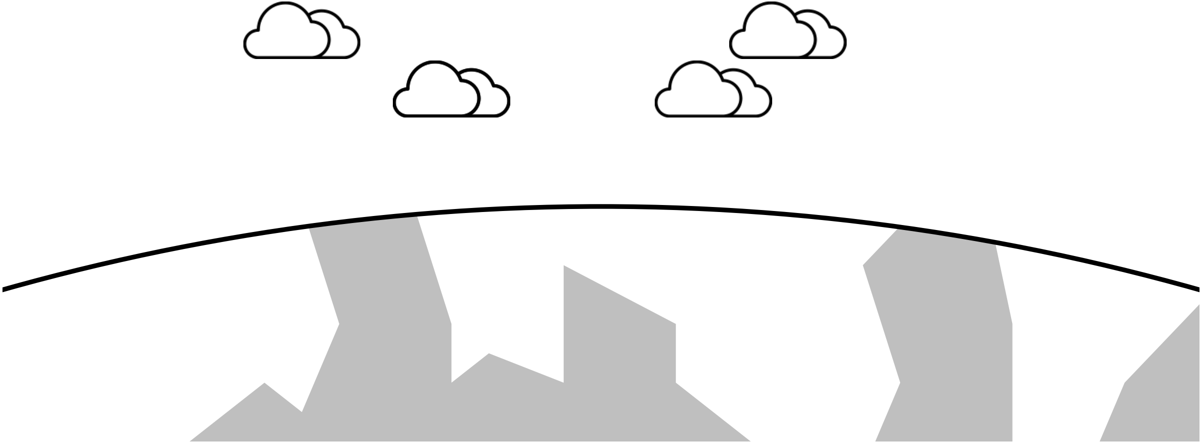
⇒ Observer tout point sur Terre à plus grande fréquence, e.g. Planet [SHAH et al., 2019]

- Mais, nécessite d'améliorer la coordination et la coopération entre les différentes ressources et acteurs
- Nous nous intéressons ici à la planification collective d'observations sur une constellation dans laquelle certains utilisateurs ont des accès exclusifs à des portions d'orbite

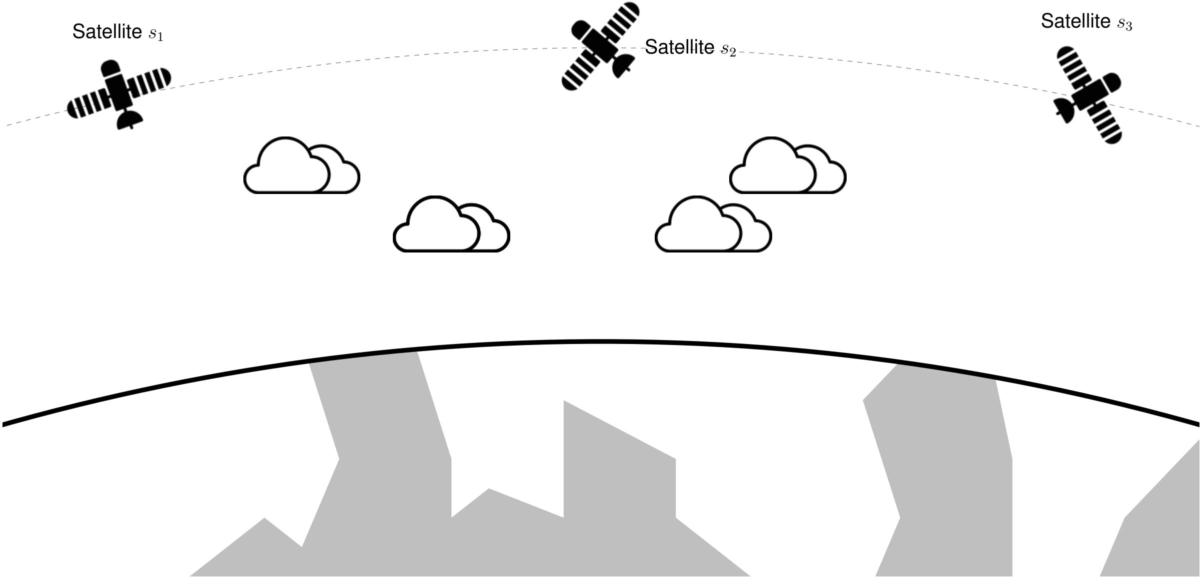
⇒ Réponse aux attentes fortes des utilisateurs pour bénéficier des avantages d'un système mutualisé (afin de réduire les coûts) et d'un système propriétaire (contrôle total et confidentialité)



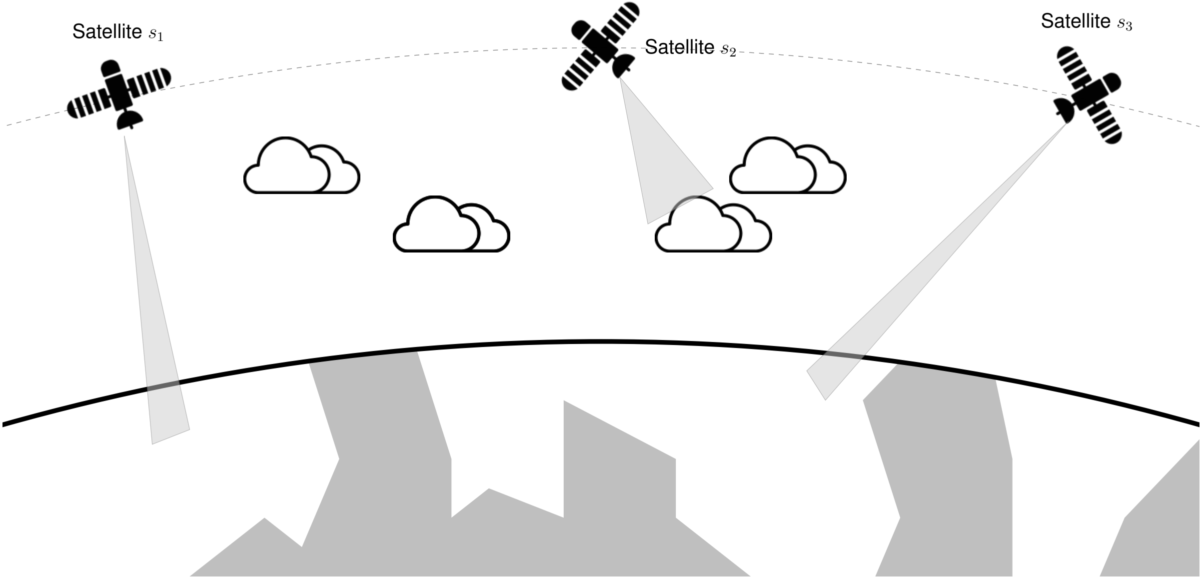
Constellation de satellites multi-utilisateurs



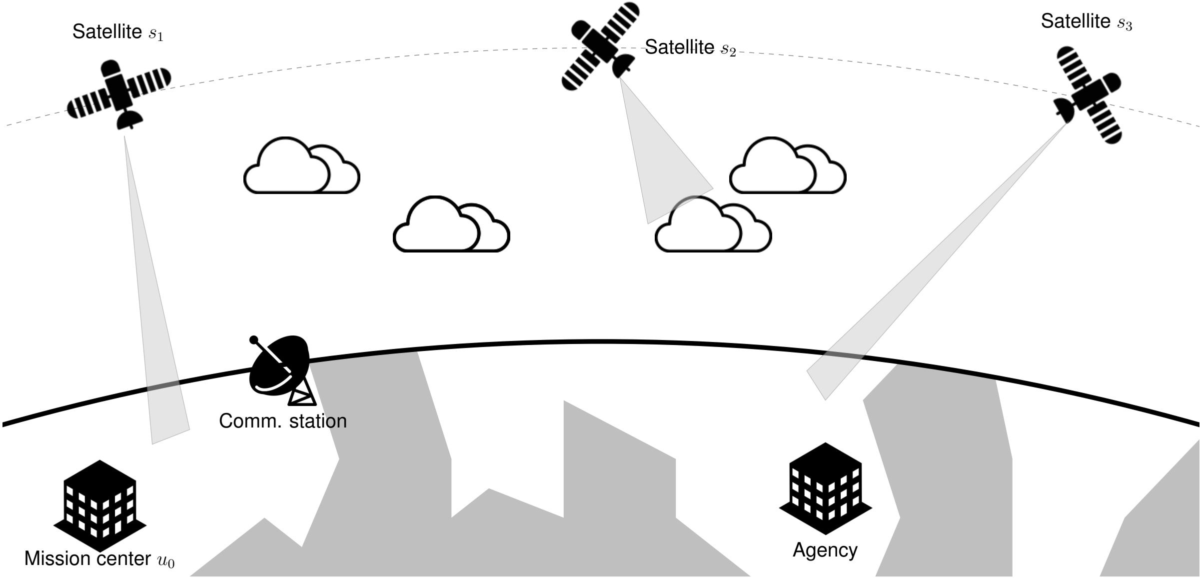
Constellation de satellites multi-utilisateurs



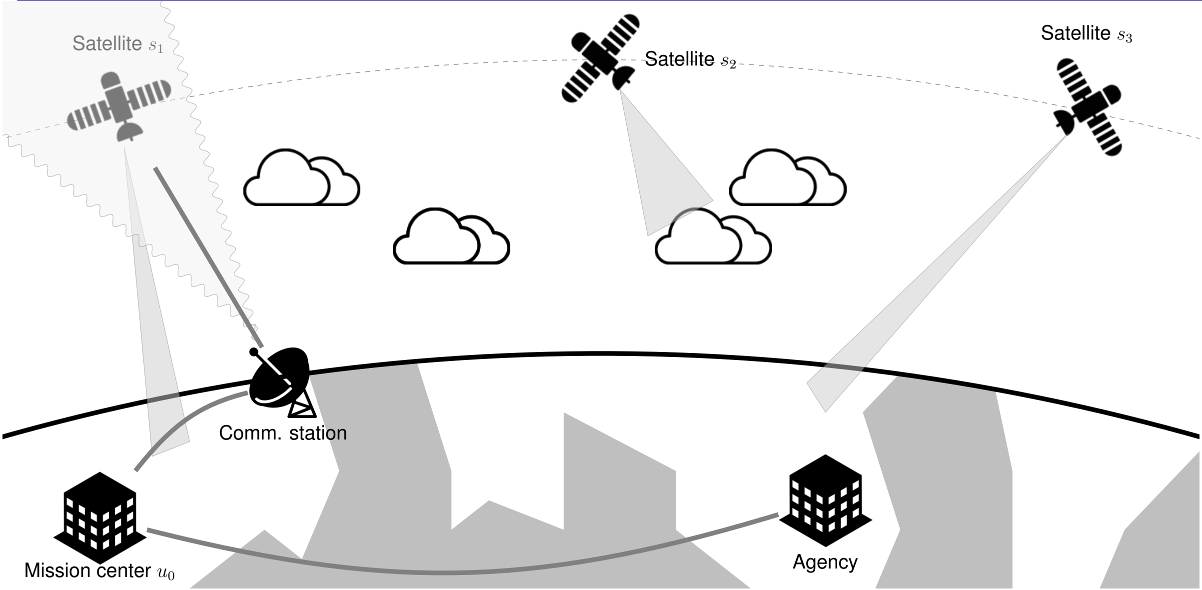
Constellation de satellites multi-utilisateurs



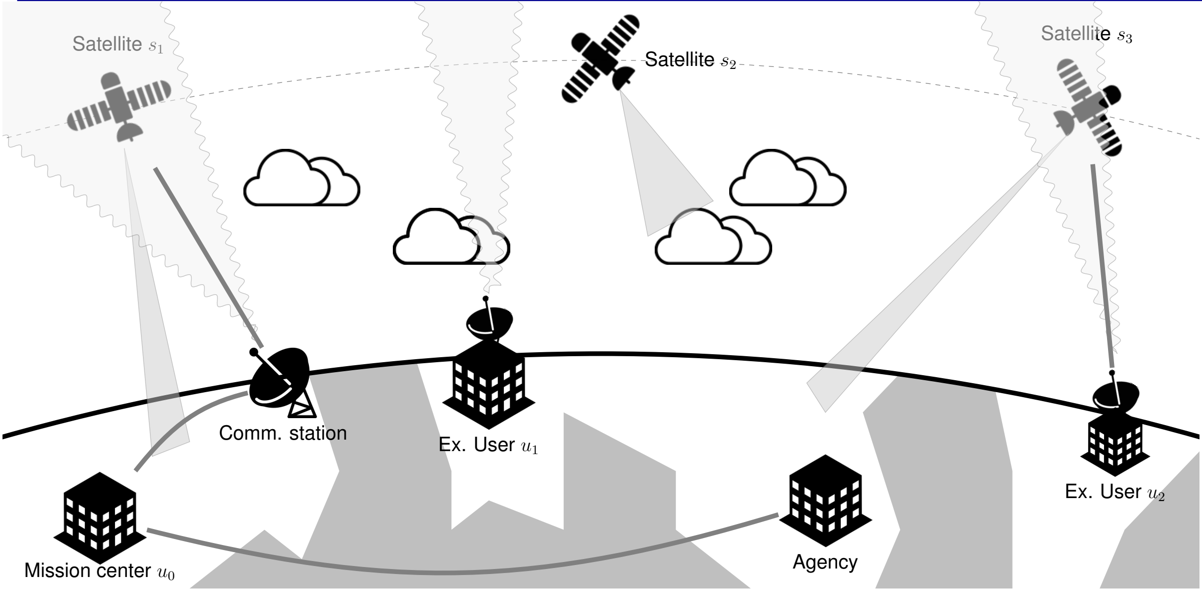
Constellation de satellites multi-utilisateurs



Constellation de satellites multi-utilisateurs



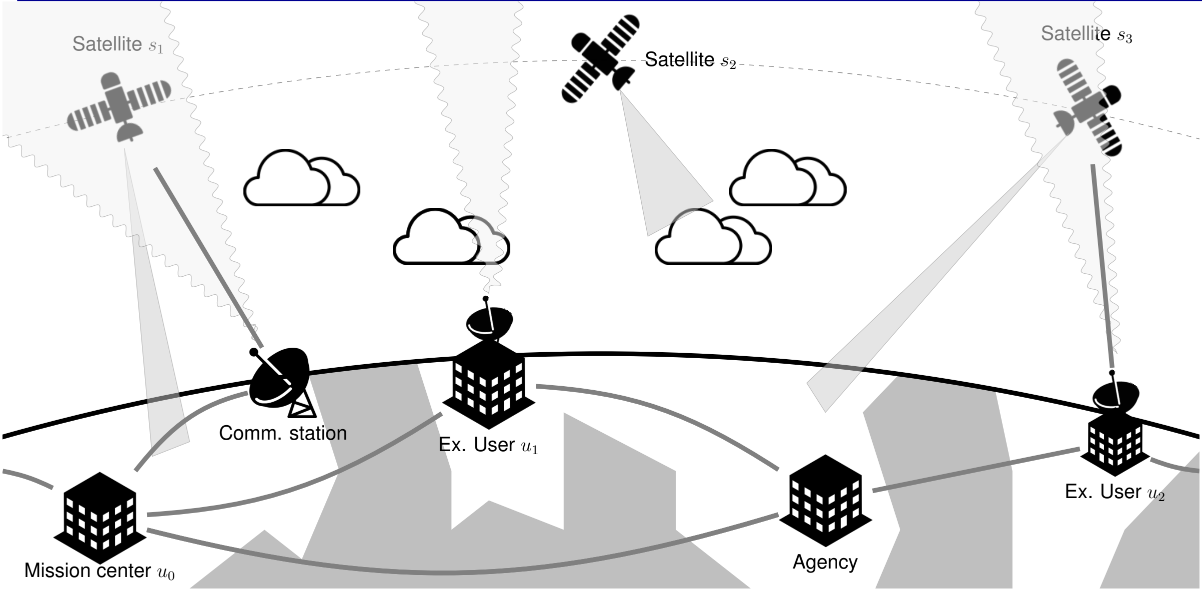
Constellation de satellites multi-utilisateurs



RÉPUBLIQUE FRANÇAISE
Liberté Égalité Fraternité

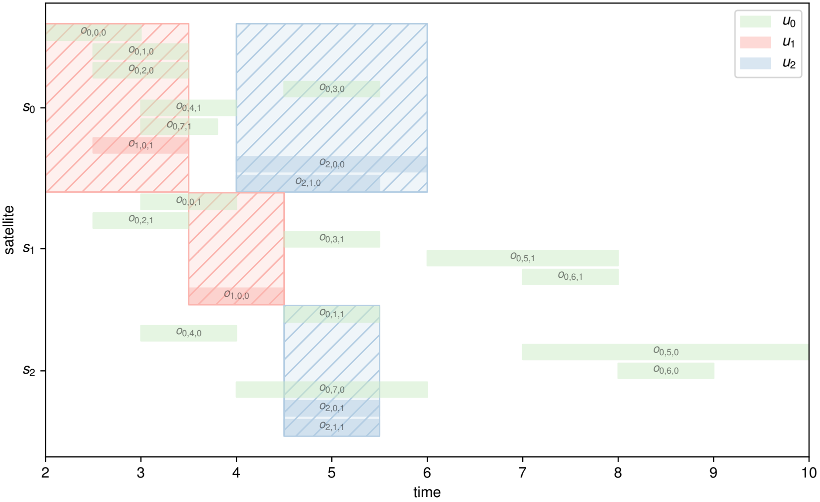
ONERA THE FRENCH AEROSPACE LAB

Constellation de satellites multi-utilisateurs



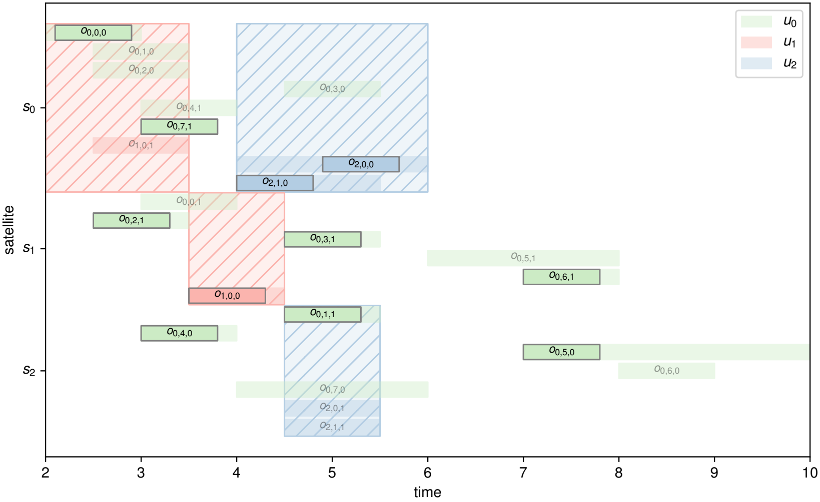
Planifier des observations sur une constellation avec de multiple portions d'orbite exclusives

Exemple jouet



Planifier des observations sur une constellation avec de multiple portions d'orbite exclusives

Exemple jouet



Plan

Définitions

Définition

Un *problème de planification de la constellation de satellites d'observation de la Terre avec des exclusivités* (ou EOSCSP) est défini par un tuple $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$, tel que $\mathcal{S}$ est un ensemble de satellites, $\mathcal{U}$ est un ensemble d'utilisateurs, $\mathcal{R}$ est un ensemble de requêtes, et $\mathcal{O}$ est un ensemble d'observations à programmer pour répondre aux requêtes de $\mathcal{R}$.

Définition

Un *satellite* est défini comme un tuple $s = \langle t_s^{\text{start}}, t_s^{\text{end}}, \kappa_s, \tau_s \rangle$ avec $t_s^{\text{start}} \in \mathbb{R}$ l'heure de début de son plan d'orbite, $t_s^{\text{end}} \in \mathbb{R}$ l'heure de fin de son plan d'orbite, $\kappa_s \in \mathbb{N}^+$ sa capacité (i. e. le nombre maximum d'observations pendant son plan d'orbite), $\tau_s : \mathcal{O} \times \mathcal{O} \rightarrow \mathbb{R}$ la fonction définissant le temps de transition entre deux observations données.

Définitions (cont.)

Définition

Un *utilisateur* est défini comme un tuple $u = \langle e_u, p_u \rangle$ avec un ensemble (éventuellement vide) de fenêtres temporelles exclusives $e_u = \{(s, (t^{\text{start}}, t^{\text{end}})) \mid s \in \mathcal{S}, [t^{\text{start}}, t^{\text{end}}] \subseteq [t_s^{\text{start}}, t_s^{\text{end}}])\} \subset (\mathcal{S} \times (\mathbb{R} \times \mathbb{R}))$, et une priorité $p_u \in \mathbb{N}^+$ (utilisée en cas de conflit). On note $\mathcal{U}^{\text{ex}}$ (resp. $\mathcal{U}^{\text{nex}}$) l'ensemble des utilisateurs possédant (resp. ne possédant pas) des exclusivités.

Définition

Une *requête* est définie comme un tuple $r = \langle t_r^{\text{start}}, t_r^{\text{end}}, \Delta_r, \rho_r, p_r, u_r, \theta_r \rangle$, avec une fenêtre temporelle de validité définie par $t_r^{\text{start}} \in \mathbb{R}$ et $t_r^{\text{end}} \in \mathbb{R}$, une durée $\Delta_r \in \mathbb{R}$, une récompense $\rho_r \in \mathbb{R}$ si r est réalisée, une position GPS pour observer p_r, un émetteur $u_r \in \mathcal{U}$ et une liste $\theta_r \in 2^{\mathcal{O}}$ d'opportunités d'observation pour valider la requête.

Définitions (cont.)

Définition

Une *observation* est définie comme un tuple $o = \langle t_o^{\text{start}}, t_o^{\text{end}}, \Delta_o, r_o, \rho_o, s_o, u_o, p_o \rangle$, avec une fenêtre temporelle de validité définie par $t_o^{\text{start}} \in \mathbb{R}$ et $t_o^{\text{end}} \in \mathbb{R}$, une requête r_o à laquelle elle contribue, une durée $\Delta_o \in \mathbb{R}$, une récompense $\rho_o \in \mathbb{R}$ (héritée de r_o), un satellite s_o sur lequel cette observation peut être planifiée, un émetteur $u_o \in \mathcal{U}$ (hérité de r_o), et une priorité $p_o \in \mathbb{N}^+$ (héritée de r_o).

Définition

Une *solution* à un EOSCSP est une allocation $\mathcal{M} = \{(o,t) \mid o \in \mathcal{O}, t \in [t_o^{\text{start}}, t_o^{\text{end}}]\}$ associant une heure de début à au plus une observation par requête de sorte que les utilisateurs exclusifs aient leurs observations planifiées sur leurs fenêtres exclusives respectives, et que la récompense globale soit maximisée (somme des récompenses des observations planifiées) : $\arg\max_{\mathcal{M}} \sum_{(o,t)\in\mathcal{M}} \rho_o$.

Définition

Un *EOSCSP pour l'utilisateur* u, noté $P[u] = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}[u], \mathcal{O}[u] \rangle$ (ou EOSCSP$[u]$), est un EOSCSP, sous-problème d'un autre EOSCSP $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$ limité aux requêtes et observations appartenant à l'utilisateur u, où $\mathcal{R}[u] = \{r \mid r \in \mathcal{R}, u_r = u\} \subseteq \mathcal{R}$ et $\mathcal{O}[u] = \{o \mid o \in \mathcal{O}, u_o = u\} \subseteq \mathcal{O}$.

Plan

Approches centralisées

Avant d'explorer des méthodes distribuées, modélisons EOSCSP de manière centralisée

- Programme linéaire en nombres mixtes (MILP) à résoudre avec des solveurs classiques
- Méthode gloutonne pour passer à l'échelle

Même dans un environnement distribué, il est envisageable de recueillir toutes les données des utilisateurs et de résoudre le problème de manière centralisée, au prix d'une divulgation massive d'informations

Modèle MILP

Solution optimale ***mais*** informations centralisées

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

Approches centralisées (cont.)

- Exclusivités : les observations pour des utilisateurs exclusifs ont une plus grande priorité pour forcer leur positionnement
- Passage à l'échelle: résoudre le MILP ne marche que sur de petites instances (e.g. < 4 satellites, < 4 users, < 100 observations)

$\Rightarrow$ Approche gloutonne:

- Trier les requêtes par priorité, récompense et date de début
- Positionner les requêtes de manière itérative sur le premier créneau disponible

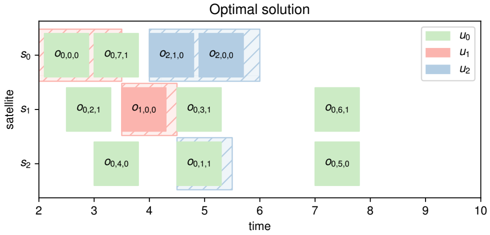
Résolution gloutonne

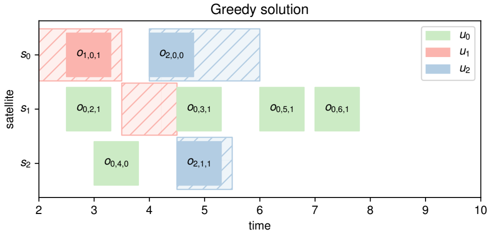
Très rapide ***mais*** sous-optimale et informations centralisées

Algorithme 1 : Solveur greedy

Données : Un EOSCSP $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$
Résultat : Une allocation $\mathcal{M}$

```
M ← {}
O^sorted ← sort(O)
R ← {(s, []) | s ∈ S}
pour chaque o ∈ O^sorted faire
    t ← first_slot(o, P, R)
    si t ≠ ∅ alors
        M ← M ∪ {(o, t)}
        O^sorted ← O^sorted \ θ(r_o)
retourner S
```





RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

Plan

Approches distribuées

Comment concevoir des schémas de communication pour résoudre collectivement EOSCSP ?

Schémas explorés

- ex2nex : approche gloutonne distribuée où les utilisateurs exclusifs positionnent d'abord leurs observations, puis les communiquent au planificateur central qui positionnent les observations restantes
- nex2ex : le planificateur central positionne les observations en dehors des portions exclusives puis demandes aux utilisateurs exclusifs de positionner les observations restantes
- itnex2ex : version itérative de nex2ex, pour éviter le surbooking

Résolution simple par communication d'exclusifs à non-exclusif (ex2nex)

Distribuée ***mais*** sous-optimale et non respectueuse de la confidentialité

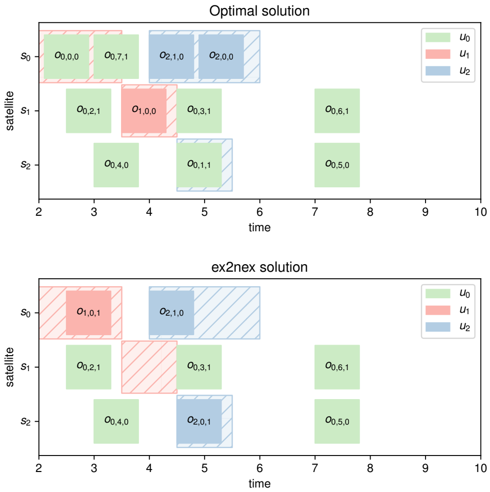
- Version distribuée de greedy
- Les plans des utilisateurs exclusifs doivent être communiqués au planificateur central

Algorithme 2 : Solveur ex2nex

Données : Un EOSCSP $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$
Résultat : Une allocation $\mathcal{M}$

pour chaque $u \in \mathcal{U}^{\text{ex}}$ **faire en parallèle**
  $\mathcal{M}_u \leftarrow \texttt{solve}(P[u])$

retourner $\mathit{solve}(P[u_0| \bigcup_{u \in \mathcal{U}^{ex}} \mathcal{M}_u])$



Résolution simple par communication non-exclusif à exclusifs

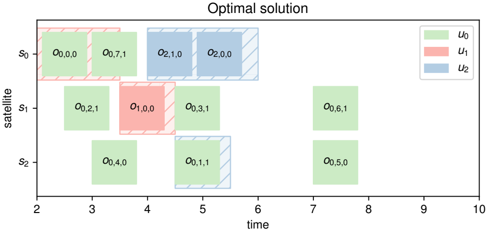
Distribuée ***mais*** sous-optimale avec chevauchements et surbookings

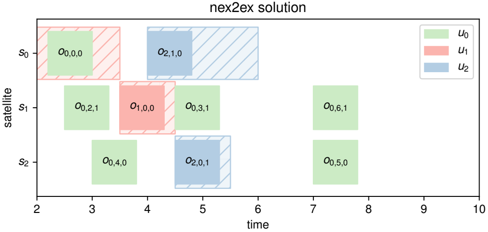
- Approche symétrique à ex2nex
- Les utilisateurs exclusifs ne partagent pas leur plans

Algorithme 3 : Solveur nex2ex

Données : Un EOSCSP $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$
Résultat : Une allocation $\mathcal{M}$

$\mathcal{M} \leftarrow \texttt{solve}(P[u_0])$
pour chaque $u \in \mathcal{U}^{\text{ex}}$ **faire**
    $\mathcal{M}_u \leftarrow \texttt{solve}(P[u, u_0|\mathcal{M}])$
    $\mathcal{M}'_u \leftarrow \{(o,t) \in \mathcal{M}_u | u_o \in \mathcal{U}^{\text{nex}}\}$
    `// send` $\mathcal{M}'_u$ `to` u_0
retourner $\mathcal{M} \cup \bigcup_{u \in \mathcal{U}^{ex}} \mathcal{M}_u$





Résolution itérative

Distribuée ***mais*** sous-optimale et non coordonnée

- Réduction des surbookings de nex2ex
- Allocation comme un processus itératif

Algorithme 4 : Solveur itnex2ex

Données : Un EOSCSP $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$

Résultat : Une allocation $\mathcal{M}$

$\mathcal{M} \leftarrow \texttt{solve}(\overline{P[u_0]})$

$\mathcal{O}^{\text{sorted}} \leftarrow \texttt{sort}(\mathcal{O} \setminus \{o | (o, t) \in \mathcal{M}\})$

pour chaque $o \in \mathcal{O}^{\textit{sorted}}$ **faire**

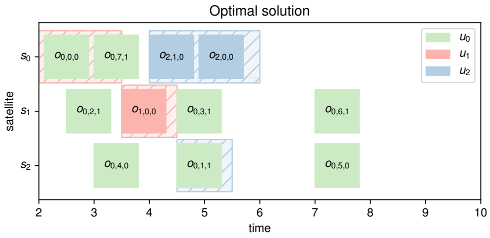
  pour chaque $u \in \textit{candidates}(o, \mathcal{U}^{\textit{ex}})$ **faire**

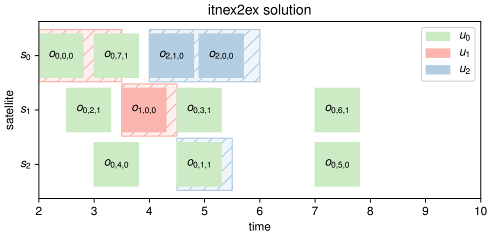
    $\mathcal{M}_u \leftarrow \texttt{solve}(P[u, u_0 | \mathcal{M}] \cup P[o])$

    $\mathcal{M}'_u \leftarrow \{(o, t) \in \mathcal{M}_u | u_o \in \mathcal{U}^{\text{nex}}\}$

    `// envoyer` $\mathcal{M}'_u$ `à` u_0

retourner $\mathcal{M} \cup \bigcup_{u \in \mathcal{U}^{\textit{ex}}} \mathcal{M}'_u$





Plan

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

Coordination et optimisation distribuée

- Comment apporter de la coordination entre les utilisateurs exclusifs ?
- Comment décider de manière collective de quelles requêtes planifier ?

$\Rightarrow$ Optimisation sous contraintes distribuées (DCOP)

À propos des DCOPs

Une façon de modéliser les problèmes de coordination entre agents consiste à les formaliser dans le cadre des problèmes d'optimisation distribuée sous contraintes (DCOP) [PETCU and FALTINGS, 2005a].

Définition

Un *problème d'optimisation sous contraintes distribuées* discret (ou DCOP) est un tuple $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{C}, \mu \rangle$, où :

- $\mathcal{A} = \{a_1, \ldots, a_{|A|}\}$ est un ensemble d'agents
- $\mathcal{X} = \{x_1, \ldots, x_n\}$ sont des variables appartenant aux agents
- $\mathcal{D} = \{\mathcal{D}_{x_1}, \ldots, \mathcal{D}_{x_n}\}$ est un ensemble de domaines finis, tel que la variable x_i prend ses valeurs dans $\mathcal{D}_{x_i} = \{v_1, \ldots, v_k\}$
- $\mathcal{C} = \{c_1, \ldots, c_m\}$ est un ensemble de contraintes, où chaque c_i définit un coût $\in \mathbb{R}^+ \cup \{+\infty\}$ pour chaque combinaison de l'affectation à un sous-ensemble de variables (une contrainte est initialement connue seulement des agents impliqués)
- $\mu : \mathcal{X} \rightarrow \mathcal{A}$ est une fonction associant chaque variable avec l'agent qui la gère
- $f : \prod \mathcal{D}_{x_i} \rightarrow \mathbb{R}$ est un objectif représentant le coût global d'une affectation complète de valeurs aux variables

À propos des DCOPs (cont.)

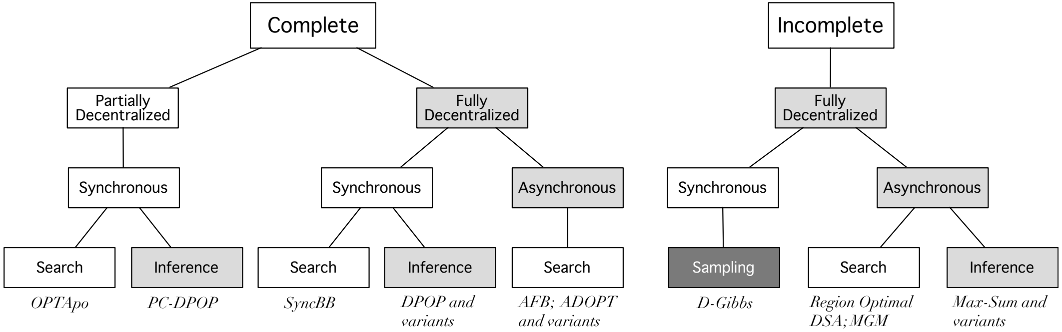
L'objectif d'optimisation est représenté par la fonction f, qui, en général, est considérée comme la somme des coûts : $f = \sum_i c_i$

Définition (Solution)

Une *solution* à un DCOP P est une affectation complète à toutes variables
Une solution est *optimale* si elle minimise f

Méthodes de résolution de DCOP

Voir FIORETTO et al., 2018



Étendre itnex2ex avec DCOP

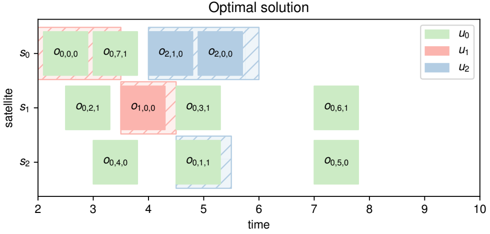
Distribuée et coordonnée ***mais*** sous-optimale

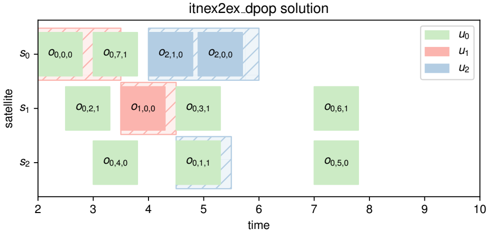
- On considère les requêtes de manière itérative
- Les utilisateurs exclusifs se coordonnent pour choisir les requêtes

Algorithme 5 : Solveur itnex2ex_DCOP

Données : Un EOSCSP $P = \langle \mathcal{S}, \mathcal{U}, \mathcal{R}, \mathcal{O} \rangle$
Résultat : Une allocation $\mathcal{M}$

$\mathcal{M} \leftarrow \texttt{solve}(\overline{P[u_0]})$
pour chaque $u \in \mathcal{U}^{\text{ex}}$ **faire en parallèle**
  $\mathcal{M}_u \leftarrow \texttt{solve}(P[u])$
$\mathcal{R}^{\text{sorted}} \leftarrow \texttt{sort}(\mathcal{R} \setminus \{r | (o,t) \in \mathcal{M}, o \in \theta_r\})$
pour chaque $r \in \mathcal{R}^{sorted}$ **faire**
  $p \leftarrow \texttt{build\_DCOP}(\theta_r, \mathcal{M}, \mathcal{M}_{u_1}, \ldots, \mathcal{M}_{u_n}, P)$
  $\mathcal{M}_{u_1}, \ldots, \mathcal{M}_{u_n} \leftarrow \texttt{solve\_DCOP}(p)$
  pour chaque $u \in \mathcal{U}^{\text{ex}}$ **faire en parallèle**
    $\mathcal{M}'_u \leftarrow \{(o,t) \in \mathcal{M}_u | u_o \in \mathcal{U}^{\text{nex}}\}$
    $\texttt{// send } \mathcal{M}'_u \texttt{ to } u_0$
retourner $solve(\overline{P[u_0]} | \mathcal{M} \cup \bigcup_{u \in \mathcal{U}^{\text{ex}}} \mathcal{M}_u)$





Modèle DCOP

Une instance de DCOP est définie étant donnés une requête r, et un plan $(\mathcal{M}, \mathcal{M}_{u_1}, \ldots, \mathcal{M}_{u_n})$

- Les agents sont les utilisateurs exclusifs pouvant potentiellement remplir la requête r:

$$\mathcal{A} = \{u \in \mathcal{U}^{\text{ex}} | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \exists o \in \theta_r \text{ t.q. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\} \quad (13)$$

- Notons $\mathcal{O}[u]^r = \{o \in \theta r | \exists (s, (t_u^{\text{start}}, t_u^{\text{end}})) \in e_u, \text{ t.q.. } s_o = s, [t_u^{\text{start}}, t_u^{\text{end}}] \cap [t_o^{\text{start}}, t_o^{\text{end}}] \neq \emptyset\}$ les opportunités relatives à r pouvant être positionnées sur les portions de u
- Chaque agent gère des variables de décision binaires, une pour chaque observation $o \in \mathcal{O}[u]^r$ et portion exclusive $e \in e_u$, spécifiant si o est positionnée dans e:

$$\mathcal{X} = \{x_{e,o} | e \in \textstyle\bigcup_{u \in \mathcal{A}} e_u, o \in \mathcal{O}[u]^r\} \quad (14)$$

$$\mathcal{D} = \{\mathcal{D}_{x_{e,o}} = \{0, 1\} | x_{e,o} \in \mathcal{X}\} \quad (15)$$

- L'application μ associe chaque variable $x_{e,o}$ au propriétaire de e

Modèle DCOP (cont.)

- Les contraintes doivent garantir qu'au plus une observation est positionnée par requête (16), que les satellites ne sont pas surchargés (17), et que au plus un agent sert une même observation (18)

$$\sum_{e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq 1, \quad \forall u \in \mathcal{X}, \forall o \in \mathcal{O}[u]^r \tag{16}$$

$$\sum_{o \in \{o \in \mathcal{O}[u]^r | u \in \mathcal{A}, s_o = s\}, e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq \kappa_s^*, \ \forall s \in \mathcal{S} \tag{17}$$

$$\sum_{e \in \bigcup_{u \in \mathcal{A}} e_u} x_{e,o} \leq 1, \quad \forall o \in \mathcal{O} \tag{18}$$

- Le coût pour intégrer une observation dans le plan courant d'un utilisateur doit être évalué pour guider le processus d'optimisation

$$c(x_{e,o}) = \pi(o, \mathcal{M}_{u_o}), \quad \forall x_{e,o} \in \mathcal{X} \tag{19}$$

  où π évalue le meilleur coût obtenu en planifiant o et tout combinaison d'observations de $\mathcal{M}_{u_o}$, afin de considérer toutes les révisions possibles du plan courant de u_o

- Ces alternatives sont évaluées en utilisant greedy

$$\mathcal{C} = \{(16), (17), (18), (19)\} \tag{20}$$

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

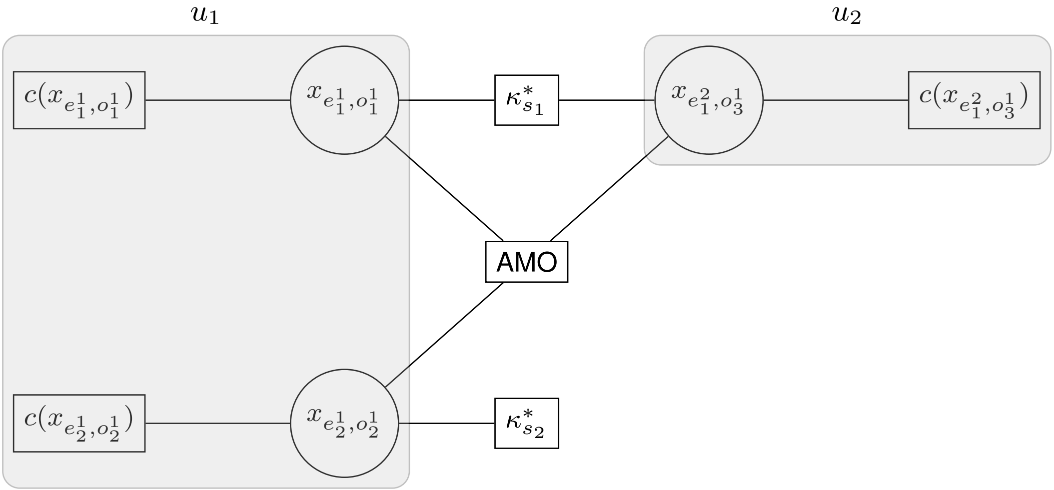
Exemple

Considérons la réquête r_1 émise par un utilisateur non exclusif (u_0) pour laquelle 3 opportunités d'observations (o_1^1, o_2^1 et o_3^1) sont positionnables sur 3 fenêtres exclusives (e_1^1, e_2^1 et e_1^2) appartenant à deux utilisateurs exclusifs (u_1 et u_2), sur 2 satellites s_1 et s_2
Le DCOP a résoudre est le suivant :

- $\mathcal{A} = \{u_1, u_2\}$
- $\mathcal{X} = \{x_{e_1^1,o_1^1}, x_{e_2^1,o_2^1}, x_{e_1^2,o_3^1}\}$
- $\mathcal{C} = \left\{ \begin{array}{c} x_{e_1^1,o_1^1} + x_{e_2^1,o_2^1} + x_{e_1^2,o_3^1} \leq 1 \\ x_{e_1^1,o_1^1} + x_{e_1^2,o_3^1} \leq \kappa^*_{s_1} \\ x_{e_2^1,o_2^1} \leq \kappa^*_{s_2} \\ c(x_{e_1^1,o_1^1}) = \pi(o_1^1, \mathcal{M}_{u_1}) \\ c(x_{e_2^1,o_2^1}) = \pi(o_2^1, \mathcal{M}_{u_1}) \\ c(x_{e_1^2,o_3^1}) = \pi(o_3^1, \mathcal{M}_{u_2}) \end{array} \right\}$

Pour résoudre un tel DCOP, nous utiliserons DPOP [PETCU and FALTINGS, 2005b]

Exemple (cont.)



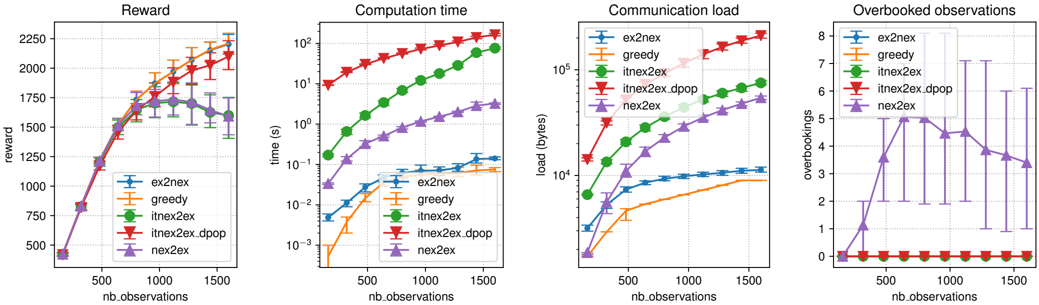
Plan

Cadre expérimental

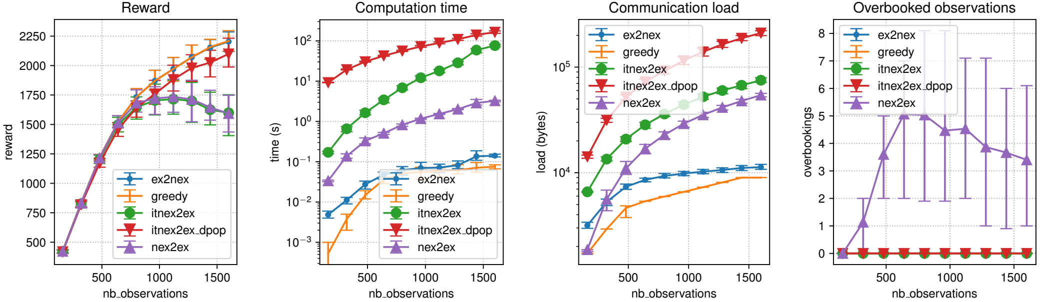
EOSCSPs générés aléatoirement

	Conflictuels	Réalistes
Satellites	3	8
Horizon	[0,300]	[0,21600]
Capacité	20	500
Utilisateurs	5	6
Requêtes par utilisateur exclusif	[2:20:2]	[20:20:200]
Requêtes générique	[8:80:8]	[100:1000:100]
Portions exclusives par utilisateur	8	10
Durée des portions	[15:20]	[300:600]
Observations par requête	10	5
Durée des observations	5	20
Durée des fenêtres d'observation	[10:20]	[40:60]
Récompenses pour les utilisateurs exclusifs	[10:50:10]	[10:50:10]
Récompenses pour les requêtes génériques	[1:5]	[1:5]
Temps de transition	1	1

Problèmes très conflictuels

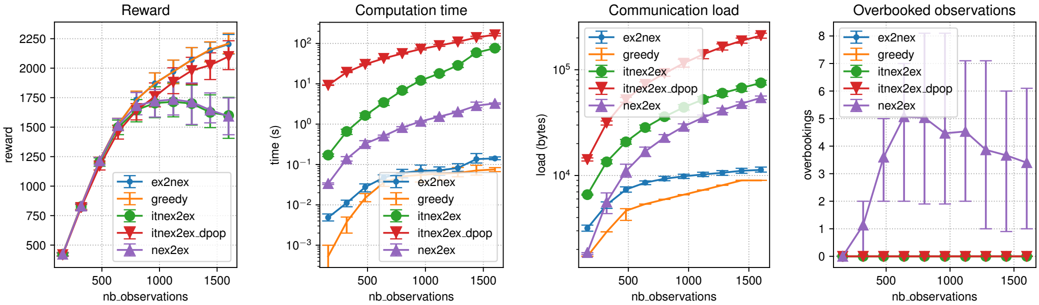


Problèmes très conflictuels



- itnex2ex_dpop offre des performances légèrement en-deça des approches sans privacité
- Les approches sans coordination affichent des performances se dégradant avec une augmentation du nombre de requêtes (surbooking et surcharge)
- Les performances de itnex2ex_dpop sont au prix d'un temps de calcul supplémentaire
- La coordination (itnex2ex_dpop) génère une légère charge de communication supplémentaire

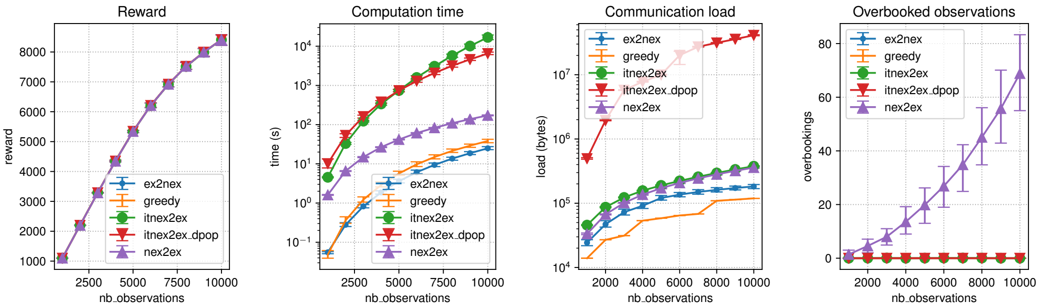
Problèmes très conflictuels



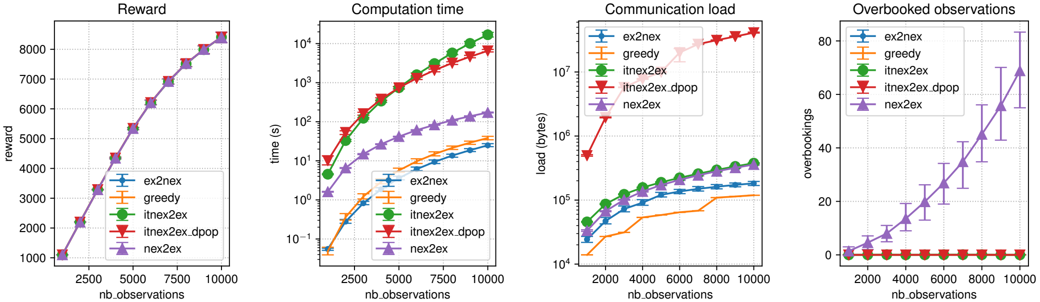
- itnex2ex_dpop offre des performances légèrement en-deça des approches sans privacité
- Les approches sans coordination affichent des performances se dégradant avec une augmentation du nombre de requêtes (surbooking et surcharge)
- Les performances de itnex2ex_dpop sont au prix d'un temps de calcul supplémentaire
- La coordination (itnex2ex_dpop) génère une légère charge de communication supplémentaire

itnex2ex_dpop fournit de bonnes solutions sans divulguer d'informations sur les utilisateurs exclusifs

Problèmes réalistes



Problèmes réalistes



- Tous les algorithmes fournissent des solutions de bonne qualité, sauf nex2ex
- Tous les algorithmes restent en dessous de 180 minutes de calcul
- itnex2ex_dpop nécessite l'échange de 30 Mo pour des instances plus grandes
- Ces instances ne sont pas trop conflictuelles

Plan

Conclusions

Résumé

- Proposition du modèle EOSCSP
- Schémas de résolution centralisés : aucune confidentialité
- Schémas distribués non coordonnés: confidentialité mais surcharge et surbooking
- Schéma distribué coordonné: confidentialité, performances équivalentes à greedy

Futurs développements

- Extension à des requêtes plus complexes
- Gestion des incertitudes
- Approximation ou compilation de π
- Autres schémas de coordination à explorer :
  - Optimisation de consensus
  - Enchères [PHILLIPS and PARRA, 2021; PICARD, 2021]

Ouverture et défis

Présentation à APIA (session du jeudi 1er Juillet 2021, 10h30-12h30)

Gauthier PICARD, Clément CARON, Jean-Loup FARGES, Jonathan GUERRA, Cédric PRALET, and Stéphanie ROUSSEL [2021]. “Défis ouverts aux systèmes multi-agents dans le cadre des constellations de satellites d’observation de la Terre”. In: *Conférence Nationale sur les Applications Pratiques de l’Intelligence Artificielle (APIA 2021)*

Remerciements

Ces travaux ont été menés grâce au financement du gouvernement français dans le contexte du Programme d'Invertissements d'Avenir, au travers du projet BPI PSPC "LiChIE" coordonné par Airbus Defence and Space

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

Planification multi-utilisateurs et multi-satellites de tâches d'observation dans des constellations avec portions d'orbites exclusives

Gauthier Picard
gauthier.picard@onera.fr

ONERA, DTIS-SYD, Université de Toulouse

29 juin 2021

Références

FIORETTO, F., E. PONTELLI, and W. YEOH (2018). "Distributed Constraint Optimization Problems and Applications: A Survey". In: *Journal of Artificial Intelligence Research* 61, pp. 623–698.

PETCU, A. and B. FALTINGS (2005a). "A scalable method for multiagent constraint optimization". In: *International Joint Conference on Artificial Intelligence (IJCAI'05)*, pp. 266–271.

— (2005b). "Superstabilizing, Fault-containing Distributed Combinatorial Optimization". In: *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI'05)*. AAAI Press, pp. 449–454. ISBN: 1-57735-236-X. URL: http://dl.acm.org/citation.cfm?id=1619332.1619405.

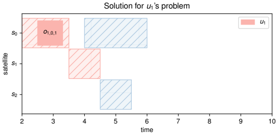
PHILLIPS, Sean and Fernando PARRA (2021). "A Case Study on Auction-Based Task Allocation Algorithms in Multi-Satellite Systems". In: *AIAA Scitech 2021 Forum*. DOI: 10.2514/6.2021-0185. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2021-0185. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2021-0185.

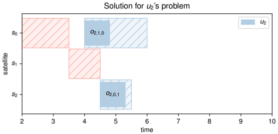
PICARD, Gauthier (2021). "Auction-based and Distributed Optimization Approaches for Scheduling Observations in Satellite Constellations with Exclusive Orbit Portions". In: *International Workshop on Planning and Scheduling for Space (IWPSS'21)*. arXiv: 2106.03548. URL: https://arxiv.org/abs/2106.03548.

PICARD, Gauthier, Clément CARON, Jean-Loup FARGES, Jonathan GUERRA, Cédric PRALET, and Stéphanie ROUSSEL (2021). "Défis ouverts aux systèmes multi-agents dans le cadre des constellations de satellites d'observation de la Terre". In: *Conférence Nationale sur les Applications Pratiques de l'Intelligence Artificielle (APIA 2021)*.

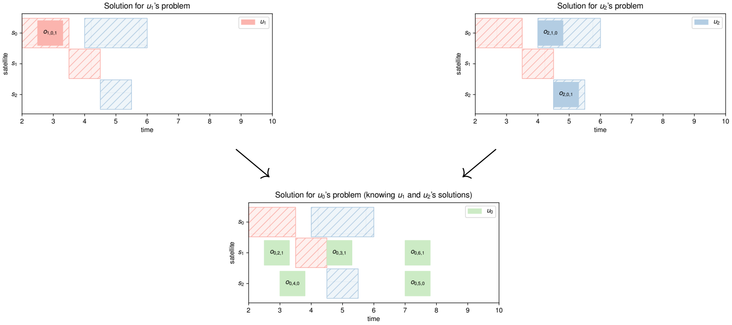
SHAH, Vishwa, Vivek VITTALDEV, Leon STEPAN, and Cyrus FOSTER (2019). "Scheduling the World's Largest Earth-Observing Fleet of Medium-Resolution Imaging Satellites". In: *IWPSS*.

Exemple d'exécution de ex2nex

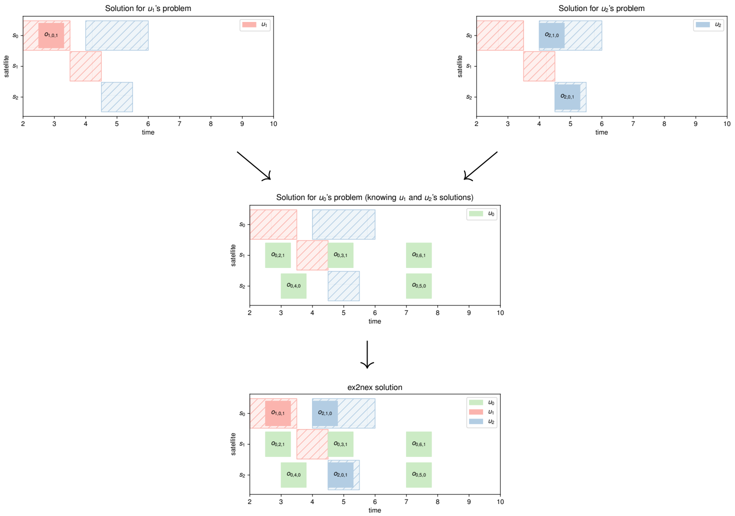




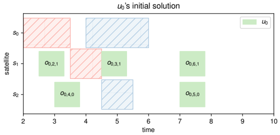
Exemple d’exécution de ex2nex



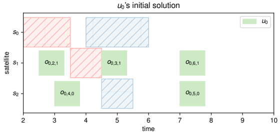
Exemple d’exécution de ex2nex

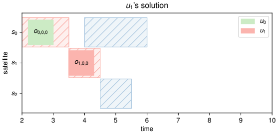


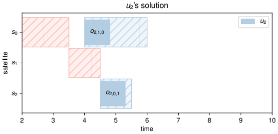
Exemple d’exécution de nex2ex



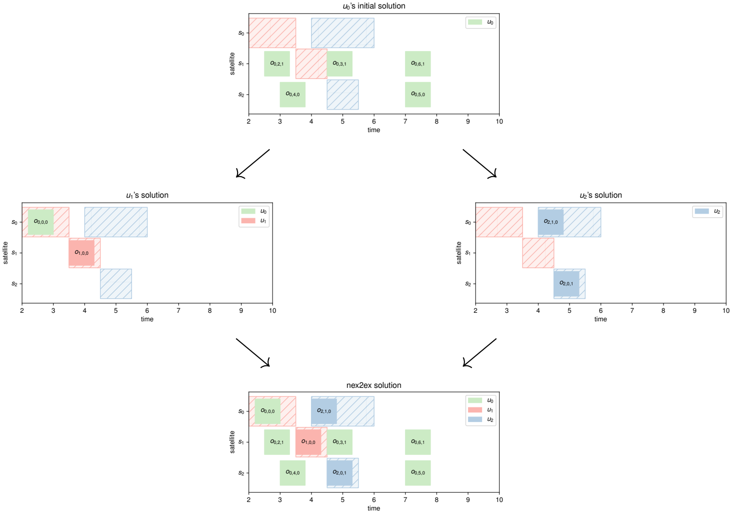
Exemple d'exécution de nex2ex



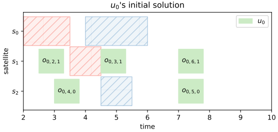




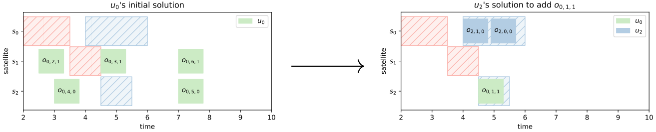
Exemple d’exécution de nex2ex



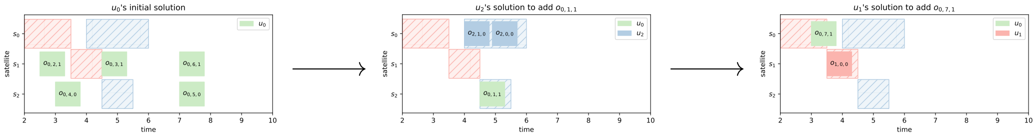
Exemple d'exécution de itnex2ex



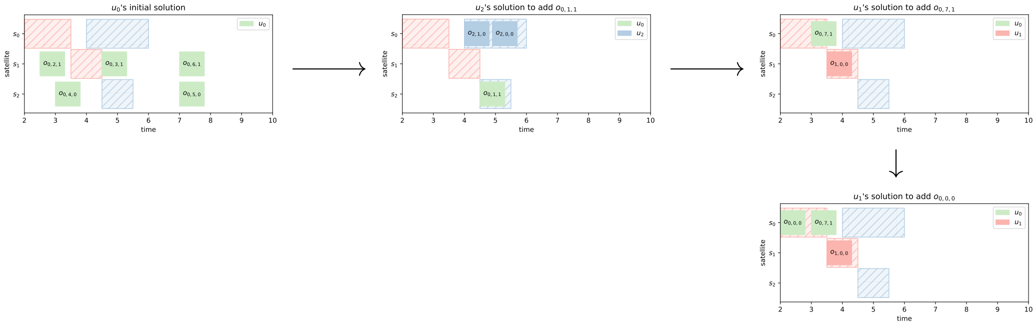
Exemple d’exécution de itnex2ex



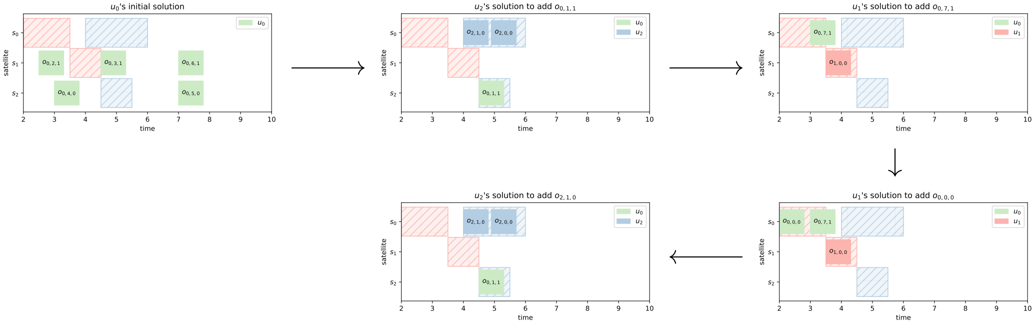
Exemple d’exécution de itnex2ex



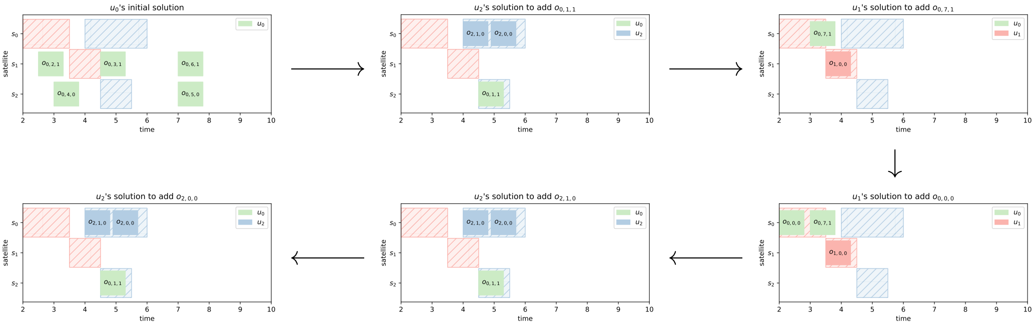
Exemple d’exécution de itnex2ex



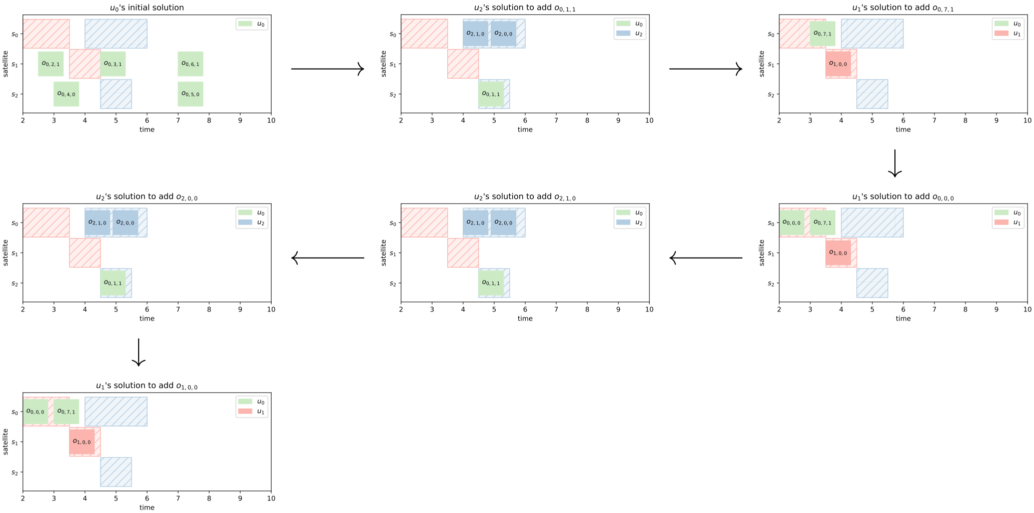
Exemple d'exécution de itnex2ex



Exemple d'exécution de itnex2ex



Exemple d’exécution de itnex2ex



Exemple d'exécution de itnex2ex