



HAL
open science

Application de la programmation par contraintes à la gestion du trafic d'un noeud ferroviaire

Joaquin Rodriguez

► **To cite this version:**

Joaquin Rodriguez. Application de la programmation par contraintes à la gestion du trafic d'un noeud ferroviaire. 2007, 24p. hal-00545858

HAL Id: hal-00545858

<https://hal.science/hal-00545858>

Submitted on 13 Dec 2010

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.

Joaquin Rodriguez, (INRETS-ESTAS)

**APPLICATION DE LA PROGRAMMATION
PAR CONTRAINTES À LA GESTION DU
TRAFIC D'UN NŒUD FERROVIAIRE**

Juillet, 2007

Programme TAT - Projet T32 "Optimisation - Recife"

Arrêté du Conseil Régional N° 06 23 0009

INRETS/INRETS/RR - 07 - 717 - FR

Nom du projet : RECIFE (REcherche sur la Capacité des Infrastructures FERroviaires)

Partenaires : – INRETS, Unité de Recherche Évaluation des Systèmes de Transport Automatisés et de leur Sécurité,
– UVHC, LAMIH - UMR CNRS 8530 - Équipe Recherche Opérationnelle et Informatique
– SNCF, Délégation Régionale Infrastructure de Lille, Études prospectives Infrastructures, Pôle exploitation.

Intervenant : – J. Rodriguez (Directeur de Recherches, INRETS-ESTAS)

Type	Programme TAT - Projet T32 "Optimisation - Recife"
Convention	N° 02 31 0045
Décision	Arrêté du Conseil Régional N° 06 23 0009
Ref.	INRETS/RR - 07 - 717 - FR

Table des matières

1	Résumé	4
2	Introduction	4
2.1	La planification du trafic	4
2.2	La programmation par contraintes	5
2.3	La théorie de l'ordonnancement	6
3	Revue de la littérature	6
4	Contraintes techniques liées à la sécurité	10
4.1	Le principe d'espacement des trains	12
4.2	Les principes d'enclenchement d'itinéraires	12
5	Contraintes liées à l'exploitation	14
6	Approche par contraintes du problème de gestion opérationnelle du trafic ferroviaire	15
7	Méthode de résolution	18
7.1	Résolution d'un CSP	19
7.2	Résolution du modèle d'ordonnancement	19
8	Résultats	21
9	Conclusion	22

1 Résumé

Pour faire face à l'accroissement de la demande du trafic sans faire de nouveaux investissements, les gestionnaires d'infrastructure se penchent sur les possibilités d'améliorer l'exploitation de certains points critiques du réseau. Ces points critiques sont parcourus par des circulations mixtes (fret, grande ligne, TGV, ...) avec de nombreux types de conflits. Ces conflits sont difficiles à arbitrer par

Les opérateurs qui gèrent les circulations doivent prendre des décisions difficiles afin d'arbitrer ces conflits. La difficulté provient principalement des nombreuses contraintes qu'il faut considérer. Le travail présenté dans ce chapitre porte sur la résolution du problème d'optimisation de la gestion opérationnelle des circulations.

Le modèle qui est proposé pour traiter ce problème s'appuie sur la théorie de l'ordonnement et la programmation par contraintes. La présentation du modèle se focalise sur la justification des hypothèses de modélisation. Ces hypothèses sont comparées avec celles d'autres travaux qui utilisent la même approche. Des expérimentations sur des instances de problèmes à partir de données réels montrent les bonnes performances de ce modèle.

2 Introduction

2.1 La planification du trafic

Comme tous les modes de transport en commun, Le train est un mode de transport caractérisé par une forte composante de planification. Pour améliorer le transfert modale avec le mode routier, le transport ferroviaire doit adapter sa planification à diverses exigences changeantes et/ou antagonistes : satisfaction de la demande, service public, rentabilité, sécurité, sûreté, qualité de service, réduction des nuisances, ... La satisfaction de ces exigences place la planification comme une des activités clés de ce secteur.

Tout comme dans d'autres secteurs, la planification ferroviaire est souvent décomposée selon trois niveaux de décisions (c.f. tableau 1) : stratégique, tactique et opérationnel. On place principalement au niveau stratégique la configuration des ressources. À ce niveau, les décisions portent par exemple sur la construction de nouvelles infrastructures, l'engagement de personnels ou l'achat de matériels roulants.

Par la suite au niveau tactique, on trouve la conception du plan de transport qui consiste à répartir les ressources dans le temps et dans l'espace tout en remplissant au mieux les exigences citées plus haut. La conception des horaires des lignes est l'élément central du plan de transport.

Enfin, au niveau opérationnel se situe la mise en oeuvre du plan de transport. Les plans conçus en phase tactique ne sont jamais complètement appliqués en phase opérationnelle. Les aléas sur la disponibilité des ressources ou des modifications des demandes imposent des re-planifications de l'offre.

Au niveau d'une gare ou d'un noeud ferroviaire, les aléas et modifications sont fréquents et nombreux. Les aléas les plus courants que l'on peut citer sont les retards à l'arrivée ou au

Niveaux de décisions	Stratégique long terme	Tactique moyen terme	Opérationnel court terme
Horizon temporel	années	mois	heures, minutes
Objet	Infrastructures, Personnel, Matériel roulant, Offre de transport	Horaires, Roule- ments du personnel, et du matériel	Gestion des dépôts, Circulations des lignes, des gares et des bifurcations

TAB. 1 – Niveaux de décisions de la planification ferroviaire

départ, les trains supplémentaires, les pannes de certains composants de la signalisation, les travaux de maintenance sur la voie, . . . Pour les clients, ces aléas se traduisent par des retards par rapport aux horaires prévus. Dans les situations où le trafic est dense et à la limite de la capacité, l'ampleur de la répercussion des aléas sur les retards peut être très importante. Par exemple sur un noeud comme celui de Pierrefite Gonesse situé au nord de Paris (cf. fig8, section 8), un retard de quelques secondes à l'arrivée d'un train peut se convertir en un retard de cinq minutes.

Dans ces situations, pour limiter ce phénomène d'amplification des retards, une solution consiste à augmenter la capacité en ajoutant des aiguillages, des voies, des sauts-de-mouton ou des quais supplémentaires. Mais, par manque d'espace disponible ou pour les coûts qu'elles entraînent, ces solutions ne sont pas envisageables lorsque l'on est situé dans des zones urbaines. Par conséquent, dans le contexte d'un trafic dense sur un noeud, il est donc impératif d'optimiser le partage de l'infrastructure.

Au niveau opérationnel, le routage et l'ordre des trains devront être reconsidérés afin d'exploiter toutes les alternatives d'itinéraires et d'ordonnement des trains. L'opérateur qui gère le trafic durant les heures de pointes doit rapidement prendre ses décisions dans un contexte de stress parmi une multitude d'alternatives. Pour l'aider dans cette tâche, un outil qui lui propose les solutions les plus intéressantes peut se montrer très bénéfique. Dans ce but, nous avons développé un modèle du problème d'optimisation du routage et de l'ordonnement des trains dans une infrastructure ferroviaire comme un noeud ou une gare. Ce modèle est caractérisé par l'utilisation de deux approches : la programmation par contraintes et la théorie de l'ordonnement. Nous allons brièvement présenter ces deux approches.

2.2 La programmation par contraintes

La programmation par contraintes est une technique de représentation de problèmes de satisfaction de contraintes. En toute généralité, un problème de satisfaction de contraintes (CSP) est défini par un triplet $\mathcal{P} = (X, D, C)$ tel que :

- $X = \{x_1, \dots, x_m\}$ est un ensemble de variables,

- $D = \{d_1, \dots, d_m\}$ est un ensemble de domaines finis, un domaine d_i est l'ensemble des valeurs possibles associé à la variable x_i ,
- $C = \{c_1, \dots, c_e\}$ est un ensemble de contraintes ou chaque contrainte c_i limite les combinaisons de valeurs possibles pour un sous-ensemble de variables de X .

Pour une instance de CSP, on peut chercher une solution, toutes les solutions ou une solution optimale vis-à-vis d'une fonction objectif. Dans ce dernier cas, un CSP associé à une fonction objectif est un problème d'optimisation combinatoire. La programmation par contraintes (CP) regroupe les langages et outils qui permettent de poser un CSP et de programmer des algorithmes de résolution. Ces outils l'utilisent un processus qui s'appuie sur les contraintes pour réduire l'espace de recherche et ainsi diminuer l'effort de calcul nécessaire à la résolution du problème. Ce processus appelé propagation de contraintes permet d'une part de valider les choix de valeurs pour certaines variables, mais surtout de supprimer dans les domaines des valeurs ne pouvant plus faire partie d'une solution compte tenu des choix opérés. C'est cette suppression de valeurs dans les domaines qui permet de converger vers une solution du problème ou au contraire vers la démonstration d'absence de solution.

2.3 La théorie de l'ordonnancement

La théorie de l'ordonnancement définit un cadre formel et des algorithmes qui visent à placer dans le temps des tâches, appelées aussi activités. La réalisation des activités nécessite des ressources soumises à des contraintes d'utilisation ou de capacité. Les variables temporelles qui décrivent l'exécution des activités sont reliées par deux types de contraintes.

Le premier type de contraintes relie directement les activités entre elles selon des principes qui sont spécifiques à chaque application. Par exemple dans une application d'ordonnancement de projets lorsque qu'une tâche nécessite la réalisation préalable d'une autre tâche, on pose la contrainte que la date de début de la première tâche doit être supérieure à la date de fin de la seconde tâche.

Le second type de contraintes sont celles liées à l'utilisation et au partage des ressources par les activités. Parmi les ressources, on distingue les ressources consommables et les ressources renouvelables, ces dernières pouvant être à capacité limitée ou à état. Par exemple, le budget d'un projet est une ressource consommable et les machines qui permettent de réaliser certaines tâches sont des ressources renouvelables. À travers le partage des ressources, c'est la satisfaction des contraintes de capacité ou d'état qui engendre des relations indirectes entre les variables temporelles des activités.

Dans la section 3 qui suit, nous allons analyser les travaux de recherche qui se rapprochent du problème traité et des deux approches utilisées pour notre modèle.

3 Revue de la littérature

Pour une revue complète des travaux sur les modèles de planification du trafic ferroviaire nous renvoyons le lecteur aux articles de Cordeau et al. [4] et de Bussieck et al. [1]

. Ces revues couvrent tous les problèmes de planification ferroviaire : planification du personnel, du matériel, triage des wagons, formation des trains, élaboration des horaires, . . .

Une revue plus récente réalisée par Törnquist [35] se focalise sur les problèmes de gestion du trafic et analyse les caractéristiques des problèmes traités, les modèles proposés et le niveau d'application de ces modèles. On remarque grâce à cette revue qu'un grand nombre de modèles s'appuient sur la programmation mathématique ou la théorie des graphes. Un second enseignement est l'insuffisance de travaux comparatifs entre les différentes approches. Cette insuffisance est notamment due à l'absence de benchmarks dont les données soient accessibles publiquement. Dans la présentation qui suit, nous nous focaliserons d'abord sur les travaux qui utilisent la programmation par contraintes pour la gestion du trafic puis sur ceux qui en plus s'inspirent de la théorie de l'ordonnancement.

Une des premières contributions à proposer la programmation par contraintes pour modéliser un problème de gestion du trafic ferroviaire est, à notre connaissance, celle de Fukumori [7, 8]. Le problème traité consiste à élaborer un horaire à partir des spécifications d'un ensemble de contraintes sur les trains. Les variables de décision portent sur les dépassements ou les croisements des trains dans les gares. Par conséquent, les principales variables du modèle sont les dates de départ et d'arrivée aux gares. Le critère à optimiser est la durée des parcours de chaque train afin de limiter les sur-stationnements dans les gares. Les contraintes du modèle traduisent :

- (Ct. 1) les durées de succession minimales que doivent respecter les trains dans les tronçons de voie qui relient les gares,
- (Ct. 2) les durées minimales pour les phases d'arrêt, de croisement et de dépassement dans les gares,
- (Ct. 3) Le nombre maximum de trains que peuvent recevoir chaque gare.

L'algorithme de résolution est de type DFS (Depth First Search), il réalise des arbitrages sur les conflits d'utilisation des tronçons à partir de priorités entre trains. L'ordre d'instanciation des variables suit l'ordre de parcours des trains. Lors de la réalisation de cette recherche, les outils de CP n'étaient pas encore très répandus, un algorithme de propagation de contraintes a donc été spécifiquement développé. Ce dernier réalise une consistance sur les bornes des domaines des variables, cette consistance est semblable à l'arc-B-consistance [21].

Le modèle de Fukumori a été validé sur des instances de problèmes d'une journée d'exploitation des lignes d'Osaka et de Nara [8]. Les exemples de l'article portent sur des instances de 14 gares et 40 trains.

Par la suite, plusieurs auteurs ont traité le problème d'élaboration d'horaires en s'inspirant du modèle de Fukumori. Le modèle CAPRES d'Hachemane [10], reprend les mêmes contraintes en les complétant avec des contraintes spécifiques aux applications traitées. L'extension la plus notable concerne les gares qui peuvent être modélisées avec une précision plus grande. Par exemple, le modèle peut inclure les incompatibilités entre itinéraires d'entrée et de sortie des gares. Mais cette possibilité n'est utilisée que pour prouver la faisabilité d'une solution. L'algorithme de consistance des contraintes est du même type que celui de Fukumori et son implémentation n'utilise pas non plus de bibliothèque CP. Le mo-

dèle CAPRES a été utilisé dans de nombreuses études de capacité (voir un bilan dans [26]), il a été intégré à la suite logicielle Viriato [32].

Le modèle du projet PRaCoSy décrit par Chiu et al. [3] reprend aussi les mêmes contraintes définies par Fukumori. L'application est différente, il s'agit d'assister un opérateur à la re-planification d'un horaire pour faire face à des perturbations. Deux nouveaux critères sont alors considérés :

- Minimiser les modifications de l'horaire théorique,
- Minimiser le plus grand retard par rapport à l'horaire théorique.

Pour la résolution, deux heuristiques sont testées. La première utilise un ordre d'instanciation basée à la plus petite valeur du domaine des variables. La seconde utilise un ordre d'instanciation qui préserve au maximum les valeurs définies par l'horaire initial. De cette façon, les variables des trains en conflit et celles des trains qui ont une partie d'itinéraire en commun avec les trains en conflit sont les dernières instanciées. La seconde heuristique a donné de meilleurs résultats sur les instances de problème de la ligne Nanjingxi-Shanghai. La taille de l'implémentation de cette instance avec la bibliothèque CP Ilog Solver donne un problème à 598 variables et 3005 contraintes. En revanche, l'article ne donne pas d'indicateurs sur la difficulté des problèmes traités (e.g. le nombre de trains en conflit).

Ingolitto et al. [14] font aussi référence au modèle de Fukumori et aux CSP. Comme dans CAPRES, il s'agit de traiter un problème de saturation d'horaires. La résolution utilise une heuristique qui partage la ligne en deux afin de définir un arbitrage d'ordonnement entre les trains saturants. Il faut noter que l'algorithme réalise uniquement la vérification des contraintes d'utilisation des voies uniques. Si la vérification échoue, les trains sont décalés.

Isaai et Singh [15] proposent un modèle de type Fukumori pour la résolution d'un problème d'élaboration d'horaires. Leur méthode de résolution combine une heuristique CP pour obtenir une solution réalisable et une heuristique tabu pour améliorer cette solution dans un voisinage. Les 5 jeux de données utilisés pour les tests sont issus de deux lignes, avec des voies uniques et doubles. Ces lignes comportent une cinquantaine de gares. Le nombre de conflits est estimé entre 25 et 62. Les résultats de l'article montrent que la méthode tabu améliore la solution initiale trouvée par l'heuristique CP.

Les variables de décision des travaux qui précèdent ont trait principalement à l'ordonnement des trains sur les ressources partagées (quais des gares, sections de voies). L'introduction de variables supplémentaires sur le routage des trains donc sur le choix des ressources a aussi été abordée avec l'approche CP. L'ajout de ces variables augmente singulièrement la taille de l'espace de recherche. Cette augmentation rend très difficile la recherche d'une solution admissible et qui plus est une solution optimale.

À notre connaissance, Gosselin dans [9] décrit la première application qui, traite à la fois du problème d'ordonnement et du routage de trains par l'approche CP. Compte tenu du nombre de variables et de contraintes, le problème à résoudre se limitait à la recherche d'une solution admissible. Pour atteindre ce but, une contribution essentielle est la définition de contraintes qui réalisent des propagations entre les dates des événements d'entrée-sortie de gares et les alternatives d'itinéraires. Une fonction d'évaluation des itinéraires guide la méthode de résolution et lui permet ainsi de trouver plus rapidement de bonnes solutions.

Afin de limiter l'espace de recherche, le nombre d'itinéraires possibles entre les gares est limité à 4. Le modèle a été appliqué sur 10 instances du réseau péri-urbain de Melbourne, la taille des instances allant de 200 à 2000 trains. Les temps de résolution étaient inférieurs à une heure ce qui donnait satisfaction à la demande des clients.

Poujade et al. [25] ont aussi traité un problème de routage avec un modèle CP. L'application est celle de l'allocation des voies à quai pour la Gare du Nord de Paris. Dans cette application, les horaires des trains sur les points de référence sont fixés. La difficulté du problème réside dans le nombre d'itinéraires, la quantité et le type des contraintes à satisfaire. Le problème initial de l'application est sur-contraints, c-à-dire qu'il n'y a pas de solution réalisable. Les contraintes sont alors divisées en trois catégories : celles qu'il faut satisfaire absolument, celles qui sont aménageables et celles qu'il faut respecter dans la mesure du possible. Dans la méthode de résolution, le nombre d'itinéraires possibles pour un train étant si important (plusieurs milliers), ces itinéraires ne sont calculés qu'après l'affectation des voies à quai (terminus et/ou arrêt intermédiaires). Puis, le choix des itinéraires se dirige vers ceux qui sont aussi directs que possible. En cas d'échec, de nouvelles affectations des voies à quais sont tentées. S'il y a échec à la fin de la recherche, les ensembles de trains en conflits sont identifiés afin de permettre à l'utilisateur de choisir les contraintes à relâcher pour relancer une nouvelle recherche. L'application est opérationnelle et elle est utilisée quotidiennement. La zone des itinéraires couvre un rayon de 16 km autour la gare du Nord, 10 gares, 100 quais pour 1700 trains avec un temps de résolution de 15 minutes.

Enfin, la théorie de l'ordonnancement est aussi à l'origine de nombreux travaux de recherche pour traiter les problèmes d'élaboration d'horaires ou de routage. Cette idée a été développée aussi bien avec des modèles CP qu'avec des modèles en programmation mathématique.

C'est avec cette dernière approche que Spzigel propose pour la première fois dans [33] l'utilisation de la théorie de l'ordonnancement pour modéliser la gestion du trafic ferroviaire. Le problème abordé est ici encore l'élaboration d'un horaire, mais sur une ligne où tous les tronçons sont à voie unique. L'idée centrale est de considérer que le problème a une structure analogue à celle d'un problème d'ordonnancement d'atelier ("job-shop scheduling"). Pour cela, les itinéraires des trains sont décomposés en plusieurs segments de voie. Une activité est associée au parcours de chaque segment de voie. Ainsi, chaque train est une séquence ordonnée d'activités sur le modèle d'un job. Une propriété majeure de ce type de modèle tient à l'expression implicite des incompatibilités entre les parcours des trains par la contrainte de capacité des segments de voie. En effet, les segments de voie sont des ressources sur lesquelles une seule activité peut être exécutée en même temps, c'est la description des segments de voie utilisés par les trains qui détermine les incompatibilités entre les trains. Pour les gares, le modèle de Spzigel suppose qu'elles peuvent recevoir un nombre de trains quelconque, c-à-dire que la capacité est infinie. La fonction objectif est la somme pondérée des durées des parcours qui est équivalente à la somme pondérée des dates d'entrée dans le dernier segment de chaque train. La méthode de résolution utilise un B&B, les choix de valeurs se font en priorité sur les contraintes disjonctives entre les trains en conflits sur un segment de voie. Les instances traitées portent sur une ligne de 5

segments de voie, les trains circulent pour moitié dans chaque sens. La durée de parcours des segments est la même, le nombre de trains varie de 4 à 10.

Le premier modèle CP qui utilise la même idée que Spzigel (mais sans y faire référence) est celui de Lamma et al. [19]. Le problème visé est celui de la gestion globale du trafic sur des gares et des lignes. Le modèle définit chaque itinéraire d'une gare comme une ressource alors que dans une ligne, chaque canton est une ressource. Une contribution importante de cette recherche est l'utilisation d'un module en charge de la gestion de la disponibilité des ressources. Ce module vient en complément d'un autre module qui est en charge des décisions d'affectation des ressources et d'ordonnement des activités. Bien que le modèle vise des applications de gestion de réseaux, les expérimentations qui sont référencées portent uniquement sur la gestion du trafic d'une gare, mais ne donnent pas d'indications sur la taille et la difficulté des instances.

Kreuger et al. dans [17] propose un modèle CP qui utilise aussi la théorie de l'ordonnement. Dans ce modèle, les activités sont les parcours sur les tronçons entre les gares. On retrouve les mêmes types de contraintes que dans le modèle de Fukumori. Pour la résolution, deux méthodes ont été expérimentées. La première méthode énumère simplement les valeurs des domaines des variables de départ des gares. Elle s'est tout de même révélée plus performante que la seconde méthode basée sur l'ordonnement des activités de chaque ressource. Les instances étudiées et publiées par la suite dans [18] comportent 2 500 à 3 000 activités, correspondant à 200 trains. Le nombre de variables varie de 9 500 à 13 000 pour 17 500 à 30 000 contraintes, mais une mesure de la difficulté des instances n'est pas fournie.

Enfin, Oliveira [24] a traité le problème d'ordonnement des trains d'une ligne à voie unique avec un modèle CP basé sur l'idée de Spiezel. La modélisation est plus fine que celle proposée par Kreuger et al. car les parcours sont décomposés en cantons. Comme dans le modèle de Lamma et al. [19], les ressources sont les cantons. Plusieurs techniques de résolution ont été proposées, on retiendra celle basée sur la modification des ordres des activités situées sur les chemins critique d'une solution.

La suite de la présentation s'organise ainsi : dans la section suivante nous présenterons succinctement quelques éléments de la signalisation et de l'exploitation ferroviaire. Ces éléments permettront de mieux comprendre leur rôle dans la définition des contraintes du problème développée dans la partie 6. La section 7 décrit la méthode de résolution du problème qui est utilisée. Après la présentation des résultats obtenus, la dernière partie dresse les conclusions de ce travail et donne d'autres perspectives de recherche.

4 Contraintes techniques liées à la sécurité

En phase opérationnelle de l'exploitation ferroviaire, la satisfaction des exigences de protection contre les risques est répartie entre des opérateurs, la signalisation, le mécanicien et le train comme l'illustre le schéma fonctionnel de la figure 1. Bien que ce schéma ne soit qu'une vue simplifiée, on remarque que la signalisation ferroviaire a un rôle central dans la gestion du trafic. En effet, c'est elle qui impose une grande partie des contraintes temporelles dans les séquences de circulations. La signalisation est un intermédiaire dans la

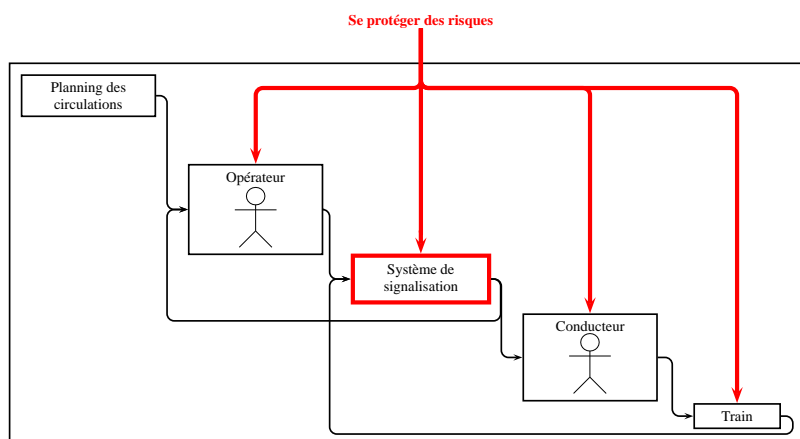


FIG. 1 – Gestion des circulations

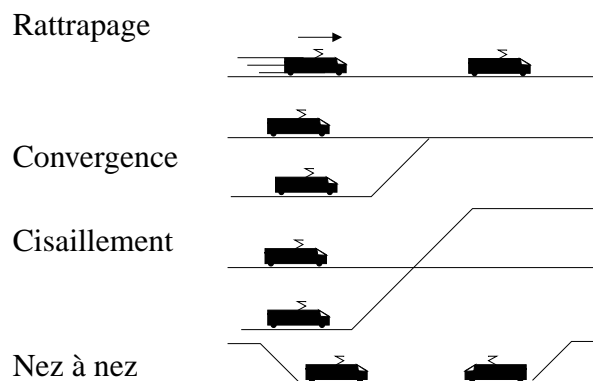


FIG. 2 – Exemples de risques de collisions ferroviaires

transmission et la réception des informations de localisation et d'autorisation de mouvement entre les opérateurs, le conducteur et le train.

Les exemples présentés par la suite s'appuient sur les principes du système de signalisation du réseau français [31], ils n'en restent pas moins représentatifs des systèmes mis en place dans les autres réseaux du monde car ils utilisent des principes similaires.

La signalisation ferroviaire a pour but de se protéger contre les risques de collision ou de déraillement. La figure 4 illustre quelques situations de risques de collisions. Pour réaliser cet objectif deux types de principes sont appliqués : le *principe d'espacement des trains* et les *principes d'enclenchement des itinéraires*.

4.1 Le principe d'espacement des trains

Le principe consiste à garder en toutes circonstances les trains circulant dans un même sens à des distances supérieures à la distance d'arrêt. L'application de ce principe est réalisée par un dispositif, appelé «block system», mis en place dès le début du chemin de fer. Ce système se décompose schématiquement en trois fonctions :

1. Localiser les trains,
2. Diviser la voie en cantons,
3. À l'entrée de chaque canton, donner des consignes de vitesse qui vérifient en toutes circonstances le principe d'espacement.

La localisation des trains est réalisée au moyen d'un dispositif électrique appelé «circuit de voie» (c.f. figure 3) en disposé tout au long de la voie. Ce dispositif fournit une information booléenne qui indique la présence ou non d'un train.

La voie est ensuite décomposée en *cantons* dont les limites sont matérialisées par des *panneaux* qui fournissent les consignes de conduite aux mécaniciens. La longueur des cantons peut varier suivant les vitesses maximales autorisées, la configuration de la voie ou les capacités de freinage des trains. Par exemple, la longueur d'un canton peut être de 500 m à l'entrée d'une gare ou de 2,5 km sur une ligne entre deux gares. Il faut souligner qu'un canton peut contenir un ou plusieurs circuits de voie.

Enfin, les types de consigne à l'entrée des cantons sont appelés *aspects* de signalisation. Le nombre d'aspects qui permettent de passer de la vitesse maximale à l'arrêt est une caractéristique importante de la ligne. La figure 3 illustre le cas d'un système de signalisation à trois aspects : voie libre (VL), avertissement (A), sémaphore (S). L'aspect (VL) indique au conducteur qu'il peut entrer dans le canton à la vitesse maximale et cela jusqu'à la fin du canton. L'aspect (A) indique qu'il peut entrer dans le canton mais qu'il devra être en mesure de pouvoir d'arrêter à la fin du canton. L'aspect (S) indique qu'il ne peut pas entrer dans le canton. À travers cet exemple, on voit que l'intervalle minimum de passage de deux trains avec l'indication de vitesse maximale correspond à la durée de la traversée de deux cantons. Cet intervalle peut être réduit en augmentant le nombre d'aspects et en diminuant la longueur des cantons. La relation entre ces trois paramètres est :

$$\text{Intervalle} = \frac{(\text{Nombre d'aspects} - 1) \times \text{longueur des cantons}}{\text{vitesse maximale}}$$

Le coût de chaque aspect supplémentaire augmente pour une réduction de l'intervalle chaque fois inférieure. On remarque aussi que le long d'un parcours, même peu étendu, l'espacement peut être contrôlé par un nombre d'aspects différents.

4.2 Les principes d'enclenchement d'itinéraires

Les principes d'enclenchement visent à imposer des conditions aux manoeuvres des appareils de voie (e.g. un aiguillage) et aux signaux de protections. Ces conditions empêchent

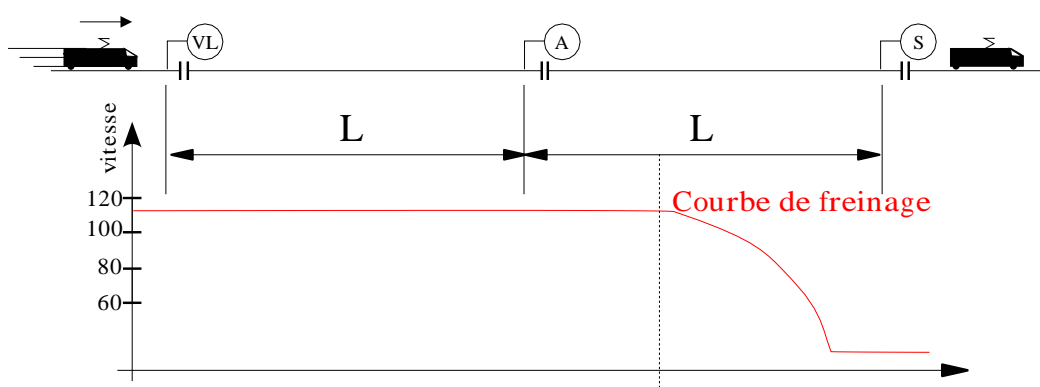


FIG. 3 – Réalisation du principe d'espacement

des manoeuvres qui aboutiraient à des situations dangereuses comme la convergence, le cisaillement ou le nez à nez.

Le nombre et la complexité des situations concernées ne peuvent être repris ici, nous nous limiterons à un cas illustré par la figure 4. Dans cet exemple, on s'intéresse au passage des circulations sur une voie banalisée (i.e. ayant deux sens de circulation possibles). Le train 1 doit réaliser le parcours qui mène en A et le train 2 doit réaliser le parcours mène en B. Si le train 1 doit passer avant le train 2, l'aiguilleur commence par *former* l'itinéraire du train 1, c'est-à-dire qu'il positionne les deux aiguillages de l'itinéraire menant en A ainsi que les signaux correspondants en voie libre. Un signal carré (C) protège l'aiguillage du circuit de voie z06 et empêche le train 2 d'accéder à la partie de voie banalisée. On dit que la position de l'aiguille enclenche le carré (à l'ouverture), c'est-à-dire qu'il est impossible de mettre ce signal en voie libre au moins tant que l'aiguille reste dans cette position. Par la suite, le signal pourra passer en voie libre lorsque le train 1 aura libéré z06. La formation des itinéraires et les enclenchements imposent une séquence d'événements aux circulations. Si cette séquence d'événements ne correspond pas à celle prévue lors de la conception des horaires, alors des retards peuvent apparaître.

Les deux séquences de passage des trains avec l'indication de vitesse maximale se traduisent par les conditions théoriques sur les événements décrites dans tableau 2. Il faut souligner que l'intervalle entre les trains est ici différent de l'intervalle défini dans le paragraphe précédent avec le principe d'espacement. Cet intervalle est lié à l'occupation et la libération de certains circuits de voie. Les expressions du tableau 2 ne sont valides que pour une signalisation à trois aspects, elles peuvent être aisément généralisées pour un nombre d'aspects quelconque. De même, il est possible de les affiner afin de tenir compte des distances de visibilité des signaux.

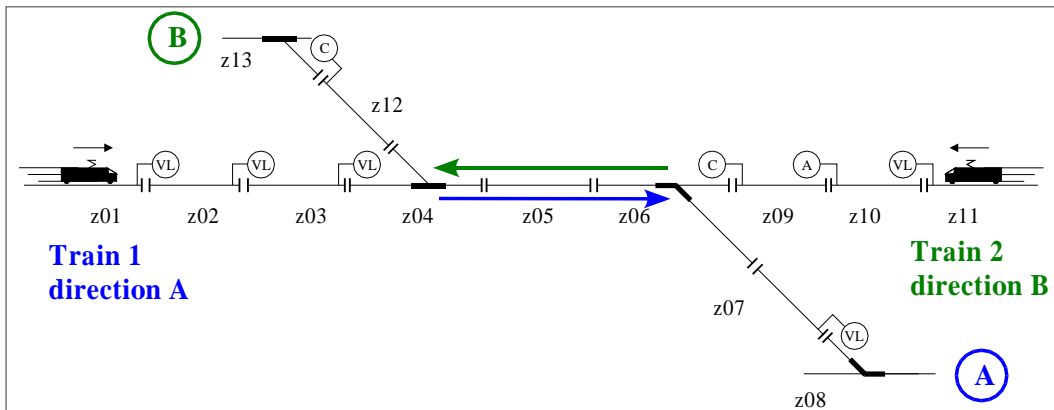


FIG. 4 – Circulations sur une voie unique

Séquences d'itinéraires	Séquences d'événements
Train1 \prec Train2	Train2.(occupation de z10) $>$ Train1.(libération de z06)
Train1 \succ Train2	Train1.(occupation de z02) $>$ Train2.(libération de z04)

TAB. 2 – Séquences d'événements pour des parcours d'itinéraires sans retard

5 Contraintes liées à l'exploitation

Dans la section précédente, nous avons présenté les contraintes de sécurité qui proviennent du partage de l'usage de la voie. Le partage d'autres types de ressources engendre des contraintes qui sont aussi incluses dans les horaires. Ce sont ces contraintes qui font l'objet de cette section.

La circulation d'un train nécessite la présence d'un conducteur. En général, les roulements des conducteurs sont conçus après les horaires des trains. Mais lors de la mise en oeuvre opérationnelle, plusieurs aléas peuvent perturber la disponibilité des conducteurs. Par exemple, si le retard d'un train dépasse l'intervalle de pause prévu pour le conducteur pour reprendre le train qui suit dans son roulement, ce dernier va subir un retard. Des précautions devront être prises dans l'optimisation des roulements afin de ne pas dégrader au final la stabilité des horaires. Cette information sur les roulements devra être disponible et contrôlée pour prendre les décisions durant l'exploitation opérationnelle.

La gestion des ressources matérielles comme les locomotives et les wagons a aussi des implications dans les horaires. Les pannes de ces matériels conduisent parfois à de fortes perturbations des horaires, car le recours à d'autres matériels est long à mettre en oeuvre. De plus, l'information concernant le couplage entre les horaires de trains liés par des ressources matérielles est cruciale. Par exemple, un changement du quai d'arrivée d'un train peut avoir un impact direct sur le quai de départ du prochain train dans le roulement. En effet, pour éviter des mouvements du matériel de deux trains d'un roulement qui se suivent sur un délai

court, on impose que les deux trains partent du même quai. Dans ce genre de situation, le choix du quai d'arrivée par l'opérateur devra impérativement tenir compte du roulement du matériel utilisé.

Les quais sont une autre ressource dont le partage va influencer sur les horaires. Les trains ne peuvent pas embarquer ou débarquer leurs passagers dans n'importe quel quai. Un quai peut être réservé uniquement une catégorie de trains. Par exemple, il est fréquent que des quais soient réservés exclusivement aux trains de banlieue ou aux trains internationaux. La longueur d'un quai par rapport à la longueur du train peut aussi être une contrainte pour l'utilisation du quai.

Enfin, pour simplifier les horaires offerts aux passagers et améliorer l'offre, certains réseaux proposent des horaires cadencés. Ce type d'offre va donc imposer des contraintes fortes dans la conception et la réalisation des horaires.

Les correspondances entre trains participent aussi à la qualité de l'offre d'un réseau. Pour assurer des correspondances, il y aura deux types de contraintes à satisfaire. Les premières sont des contraintes temporelles entre les dates d'arrivée et de départ aux quais des deux trains. La seconde concerne les quais des deux trains qui devront être soit en vis à vis soit ne pas être trop éloignés dans la gare afin de faciliter le transbordement des passagers.

6 Approche par contraintes du problème de gestion opérationnelle du trafic ferroviaire

Dans cette section nous allons décrire un modèle gestion opérationnelle du trafic ferroviaire qui combine les concepts d'ordonnancement d'ateliers et les techniques CP.

Ce modèle, que nous allons appeler M_{CBS} (pour Constraint-Based Scheduling), doit s'appliquer à la gestion de noeuds ferroviaires et de gares. Dans ce type d'application, les longueurs et les durées des parcours sont beaucoup plus courtes que pour des applications de gestion de lignes. Cette singularité nous a conduit à inclure dans le modèle les caractéristiques techniques du système de signalisation décrites dans la section 4.

Dans la revue de la littérature (section 3), deux principes de modélisation se détachent pour modéliser les conflits d'utilisation de segments de la voie :

- poser directement les contraintes d'incompatibilité entre trains [3, 7, 10, 14, 15, 17].
- poser les contraintes d'utilisation des segments de la voie et des contraintes de capacité de ces segments [9, 19, 24, 33].

La seconde approche a été retenue car elle présente deux avantages. Le premier est d'éviter de poser toutes les combinaisons de contraintes entre chaque train ayant une partie d'itinéraire commune. Le second avantage est qu'elle permet de faire correspondre les segments de la voie partagés à des éléments du système de signalisation. Cela facilite l'interprétation des résultats et les actions sur le système ferroviaire. De plus, l'instanciation du modèle peut être réalisé à partir de la description du plan de voie et des paramètres de sa signalisation. Il est alors possible de raisonner plus globalement sur le partage d'un segment de voie entre un ensemble de trains plutôt qu'entre paires de trains en conflit.

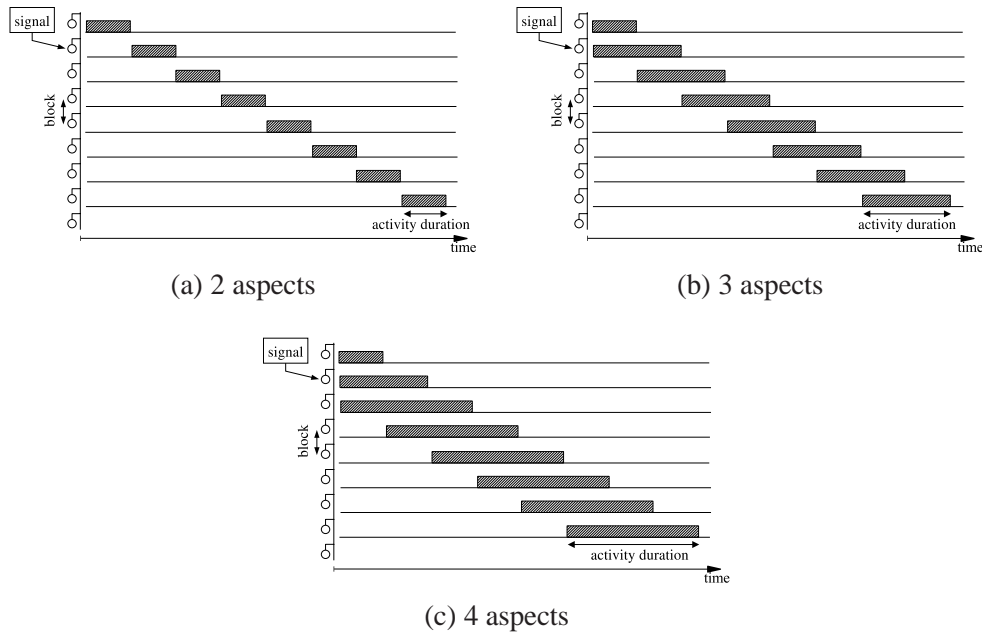


FIG. 5 – Modèle d’une ressource par canton : Diagrammes de Gantt selon le nombre d’aspects de signalisation.

La définition des limites de segments de voie correspond à la définition des ressources partagées entre les trains. Ici encore deux approches se dégagent. Dans [9, 33] les segments de voie sont les tronçons qui relient deux gares. Dans [19, 24], les segments de voie sont plus précis et correspondent aux cantons de la voie.

Dans la seconde approche, on gagne en précision dans la définition des ressources, mais, aucune publication ne précise l’influence du nombre d’aspects de signalisation. Or les relations temporelles entre activités et leurs durées sont très différentes selon le nombre d’aspects de la signalisation. La figure 5 montre les diagrammes de Gantt d’un train avec différents nombre d’aspects. Lorsque le nombre d’aspects n de la signalisation augmente d’une unité, la durée des activités augmentent de la durée de parcours d’un canton. L’ajout d’un aspect de signalisation se traduit par l’ajout d’un canton d’espacement entre les trains. En conclusion, pour une signalisation à n aspects avec $n > 2$, les activités doivent se chevaucher sur $(n - 2)$ fois la durée de parcours d’un canton.

Dans le contexte de notre application (c.f. [27, 28]), le canton n’est pas suffisamment précis pour représenter correctement les incompatibilités. C’est le circuit de voie qui convient le mieux pour définir la ressource à partir de laquelle les incompatibilités entre les horaires des trains sont déterminées. Dans l’exemple de la figure 1, le tableau 2 donne les conditions de compatibilité des horaires des trains 1 et 2 en fonction d’événements de libération ou d’occupation de circuits de voie. Cet exemple illustre le bien fondé de ce choix de modélisation car pour avoir sa voie libre, le train 2 doit attendre que le train 1 libère le circuit de voie $z6$ et non qu’il libère le canton (i.e. $z7$). Nous avons utilisé les limites

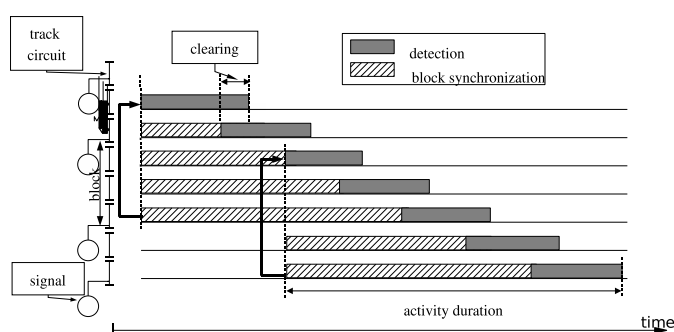


FIG. 6 – Modèle d’une ressource par circuit de voie : diagramme de Gantt pour 3 d’aspects de signalisation.

des circuits de voie pour définir les limites des ressources partageables, mais pour être plus général, il est possible de considérer n’importe quel dispositif qui fournit l’information de localisation au système de signalisation. Dans ces conditions, une pédale peut par exemple constituer aussi une limite de ressource.

Comme cela a été précisé dans la section 4, un canton peut comporter plusieurs circuits de voie. Dans ce cas, il faut synchroniser le début de chaque activité d’un canton. En revanche, la fin de chaque activité se fera au fur et à mesure des libérations des circuits de voie (c.f. figure 6).

Pour définir cette synchronisation, nous avons décomposé chaque activité en deux phases. Une première phase correspond à la réservation de la ressource pour satisfaire la contrainte d’espacement. La durée de cette phase est fonction du nombre d’aspects. Une seconde phase correspond à la détection du train par le circuit de voie. La synchronisation des dates de début des activités d’un canton pour un canton d’indice i , se fait sur la date de début de la phase de détection du canton $i - (n - 2)$ (c.f. figure 6).

Ces contraintes de recouvrement et de synchronisation permettent de décrire correctement toutes les relations temporelles qui découlent des principes d’espacement et d’enclenchement décrits dans la section 4. Le lecteur intéressé par la formulation de ce modèle peut se référer à l’article [29].

Le modèle qui vient d’être exposé permet aussi d’exprimer facilement les contraintes d’exploitation décrites dans la section 5. Ces contraintes se formulent à partir des variables temporelles du modèle car elles représentent tous les événements essentiels d’un parcours. De même, elles peuvent aussi se formuler à partir des variables des ressources utilisées. Par exemple, pour exprimer une contrainte d’équilibre entre deux trains, on posera une contrainte d’égalité entre la ressource de la dernière activité d’un train et la ressource de la première activité du second train. Pour faire une correspondance entre deux trains, on posera une contrainte de précédence entre les variables de début et de fin d’activités correspondant aux arrivées et aux départs des trains à quai.

Deux extensions ont été apportées à ce modèle. La première vient palier à une hypothèse simplificatrice sur le comportement des trains durant les perturbations. En effet, on

suppose qu'après la libération des ressources en conflit, les trains circulent immédiatement à la vitesse maximum. Cette hypothèse est plausible dans l'absolu si le mécanicien a les moyens d'anticiper la date du passage en voie libre. Cette anticipation lui permettrait d'accélérer plus tôt et, ainsi, d'atteindre la vitesse maximum juste au moment du passage en voie libre. Les dispositifs actuels de contrôle-commande des circulations ne permettent pas ce type de conduite, et sont, de plus, contraire à la sécurité pour certains conflits (e.g. conflits de convergence ou de cisaillement). Pour modéliser complètement les phases d'accélération/décélération, il est possible de recourir à un nombre important de simulations comme dans Tariel et Gauyacq [34]. Une autre solution consiste à étendre le modèle présenté en incluant des contraintes qui allongent la durée des activités situées dans le voisinage de la ressource indisponible. Cette solution a été implémentée et a montré des résultats très proches de ceux obtenus par simulation ([29]). En contre partie, la modélisation des phases d'accélération/décélération augmente les temps de calcul. De plus, les expérimentations ont montré que les solutions obtenues sont souvent les mêmes que celles obtenues sans cette extension, il convient donc de bien étudier les situations pour lesquelles le recours à ce type d'extension s'impose.

La seconde extension a pour but d'améliorer les méthodes de résolution en ajoutant des contraintes redondantes. Les contraintes redondantes permettent de synthétiser des sous-ensembles de contraintes élémentaires et ainsi aboutir à des consistances plus fortes. On rappelle qu'une consistance plus forte conduit à réduire encore plus l'espace de recherche. Mais pour que ces contraintes soient intéressantes, il faut que les temps de propagation ne viennent pas compenser le gain issu de la réduction du nombre de noeuds explorés.

Les contraintes redondantes qui ont été introduites portent sur les conflits de contre-sens dans des séquences de circuits de voie. Dans la figure 4, la séquence de circuits de voie $\langle z_4, z_5, z_6 \rangle$ est un exemple de conflit de contre-sens. Pour chaque conflit de contre-sens on définit une ressource à deux états. Cette ressource va permettre de modéliser le sens de circulation sur cette partie de la voie. On définit aussi une activité qui recouvre la séquence des activités d'un train sur le conflit de contre-sens. Si un train doit circuler sur un conflit de contre-sens, l'activité de couverture requiert la ressource dans un état correspondant à son sens de circulation. La contrainte de couverture et la contrainte d'état de la ressource permettent de propager des décisions d'ordonnancement sur des activités du conflit de contre-sens (voir figure 7). On trouvera dans [30], une première formulation de cette extension. Par la suite, le modèle avec cette extension sera noté M_{CBS}^+ .

7 Méthode de résolution

Dans cette section nous allons dans un premier temps faire une brève présentation des méthodes de résolution d'un modèle CSP. Puis, nous exposerons les méthodes de résolution qui ont été appliquées au modèle du problème d'ordonnancement de trains présenté dans la section précédente.

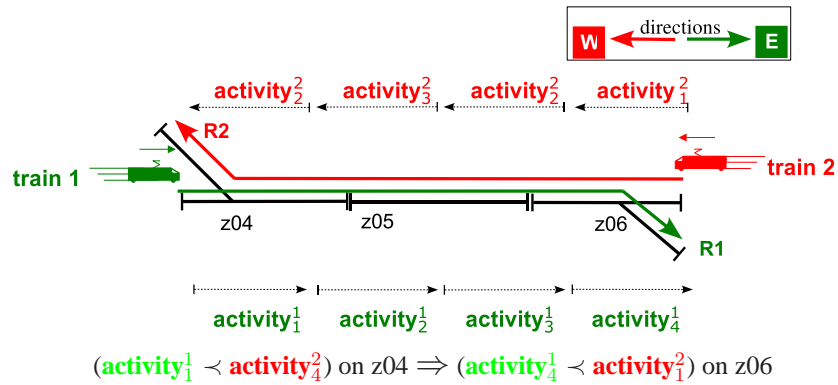


FIG. 7 – Exemple de propagation de décision d’ordonnement sur un conflit de contresens

7.1 Résolution d’un CSP

La méthode de résolution d’un CSP est un arbre de recherche où deux procédures sont effectuées à chaque noeud :

- Une procédure d’instanciation des variables
- Une procédure de consistance (ou de propagation) des contraintes.

Durant la procédure d’instanciation, une valeur du domaine d’une variable est sélectionnée. Le domaine courant de la variable est réduit à la valeur sélectionnée. Cette réduction de domaine engendre d’autres réductions de domaines à travers la procédure de consistance des contraintes. Lorsque la propagation engendre un domaine de variable vide, cela signifie qu’il n’existe pas de solution dans le sous-arbre. Dans ce cas, une nouvelle valeur est essayée sur une instanciation précédente. Une solution réalisable est trouvée lorsque toutes les variables sont instanciées.

Une procédure de consistance supprime les valeurs qui sont inconsistantes avec les contraintes. Plusieurs types de consistance ont été définis, e.g. la consistance d’arc [22], la chemin-consistance [23], la k-consistance [6], l’arc-B-consistance [21].

Dans le cas d’un CSP avec une fonction objectif, la méthode de résolution utilise un algorithme de type branch and bound. Pour un problème de minimisation d’un critère c , chaque fois qu’une solution est trouvée avec une valeur c^* pour le critère, on ajoute au modèle la contrainte $c < c^*$ pour tous les autres noeuds de l’arbre de recherche.

Les performances d’une méthode de résolution d’un CSP dépendent de trois paramètres essentiels : l’ordre de choix des variables et des valeurs d’instanciation, la ou les procédures de consistances et la technique de backtracking.

7.2 Résolution du modèle d’ordonnement

La résolution de notre problème consiste à définir quand les activités doivent être réalisées et avec quelles ressources. C’est donc à la fois un problème d’ordonnement d’acti-

vités et un problème d'affectation de ressources.

L'approche la plus répandue est d'abord de résoudre le problème d'affectation des ressources, puis de résoudre le problème d'ordonnement.

Pour les expérimentations présentées ici, le problème d'affectation des ressources est résolue avec une procédure d'énumération complète des itinéraires des trains. La méthode utilise une ordre statique basé sur la taille des domaines.

Le problème d'ordonnement est résolue par une méthode exacte issue de la library Ilog Scheduler [13]. L'algorithme utilise les étapes suivantes :

1. Obtenir rapidement une première solution par une heuristique avec laquelle on définit une borne inférieure au critère.
2. Sélectionner une ressource parmi les ressources requises par les activités non-ordonnées.
3. Sélectionner une activité à ordonner en premier parmi les activités qui requièrent la ressource choisie. Poser les contraintes de précédente correspondantes. Garder les autres activités comme alternatives à essayer en cas d'échec.
4. Itérer l'étape 3 jusqu'à ce que toutes les activités qui requièrent la ressource choisie soient ordonnées.
5. Itérer les étapes 2 à 4 jusqu'à ce que toutes les activités qui requièrent une ressource communes soient ordonnées.

L'heuristique de l'étape 1 consiste à appliquer une règle d'ordonnement de type First Come First Served (FCFS).

À l'étape 2, la ressource sélectionnée est celle dont la "slack" est minimale. La slack est la différence entre la disponibilité (ou "l'offre") d'une ressource et la demande de cette ressource sur un intervalle de temps (c.f. [13]). La slack équivaut à une mesure de la criticité d'une ressource. À l'étape 3, l'activité à ordonner en premier est celle ayant la plus petite valeur de début au plus tôt, et, en cas d'égalité, celle ayant la plus petite valeur de début au plus tard.

Pour ce qui concerne les procédures de consistance, en plus de l'arc-consistance, trois mécanismes ont été utilisés pour propager l'utilisation des ressources et ajuster les limites des activités [20] :

- Le premier mécanisme repose sur une structure de données appelée "timetable". Cette structure explicite l'utilisation d'une ressource et sa disponibilité dans le temps.
- Le second mécanisme est la "propagation de contraintes disjonctives", il consiste à maintenir une arc-B-consistance sur les contraintes temporelles (c.f. [21]),
- Le troisième mécanisme est connu sous le nom de "edge finding" (c.f. [2]), il considère un ensemble d'activités $\Omega \cup A_i$ qui requière la même ressource et détermine si A_i doit être placée avant ou après toutes les activités de Ω .

Avec ces trois mécanismes, il est possible, en ajustant les limites des activités, d'éviter l'exploration d'un grand nombre de noeuds qui mènent à un échec.

Le backtracking consiste à choisir un nouveau noeud à explorer. Ce choix est fait soit après avoir trouvé une solution admissible soit après un échec. Pour tous les résultats qui

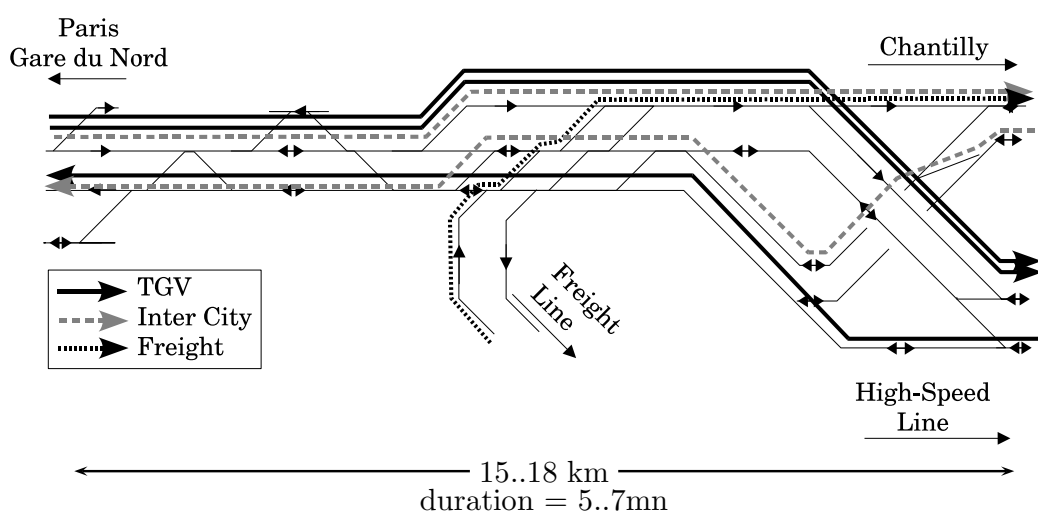


FIG. 8 – The instance problem #1 at the Pierrefitte-Gonesse junction

sont présentés ici, nous avons utilisé uniquement le backtracking chronologique : le nouveau noeud à explorer est celui de la dernière variable instanciée ; si toutes les valeurs du domaine ont été testées, on revient sur la précédente variable.

8 Résultats

Dans cette section nous allons présenter quelques résultats obtenus avec le modèle et la méthode de résolution présentés auparavant.

Le jeu de données utilisés pour les expérimentations est issu de l'exploitation du noeud de Pierrefitte Gonesse (voir figure 8) où circule un trafic mixte : TGV, trains régionaux et trains de marchandise. Ce trafic mixte rend les décisions de routage et d'ordonnement plus complexes.

L'ensemble des instances de problèmes sur lequel nous avons travaillé se base sur un scénario réel qui est illustré dans la figure 8. Cette instance de problème est constituée de six trains : 3 TGV, 2 trains régionaux et 1 train de marchandise. Le cisaillement des voies par les trains de marchandise provoquent des conflits avec tous les autres trains, qui eux mêmes génèrent d'autres conflits.

Le tableau 3 montrent les caractéristiques de l'ensemble des instances. Neuf instances supplémentaires ont été définies à partir de cette instance réelle #1. Pour constituer les instances #2 à #10, deux trains ont été successivement ajoutés à l'instance #1 en gardant la même proportion de type de conflit. Les colonnes intitulées M_{CBS} se réfèrent au modèle de base de la section 6, et les colonnes M_{CBS}^+ au modèle étendu avec les contraintes de conflits de contre-sens. Ces chiffres montrent qu'il y a à peu près autant de variables que de contraintes. Ils montrent aussi que l'extension M_{CBS}^+ augmente le nombre de variables et

de contraintes de 30 % et 40 % respectivement.

Les résultats sur l'ensemble des instances sont résumés dans le tableau 4. La colonne intitulée "first sol." indique la durée totale des retards pour la première solution trouvée, elle correspond au scénario appliqué sans modification du routage par défaut. Les colonnes "delays" indiquent les retards obtenus avec la meilleure solution. Les colonnes "gap" indiquent la proportion d'amélioration des retards par rapport la première solution.

La durée des scénarios varient entre 6 minutes et 35 minutes. La condition d'arrêt a été fixé à la moitié de la plus petite durée des scénarios, soit 180 secondes. On observe que le modèle M_{CBS} permet une amélioration importante des retards, puisqu'il permet une réduction des retards de 76 % à 10 % par rapport aux retards de la première solution.

Le modèle M_{CBS}^+ n'améliore pas notablement la qualité des solutions puisque l'amélioration du gap n'est en moyenne que de 2.2 %. Seule l'instance #5 avec 14 trains améliore de manière plus significative le critère (10.8 %).

Afin de mieux apprécier la qualité des solutions obtenues, nous avons recherché les valeurs optimales des critères en allongeant le temps de traitement (jusqu'à 15 heures). Il en résulte que 4 instances (#1 à #4) ont les valeurs optimales. Les autres instances ont des critères situés entre 16 % et 40 % de l'optimum. Cet écart s'explique par la taille de l'espace de recherche de ces instances. Pour les instances #5 à #10, il a entre 1×10^{10} et 1×10^{16} combinaisons de routages qu'il faut multiplier par les combinaisons d'ordonnancement pour résoudre les conflits.

En revanche, le modèle M_{CBS}^+ permet de réduire de 63.7 % en moyenne le temps CPU pour les instances qui ont le même critère (instances #2,#3,#4,#9,#10). Pour les autres instances, comme les valeurs du critère sont différentes, ce temps ne peut être comparé. Malgré tout, si l'on excepte l'instance #7, les temps de traitement du modèle M_{CBS}^+ sont inférieur à ceux de M_{CBS} .

Ces modèles d'optimisation a été intégrés à différent outils de traitements et de visualisation des résultats. Par exemple, la figure montre l'outil de visualisation d'un diagramme de Gantt avec la solution optimale pour l'instance #1. Ce type d'outil permet d'analyser finement les solutions et d'identifier les ressources critiques d'un parcours.

9 Conclusion

La programmation par contraintes est un outil de modélisation et de résolution qui a montré ses preuves pour la résolution de nombreux problèmes combinatoires. Nous avons montré comment cette approche de modélisation peut être utilisée pour traiter le problème de routage et d'ordonnancement de trains.

Le modèle présenté est capable de prendre en compte de nombreuses caractéristiques techniques et commerciales issues du terrain.

Des expérimentation sur des problèmes de taille croissante ont montré qu'il est possible d'obtenir des solutions de bonne qualité avec des temps de traitement conformes aux contraintes opérationnelles.

Des perspectives d'amélioration de la méthode de résolution portent sur :

TAB. 3 – The set of problem instances studied

Inst.	# trains	Train categories			M_{CBS}		M_{CBS}^+	
		TGV	IC	Freight	# var.	# const.	# var.	# const.
1	6	3	2	1	2773	2687	3596	4054
2	8	3	3	2	3548	3320	4568	4987
3	10	4	4	2	4554	4131	5811	6134
4	12	4	5	3	5329	4764	6783	7067
5	14	5	6	3	6331	5572	7773	7891
6	16	5	7	4	7110	6208	8998	9147
7	18	6	8	4	8116	7019	10241	10294
8	20	6	9	5	8891	7652	11213	11227
9	22	7	10	5	9897	8463	12456	12374
10	24	8	10	6	10672	9096	13428	13307

 TAB. 4 – Résultats avec les modèles M_{CBS} et M_{CBS}^+

Inst.	first sol. ^a	M_{CBS}			M_{CBS}^+		
		delays ^a	% gap ^b	CPU time ^c	delays ^a	% gap ^b	CPU time ^c
1	156	64	58.97	0.64	59	62.18	0.56
2	269	64	76.21	0.71	64	76.21	0.51
3	304	161	47.04	15.8	161	47.04	8.42
4	417	161	61.39	24.11	161	61.39	12.2
5	452	350	22.57	140.25	301	33.41	136.27
6	565	352	37.7	121.56	313	44.6	106.11
7	600	500	16.67	109.85	498	17	160.38
8	713	512	28.19	170.52	512	28.19	92.41
9	748	660	11.76	47.01	649	13.24	170.34
10	861	713	17.19	85.87	713	17.19	77.76

^aIn seconds

^bRelative gap of delays in %, i.e., $100 \times (\text{first sol.} - \text{delays}) / \text{first sol.}$

^cIn seconds using a 1.66 GHz Intel® Duo Core™ T2300 processor, Ilog Solver 5.3 and Ilog Scheduler 5.2

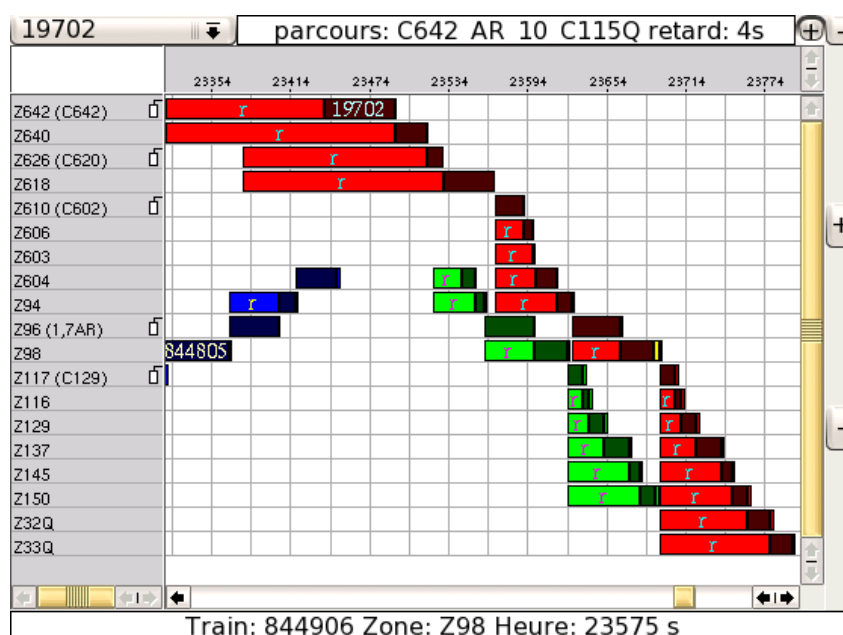


FIG. 9 – Outil de visualisation du diagramme de Gantt d’une solution

- L’évaluation de techniques de décomposition du problème afin de mieux résoudre les problèmes de grande taille,
- L’étude de l’apport d’heuristiques de recherche comme LDS (Limited Discrepancy Search [11]) ou DDS (Depth-bounded Discrepancy Search [36]) qui ont montré de bons résultats pour certains CSP.

Comme l’a indiqué Tornquist [35], de nombreux modèles qui utilisent des approches ont aussi montré des résultats très intéressants sur des problèmes similaires (e.g. [5]). L’absence de données sur des instances de problèmes accessibles publiquement ne permettent pas de faire des comparaisons sur les avantages et les inconvénients des différentes approches. Ce type de comparaisons pourraient déboucher des modèles hybrides qui intègrent certains algorithmes de Recherche Opérationnelle ou de théorie des graphes dans un modèle CP. Les résultats publiés sur l’utilisation d’approches hybrides pour des problèmes d’ordonnement dans d’autres secteurs (voir [12, 16]) montrent que c’est une voie de recherche très prometteuse.

Remerciements

Ces travaux ont bénéficié du soutien financier du Conseil Régional Nord-Pas-de-Calais et des fonds FEDER de l’Union Européenne.

Références

- [1] M.R. Bussieck, T. Winter, and U.T. Zimmermann, *Discrete optimization in public rail transport*, Mathematical Programming **79** (1997), 415–444.
- [2] J. Carlier and E. Pinson, "Adjustment of heads and tails for job-shop problem", European Journal of Operational Research (1994), no. 78, 146–161.
- [3] C. K. Chiu, C. M. Chou, J. H. M. Lee, Ho-Fung Leung, and Y. W. Leung, *A constraint-based interactive train rescheduling tool*, Second International Conference on Principles and Practice of Constraint Programming, 1996, pp. 104–118.
- [4] J-F. Cordeau, P. Toth, and D. Vigo, *A survey of optimization models for train routing and scheduling*, Transportation Science **32** (1998), no. 4, 380–404.
- [5] Andrea D’Ariano and Marco Pranzo, *An advanced real-time train dispatching system for minimizing the propagation of delays in a dispatching area under severe disturbances*, 2nd International Seminar on Railway Operations Modelling and Analysis (Hannover, Germany), March 2007.
- [6] E.C. Freuder, "A sufficient condition for backtrack-free search", Journal of the ACM **29** (1982), no. 1, 24–32.
- [7] Koji Fukumori, *Fundamental scheme for train scheduling : Application of range-constriction search*, Tech. Report A.I. Memo No. 596, Massachusetts Institute of Technology Artificial Intelligence Laboratory, USA, 1980.
- [8] Koji Fukumori and Hiroshi Sano, *Fundamental algorithm for train scheduling based on artificial intelligence*, Systems and Computers in Japan **18** (1987), no. 3, 52–64.
- [9] Vincent Gosselin, *Train scheduling using constraint programming techniques*, Charme Technical Bulletin (1993), no. 3, 3–13.
- [10] Patrick Hachemane, *Évaluation de la capacité de réseaux ferroviaires*, Thèse 1632, École Polytechnique Fédérale de Lausanne, 1997, (in french).
- [11] W.D. Harvey and M.L. Ginsberg, *Limited discrepancy search*, Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95); Vol. 1 (Montréal, Québec, Canada) (C. S. Mellish, ed.), Morgan Kaufmann, 1995, August 20-25 1995, pp. 607–615.
- [12] John N. Hooker, *An integrated method for planning and scheduling to minimize tardiness.*, Constraints **11** (2006), no. 2-3, 139–157.
- [13] Ilog, *Ilog scheduler 5.2 reference manual*, 2002.
- [14] L. Ingolitto, F. Barber, P. Tormos, A. Lova, M. Salido, and M. Abril, *An efficient method to schedule new trains on a heavily loaded railway network*, Advances in Artificial Intelligence, IBERAMIA 2004, 9th Ibero-American Conference on AI (Puebla, Mexico) (C.A. Reyes and J.A. Gonzales, eds.), Lecture Notes in Computer Science, Springer Verlag, November 2004, pp. 164–173.

- [15] M.T. Isaai and M.G. Singh, *An intelligent constraint-based search method for a single-line passenger-train scheduling problems*, The International Conference on the Practical Application of Constraint Technologies and Logic Programming, 2000, pp. 79–91.
- [16] Vipul Jain and Ignacio E. Grossmann, *Algorithms for Hybrid MILP/CP Models for a Class of Optimization Problems*, *Inform Journal on Computing* **13** (2001), no. 4, 258–276.
- [17] P. Kreuger, M. Carlson, J. Olsson, T. Sjöland, and Emil Aström, *Trips scheduling on single track networks - the tuff train scheduler*, CP'97 Workshop on Industrial Constraint - Directed Scheduling, 1997, pp. 1–12.
- [18] Per Kreuger, Mats Carlsson, Thomas Sjöland, and Emil Aström, *Sequence dependent task extensions for trip scheduling*, Tech. Report SICS-T-2001/14-SE, Swedish Institute of Computer Science (SICS), May 2001.
- [19] E. Lamma, P. Mello, M. Milano, and A. Torreggiani, *A constraint logic programming approach to the train scheduling problem*, Expert systems 94, 14 th annual conference of the British Computer Society Specialist Group on Expert systems, Application and Innovations in Expert Systems II (Cambridge, UK), December 1994, pp. 293–306.
- [20] C. Le Pape, *Three mechanisms for managing resource constraints in a library for constraint based scheduling*, INRIA/IEEE Conference on Emerging Technologies and Factory Automation (Paris-France), 1995.
- [21] O. Lhomme, *"Consistency techniques for numeric CSPs"*, 13th International Joint Conference on Artificial Intelligence (Chambéry, France), 1993, pp. 232–238.
- [22] A. Mackworth, *"Consistency in network of relations"*, *Artificial Intelligence* **8** (1977), 99–118.
- [23] H. Montanari, *"Network of constraints : Fundamental properties and applications to pictures processing"*, *Information Sciences* **7** (1974), 95–132.
- [24] Elias Silva Oliveira, *Solving single-track railway scheduling problem using constraint programming*, Ph.D. thesis, School of Computing - University of Leeds, 2001.
- [25] M. Poujade, P. Moyon, and Puget J.F., *Sagitaire : Allocation des quais de la gare Paris-Nord*, Conférence IA'95, 1995, (in french).
- [26] Robert Rivier, Luigi Lucchini, and Anne Curchod, *Analysing the capacity of railway networks : Summing up the experience*, 9 th World Conference on Transportation Research, July 2001.
- [27] J. Rodriguez and L. Kermad, *Constraint programming for real-time train circulation management problem in railway nodes*, Computers in Railways VI (Lisbonne-Portugal), September 1998, pp. 597–606.
- [28] Joaquín Rodriguez, *Technique d'aide à la fluidification des circulations dans un noeud ferroviaire complexe - résultats obtenus et analyses*, Rapport Contrat de Recherche SNCF N° 86-96, INRETS, December 1998, In French.

- [29] Joaquín Rodríguez, *A constraint programming model for real-time train scheduling at junctions*, *Transportation Research Part B : Methodological* **41** (2007), no. 1, 231–245.
- [30] Joaquín Rodríguez, *A study of the use of state resources in a constraint-based model for routing and scheduling trains*, 2nd International Seminar on Railway Operations Modelling and Analysis (Hannover, Germany), March 2007.
- [31] R. Réteveau, *La signalisation ferroviaire*, Presse de l'École Nationale des Ponts et Chaussées, 1987, In French.
- [32] SMA und Partner AG, *Future module : CAPRES - Elaboration And Saturation Of Railway Network*, Viriato|OpenTrack User-Conference (Zurich, Suisse), May 2005.
- [33] Bernado Spzigel, *Optimal train scheduling on a single track railway*, OR'72 (M. (ed.) Ross, ed.), North Holland Publishing Co., 1973, pp. 343–352.
- [34] Pascal Tariel and Daniel Gauyacq, *SAFIR : A decision support system to optimise railway traffic through complex network junctions*, World Conference on Railway Research (Köln), November 2001.
- [35] J. Törnquist, *Computer-based decision support for railway traffic scheduling and dispatching : A review of models and algorithms*, ATMOS2005 (Algorithmic MeThods and Models for Optimization of RailwayS) (Palma de Mallorca, Spain), Dagstuhl Research Online Publication Server (DROPS), October 2005.
- [36] T. Walsh, *Depth-bounded discrepancy search*, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), vol. 2, Août 1997, pp. 1388–1393.