

Un nouveau modèle en programmation par contraintes de gestion temps réel des circulations ferroviaires basé sur le concept d'intervalles optionnels

Grégory Marlière¹, Sonia Sobieraj Richard¹, Paola Pellegrini², Joaquin Rodriguez¹

¹ Univ. Lille Nord de France, IFSTTAR, COSYS, ESTAS, F-59666 Villeneuve d'Ascq, France
{gregory.marliere,sonia.sobieraj,joaquin.rodriguez}@ifsttar.fr

² Univ. Lille Nord de France, IFSTTAR, COSYS, LEOST, F-59666 Villeneuve d'Ascq, France
paola.pellegrini@ifsttar.fr

Mots-clés : *recherche opérationnelle, programmation par contraintes, gestion temps-réel des circulations ferroviaires.*

1 Introduction

La conception des services ferroviaires est un processus complexe dans lequel la planification de l'horaire des trains est définie. Le retard d'un train peut engendrer des conflits au niveau opérationnel qui se propagent vers d'autres trains par effet domino. Pour limiter cette propagation, le régulateur en charge de la gestion du trafic peut agir sur l'ordre des trains dans les gares ou les jonctions et sur le choix des itinéraires. Le problème permettant de trouver un choix optimal parmi ces alternatives est nommé « real-time Railway Traffic Management Problem » (rtRTMP) [3].

Dans cet article, nous présentons les résultats expérimentaux d'une formulation du rtRTMP qui exploite une approche basée sur la programmation par contraintes (CP) comprenant ses dernières évolutions, comme les variables conditionnelles d'intervalles de temps introduites dans la bibliothèque Ilog CPOptimizer [1] ainsi que l'utilisation combinée de techniques de programmation linéaire [2] et d'algorithmes de recherche locale [4] dans les phases de recherche de solutions.

Cette nouvelle formulation nommée RECIFE-CPI a été validée à l'aide d'expérimentations sur un grand nombre d'instances et ses performances ont été comparées à celles d'un modèle état-de-l'art qui utilise la programmation linéaire mixte : RECIFE-MILP [3].

2 Formulation du rtRTMP

L'approche globale se place à un niveau microscopique dans lequel les mouvements élémentaires d'un train sur les sections de voie sont modélisés par une séquence d'intervalles temporels.

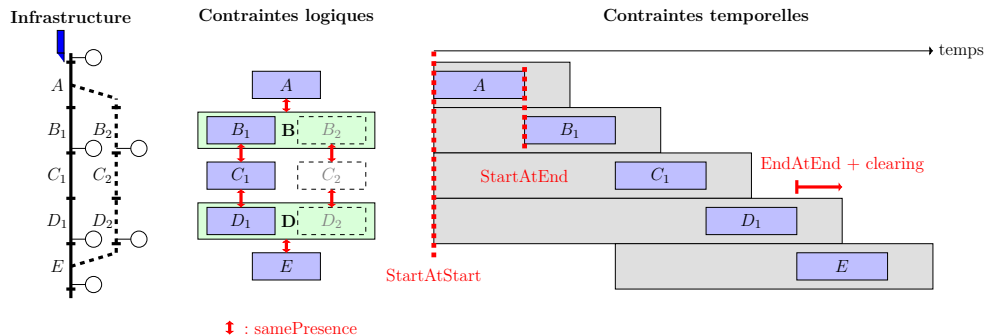


FIG. 1 – Contraintes logiques et temporelles entre intervalles : intervalles de type « Run » en bleu, intervalles de type « Res » en gris, et intervalles résultants d'un choix alternatif en vert. Rectangles pleins pour les intervalles présents dans la solution et pointillés pour les absents.

Le choix d'un itinéraire parmi les alternatives se traduit par un ensemble de choix logiques sur la présence d'intervalles optionnels. Des contraintes forçant le même état de présence entre plusieurs intervalles (samePresence) sont posées afin de garantir la cohérence de la solution. Dans l'exemple en Figure (1), l'intervalle B est déclaré comme résultante d'un choix entre l'utilisation des intervalles optionnels B_1 et B_2 . Dans ce cas, si A est présent (i.e, le train passe en A), B doit être présent, donc un choix alternatif doit être fait entre B_1 et B_2 . Si B_1 est choisi, C_1 doit être présent dans la solution, etc.

Du point de vue temporel, les intervalles de type « Run » modélisent les temps de déplacement de la tête du train sur les sections de voie, ils se succèdent donc directement grâce à des contraintes de type « StartAtEnd ». Leur durée minimale correspond au temps de passage en voie libre sur l'infrastructure. Cette durée peut être prolongée en cas de présence de retards dans la solution. Les intervalles de type « Res » servent à modéliser l'espace réservé par le train, leur début est synchronisé sur l'entrée physique dans une section de voie de référence dépendante de la signalisation (StartAtStart), leur fin est déterminée par le temps de dégagement du train (clearing time). Une contrainte de non recouvrement (NoOverlap) est posée entre tous les intervalles « Res » appartenant à la même section de voie afin d'assurer un espacement réaliste entre tous les trains passant sur l'infrastructure.

Ce modèle implémente également les contraintes liées aux durées et horaires d'arrêts en gare ainsi que celles liées à la réutilisation de matériel.

3 Expérimentations

Les expérimentations impliquent RECIFE-CPI et RECIFE-MILP [3] afin de comparer leurs performances pour quatre cas d'études : la jonction de Pierrefitte-Gonesse, une ligne entre Mantes-La-Jolie et Rouen ainsi que les gares de Lille-Flandres et St-Lazare. Les taux de perturbations (nombre et valeurs des retards à l'entrée des trains) sont variables afin de couvrir un large panel de difficultés. Les deux algorithmes sont initialisés avec une solution dite « timetable order » considérant uniquement les ordres et les itinéraires planifiés, puis ils disposent de 180 secondes de calcul pour améliorer cette solution. Le critère choisi est la minimisation de la somme des retards en gares et à la sortie de l'infrastructure.

4 Conclusion

Les résultats préliminaires de ces expérimentations montrent une bonne performance de l'approche proposée par rapport à l'algorithme RECIFE-MILP. Une analyse approfondie des faiblesses et des atouts de RECIFE-MILP et de RECIFE-CPI devrait permettre de proposer une approche hybride pour améliorer la résolution de ce type de problème.

Références

- [1] Philippe Laborie and Jérôme Rogerie. Reasoning with conditional time-intervals. In *Twenty-First International FLAIRS Conference*, 2008.
- [2] Philippe Laborie and Jérôme Rogerie. Temporal linear relaxation in ibm ilog cp optimizer. *Journal of Scheduling*, 4(19) :391–400, 2016.
- [3] P. Pellegrini, G. Marlière, and J. Rodriguez. Optimal train routing and scheduling for managing traffic perturbations in complex junctions. *Transportation Research Part B : Methodological*, 59 :58–80, 2014.
- [4] Petr Vilić, Philippe Laborie, and Paul Shaw. Failure-directed search for constraint-based scheduling. In *International Conference on AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, page 437–453. Springer, Cham, 2015.