

Ordonnancement avec modes pour la réalisation de requêtes complexes par une constellation de satellites d'observation

Samuel Squillaci, Stéphanie Roussel, Cédric Pralet

ONERA/DTIS, Université de Toulouse, F-31055 Toulouse, France
{samuel.squillaci,stephanie.roussel,cedric.pralet}@onera.fr

Mots-clés : *constellation de satellites d'observation, réparation itérative, ordonnancement*

1 Problématique

Nous considérons un problème dans lequel des requêtes d'observation de zones à la surface de la Terre doivent être réalisées par une constellation (un ensemble) de satellites en orbite basse. Vu de loin, ce problème correspond à un *Team Orienteering Problem with Time Windows* (TOPTW) dans lequel des véhicules (les satellites) doivent visiter des clients candidats (les zones d'observation) sachant qu'il existe pour chaque candidat une fenêtre temporelle de visite autorisée (la fenêtre pendant laquelle un satellite survole la zone concernée) [1, 2, 4].

En pratique cependant, chaque requête d'observation peut englober plusieurs opportunités d'acquisition candidates. Chaque requête peut de plus nécessiter la réalisation conjointe de plusieurs activités élémentaires, comme dans le cas des requêtes d'observation stéréoscopique ou d'observation périodique, et plus généralement dès que l'on prend en compte à la fois l'observation et le téléchargement des données vers les stations sol (processus de *vidage de données*, avec une contrainte de précedence entre activité d'observation et activité de vidage). En plus de cela, au niveau applicatif, il est intéressant d'intégrer le fait qu'en toute rigueur, le profit collecté pour une requête n'est pas forcément décomposable en une somme de profits élémentaires associés aux activités couvertes par la requête. Notre objectif est de définir un problème générique permettant de couvrir ce spectre de récompenses complexes.

2 Modélisation du problème

Pour modéliser les requêtes complexes, nous introduisons, pour chaque requête d'observation, un ensemble de *modes de réalisation* candidats. Chaque mode correspond à une manière de couvrir la requête à l'aide d'activités élémentaires d'observation et de vidage de données, et à chaque mode est associée une récompense globale qui témoigne de la qualité de réalisation de la requête dans ce mode. Cette récompense peut représenter toutes sortes de mesures : qualité des prises de vue étant donné les prévisions météo, délai de vidage après observation, distance temporelle entre observations successives, importance de la requête en elle-même...

Au niveau de chaque activité élémentaire a (observation ou vidage) réalisable sur un satellite donné de la constellation, on définit une fenêtre de début de réalisation autorisée $[Start_a, End_a]$ et une durée de monopolisation du satellite δ_a . On définit également, pour toute paire d'observations (a, a') , la durée $\tau_{aa'}$ requise par le satellite pour passer d'un pointage vers la cible associée à l'activité a à un pointage vers la cible associée à l'activité a' . A partir de là, une solution correspond à un ensemble de séquences d'activités (une séquence par satellite). Pour chaque activité a dans une séquence, nous pouvons alors définir, en utilisant la date $Start_a$ et les temps de transition requis entre activités, une date de début de réalisation au plus tôt $earliest_a$ pour a , et on dit qu'une activité est en retard lorsque $earliest_a > End_a$. L'objectif est alors de produire une solution sans retard et qui maximise la somme des récompenses associées aux modes de réalisation choisis pour les requêtes.

3 Algorithme implémenté et résultats

La notion de mode de réalisation des requêtes complexes invalide le fonctionnement des algorithmes classiques qui travaillent habituellement directement sur des insertions/retraits d'activités élémentaires. Nous avons donc développé un nouvel algorithme qui raisonne à l'échelle des insertions/retraits de modes. Nous avons tout d'abord spécifié un algorithme glouton qui à chaque étape tente d'insérer dans le plan un mode apportant la plus forte récompense et associé à une requête non déjà couverte. Un mode est effectivement inséré seulement si la solution reste réalisable après insertion.

Nous avons ensuite développé un algorithme de recherche locale fonctionnant par réparation itérative. Ce second algorithme initialise la solution avec l'algorithme glouton vu précédemment. Ensuite, les modes rejetés sont à nouveau considérés, mais cette fois un peu plus de travail est réalisé pour tenter d'intégrer chaque mode au plan courant. Plus précisément, trois types de stratégies de réparation sont successivement appliquées : (1) une stratégie cherchant des permutations sur les séquences d'observations de chaque satellite afin de diminuer les retards, (2) une stratégie qui identifie le mode m qui explique le plus le retard total de la solution et qui cherche à remplacer m par autre mode qui couvre la même requête, et (3) une stratégie qui retire des modes qui expliquent le plus le retard total. Des opérations de perturbation permettent enfin de s'échapper des optima locaux. La figure 3 présente un exemple de solution trouvée pour un problème impliquant 230 requêtes candidates sur un horizon d'une journée.

Au niveau des perspectives, l'algorithme défini doit être adapté pour prendre en compte la dépendance temporelle des durées de dépointage [3], qui peut invalider les solutions que nous produisons actuellement. Un autre point concerne la génération de modes à la volée au cours de la résolution, de manière à contourner la combinatoire relative à l'assemblage d'activités élémentaires utilisables pour définir un mode. Ces travaux sont réalisés avec le support du Programme d'Investissements d'Avenir (projet BPI PSPC LiChIE mené par AIRBUS Defence and Space).

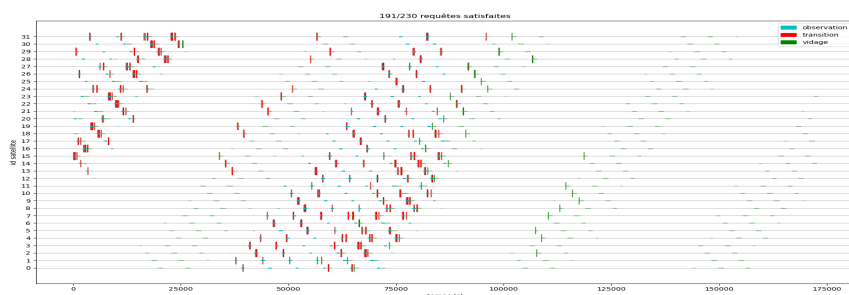


FIG. 1 – Plan trouvé pour 32 satellites (temps en abscisse et id du satellite en ordonnée). Les dépointages entre observations et les vidages sont respectivement représentés par des rectangles rouge et vert.

Références

- [1] N. Bianchessi and G. Righini. Planning and scheduling algorithms for the COSMO-SkyMed constellation. *Aerospace Science and Technology*, 12(7) :535–544, 2008.
- [2] L. He, X. Liu, G. Laporte, Y. Chen, and Y. Chen. An improved adaptive large neighborhood search algorithm for multiple agile satellites scheduling. *Computers & Operations Research*, 100 :12–25, 2018.
- [3] M. Lemaître, G. Verfaillie, F. Jouhaud, J.-M. Lachiver, and N. Bataille. Selecting and scheduling observations of agile satellites. *Aerospace Science and Technology*, 6(5) :367–381, 2002.
- [4] X. Wang, G. Wu, L. Xing, and W. Pedrycz. Agile earth observation satellite scheduling over 20 years : formulations, methods and future directions. *IEEE Systems Journal*, pages 1–12, 2020.