



**HAL**  
open science

# TDOPTW dynamique : algorithmes incrémentaux et application aux satellites d'observation

Cédric Pralet, Xavier Pucel, Stéphanie Artigues, Nicolas Théret

## ► To cite this version:

Cédric Pralet, Xavier Pucel, Stéphanie Artigues, Nicolas Théret. TDOPTW dynamique : algorithmes incrémentaux et application aux satellites d'observation. ROADEF 2020, Feb 2020, MONTPELLIER, France. hal-02904381

**HAL Id: hal-02904381**

**<https://hal.science/hal-02904381>**

Submitted on 22 Jul 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# TDOPTW dynamique : algorithmes incrémentaux et application aux satellites d'observation

Cédric Pralet<sup>1</sup>, Xavier Pucel<sup>1</sup>, Stéphanie Artigues<sup>2</sup>, Nicolas Théret<sup>2</sup>

<sup>1</sup> ONERA/DTIS, Université de Toulouse, F-31055 Toulouse, France  
{cedric.pralet,xavier.pucel}@onera.fr

<sup>2</sup> CNES, 18 av. Edouard Belin 31401 Toulouse Cedex 9, France  
{stephanie.artigues,nicolas.theret@cnes.fr}

**Mots-clés :** *TDOPTW, problème dynamique, recherche locale, satellites d'observation*

## 1 Introduction

Nous considérons une classe particulière de TSP (*Traveling Salesman Problem*) dans laquelle : (1) un profit et une fenêtre temporelle sont associés à chaque client, (2) la durée de transition minimale requise entre deux clients est *time-dependent* (fonction de la date précise à laquelle la transition est réalisée), (3) il n'est pas forcément possible de servir tous les clients si bien qu'un sous-ensemble d'entre eux doit être sélectionné de manière à maximiser le profit total, et (4) l'arrivée dynamique de nouveaux clients requiert des opérations de réoptimisations rapides de l'ensemble des clients sélectionnés et de la séquence de visite de ces clients.

Techniquement, le problème abordé est un TDOPTW (*Time-Dependent Orienteering Problem with Time Windows* [1]) dynamique. L'étude de ce problème est motivée par une application de gestion d'un satellite d'observation de la Terre [2]. La mission d'un tel satellite (le véhicule) consiste à réaliser des observations de zones au sol (les clients) demandées par des utilisateurs. Chaque zone  $z$  est survolée par le satellite seulement pendant certaines fenêtres temporelles calculables à partir des paramètres orbitaux de la trajectoire du satellite, et la réalisation de  $z$  apporte une certaine récompense. L'aspect *time-dependent* vient du fait que la durée minimale requise pour passer d'un pointage vers une zone à un pointage vers la zone suivante dépend de la date à laquelle la transition est enclenchée. Ces durées minimales sont calculées via des appels à une bibliothèque externe de calculs cinématiques. Notons que le problème de planification d'observations pour un satellite est connu pour être NP-difficile [2].

Une des nouveautés est que dans le cadre de la démocratisation des données spatiales, il existe un besoin de répondre de manière plus réactive à de nouvelles demandes d'observation formulées au fil de l'eau par les utilisateurs du système, d'où l'aspect dynamique lié au besoin de réoptimiser régulièrement le plan d'observation en fonction des nouvelles demandes.

## 2 Algorithmes incrémentaux de type recherche locale

Nous avons défini plusieurs algorithmes pour traiter ce nouveau concept de *programmation réactive* des satellites, dans lequel le plan courant est "perturbé" suite à l'arrivée de nouvelles demandes. Le problème global est tout d'abord décomposé en un ensemble de sous-problèmes élémentaires qui correspondent chacun à une *composante connexe* (groupe de zones à observer qui ne peuvent pas être traitées séparément du fait de la proximité temporelle de leurs fenêtres de survol). Ensuite, comme illustré à la Figure 1a, l'algorithme de base maintient une liste de demandes candidates et une liste de demandes rejetées. Tant qu'il reste des demandes candidates pour une insertion, l'algorithme choisit une demande de plus forte valeur et l'insère à une position considérée comme optimale du point de vue temporel. Si le plan obtenu

n'est pas faisable temporellement, deux stratégies de réparation sont successivement utilisées, avec (1) une stratégie de réparation par réagencement des observations sélectionnées pour le satellite, et (2) une stratégie de réparation par retraits d'observations de plus faible valeur. D'autres mécanismes ont aussi été implémentés pour diversifier l'espace des plans considérés, notamment la métaheuristique ILS (*Iterated Local Search*) déjà utilisée dans la littérature pour des TDVRPTW [3] sans aspect sélectif ou dynamique.

### 3 Evaluations

Des expérimentations ont été réalisées sur un scénario CNES couvrant une période de deux semaines et impliquant plus de 80 000 demandes utilisateurs. Un simulateur a été construit pour modéliser le processus des arrivées successives de nouvelles demandes et le processus de replanification régulière toutes les 10 minutes sur la période de deux semaines. Sur l'ensemble de la simulation, cela conduit à environ 90 000 appels au planificateur élémentaire qui raisonne sur une composante connexe. Environ 100 variantes algorithmiques ont été testées sur plusieurs composantes connexes, avec différents temps de calcul maximum. Ces expérimentations ont permis d'extraire trois paramétrages (*very fast*, *fast* et *optimized*) qui correspondent à différents compromis entre temps de calcul et qualité de la solution produite. Sur une machine standard (Intel® Xeon® 3.6GHz), la simulation complète prend environ 2 minutes en mode *very fast* et 6 minutes en mode *fast*, mais est par contre plus longue en mode *optimized*. La Figure 1b montre un exemple de plan d'observation obtenu à l'issue de la simulation. La définition de méthodes exactes ou la comparaison avec des bornes a priori reste par ailleurs un point ouvert.

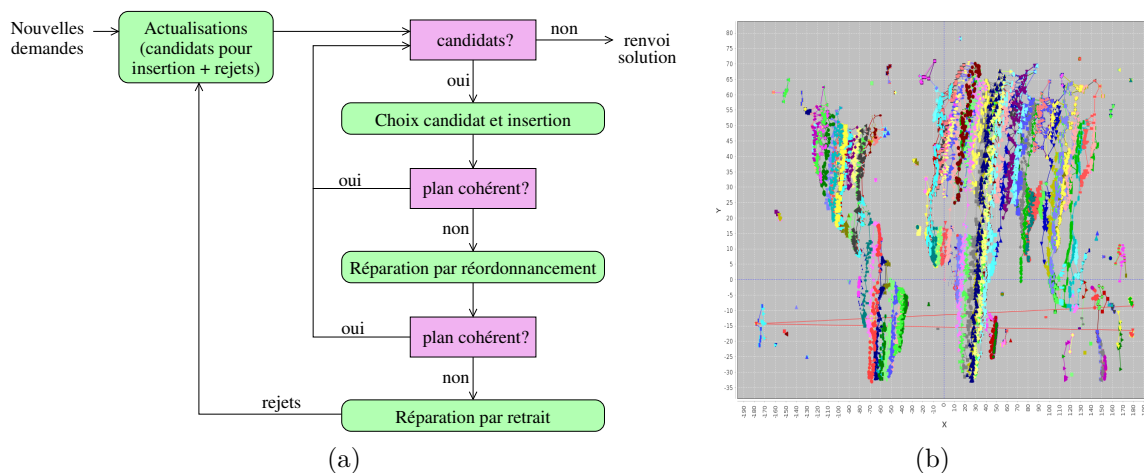


FIG. 1 – (a) Algorithme de recherche locale; (b) Exemple de plan (les points correspondent aux observations réalisées, les arêtes correspondent aux transitions entre observations, et les couleurs correspondent aux composantes connexes présentes sur les différentes révolutions du satellite)

### Références

- [1] A. Garcia, O. Arbelaitz, P. Vansteenwegen W. Souffriau, and M. T. Linaza. Hybrid approach for the public transportation time dependent orienteering problem with time windows. Proc. of the 5th International Conference on Hybrid Artificial Intelligence Systems (2010) 151–158
- [2] M. Lemaître, G. Verfaillie, F. Jouhaud, J. M. Lachiver, and N. Bataille. Selecting and scheduling observations of agile satellites. Aerospace Science and Technology **6** (2002) 367–381
- [3] H. Hashimoto, M. Yagiura, and T. Ibarakic. An iterated local search algorithm for the time-dependent vehicle routing problem with time windows. *Discrete Optimization*, 5(2) :434–456, 2008.