

L'agrégation des contraintes de ressources dans un problème de plus court chemin

Anass Nagih

*GERAD et École des Hautes Études Commerciales de Montréal
3000, Chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7
anass@crt.umontreal.ca*

François Soumis

*GERAD et École Polytechnique de Montréal
Département de Mathématiques et Génie Industriel
C.P. 6079, Succ. Centre-Ville
Montréal, Québec, Canada H3C 3A7
soumis@crt.umontreal.ca*

février, 1999

Les Cahiers du GERAD

G-99-02

Résumé

Le problème de plus court chemin avec contraintes de ressources consiste à trouver un chemin d'un point origine à un point destination de coût minimum et respectant les contraintes sur les consommations de ressources. La complexité d'un algorithme de type programmation dynamique augmente avec le nombre de ressources. Afin d'obtenir rapidement une bonne solution heuristique, nous proposons la réduction de l'espace des états par agrégation des ressources. Il s'agit de projeter n ressources sur un vecteur de dimension inférieure, puis d'ajuster dynamiquement la matrice de projection donnant la meilleure approximation de la solution optimale. Le but de ce rapport est de présenter un travail préliminaire pour améliorer la compréhension de ce problème. On se propose de formuler mathématiquement la procédure de sélection des étiquettes efficaces pour mieux comprendre ce que représente la dominance par rapport à un sous-ensemble de ressources, et d'étudier ainsi les propriétés des solutions induites. Afin de contourner les difficultés liées aux approches directes de construction de solutions primales réalisables, nous proposons une alternative basée sur les relaxations lagrangienne et "surrogate" combinée à un algorithme de réoptimisation pour construire des solutions primales réalisables.

Abstract

The shortest path problem with resource constraints consists in finding the minimum cost path between two specified points while respecting constraints on the resource consumptions. The complexity of a dynamic programming algorithm increases with the number of resources. To rapidly obtain a good heuristic solution we propose to reduce the state space by aggregating resources. Our approach consists in projecting n resources on a vector with a smaller dimension and then to dynamically adjust the projection matrix which gives a better approximation of the optimal solution. This report presents a preliminary work performed to improve the understanding of the problem. We propose to mathematically formulate the process of selecting efficient labels in order to better understand the dominance process in a subset of resources. To avoid the difficulties related to direct approaches used to construct feasible solutions, we propose an alternative based on the Lagrangean relaxation and surrogate relaxation combined with a reoptimization algorithm to construct feasible solutions.

1 Introduction

Soit $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ un graphe orienté avec \mathcal{A} l'ensemble des arcs (i, j) et $\mathcal{V} = \mathcal{N} \cup \{o, d\}$ l'ensemble des noeuds où \mathcal{N} est l'ensemble des noeuds i qui peuvent être visités entre l'origine o et la destination d . Soit \mathcal{R} l'ensemble des ressources, avec $|\mathcal{R}| = n$. À chaque noeud $i \in \mathcal{V}$, est associé n intervalles de contraintes (fenêtres de ressources) $[a_i^r, b_i^r]$, $r \in \mathcal{R}$, restreignant les quantités des ressources utilisées pour atteindre le noeud i . Un arc $(i, j) \in \mathcal{A}$, consomme une quantité t_{ij}^r de chaque ressource $r \in \mathcal{R}$ et a un coût c_{ij} . Tout arc ne permettant pas de visiter le noeud j après le noeud i , c'est-à-dire ne respectant pas les fenêtres de ressources :

$$a_i^r + t_{ij}^r \leq b_j^r, \quad \forall r \in \mathcal{R}$$

est supprimé. On associe à un chemin P_i , de l'origine o au noeud i , un coût C_i somme des coûts des arcs le composant, et un n -vecteur d'état T_i correspondant aux consommations cumulées T_i^r de chaque ressource $r \in \mathcal{R}$. Un état T_i est dit réalisable ou légal si :

$$a_i^r \leq T_i^r \leq b_i^r, \quad \forall r \in \mathcal{R}.$$

Dans le cas d'une seule ressource, l'indice r peut être omis. Un chemin P_i est dit réalisable s'il correspond à un état réalisable. Le problème de plus court chemin avec contraintes de ressources consiste à trouver un chemin réalisable (légal) de coût minimum allant de l'origine o à la destination d . Ce problème a été introduit par Desrochers (1986) comme une généralisation du problème de plus court chemin avec fenêtres de temps. Il apparaît comme un sous problème lors de la fabrication d'horaires de travail ou de routes de véhicules.

Plusieurs synthèses des applications et méthodes de résolution du problème de plus court chemin avec contraintes de ressources ou ses variantes ont été décrites dans la littérature (Minoux 1975, Handler et Zang 1980, Aneja et al. 1983, Desrosiers et al. 1983, Jaffe 1984, Desrochers 1986, Desrochers et Soumis 1988a,b, Maculan et Salles 1991, Desrosiers et al. 1995, Desaulniers et Villeneuve 1997).

Soit $P_i^{(k)}$ le k ème chemin de l'origine o au noeud i . On lui associe une étiquette $(T_i^{(k)}, C_i^{(k)}) = (T_i^{1(k)}, T_i^{2(k)}, \dots, T_i^{n(k)}, C_i^{(k)})$ représentant son état des ressources et son coût.

Définition 1 Soient $(T^{(1)}, C^{(1)})$ et $(T^{(2)}, C^{(2)})$ deux étiquettes associées à deux chemins réalisables de o à i . On dit que $(T^{(1)}, C^{(1)})$ domine $(T^{(2)}, C^{(2)})$ et on note $(T^{(1)}, C^{(1)}) \preceq (T^{(2)}, C^{(2)})$ si et seulement si

$$C^{(1)} \leq C^{(2)} \quad \text{et} \quad T^{r(1)} \leq T^{r(2)}, \quad \forall r \in \mathcal{R}.$$

Définition 2 Une étiquette associée à un chemin réalisable de o à i , est dite efficace s'il n'existe pas d'étiquette en i qui la domine. Un chemin est dit efficace s'il est associé à une étiquette efficace.

Pour résoudre le problème de plus court chemin avec contraintes de ressources, Desrochers et Soumis (1988a) proposent un algorithme de programmation dynamique. Cet algorithme procède en trois grandes étapes. En chaque noeud $j \in \mathcal{N}$, il effectue les opérations suivantes :

1. prolongation du chemin (génération des étiquettes),
2. filtrage (test de réalisabilité),
3. dominance (élimination des étiquettes non efficaces).

La relation \preceq étant une relation d'ordre partiel, le nombre d'étiquettes efficaces à traiter augmente d'une façon exponentielle en fonction du nombre de ressources. Ce qui rend la procédure de prolongation très ardue. La taille de l'espace de recherche peut être réduite, en augmentant l'élimination de la procédure de dominance. Cela peut se faire par exemple, en réduisant la dimension de l'espace des ressources, ou bien en définissant de nouvelles règles de dominance ou de nouvelles fonctions de prolongation.

Une première alternative consiste à projeter l'espace des étiquettes, de dimension $n + 1$, sur un espace de dimension plus petite, puis appliquer les règles de dominance dans cet espace de dimension réduite pour définir les étiquettes efficaces.

Soient $\Pi = (\Pi_{pq})$ une matrice réelle $m \times (n + 1)$, avec $m \leq n$, et T' le nouveau vecteur de ressources tel que

$$\begin{cases} T'^1 &= \Pi_{11}T^1 + \dots + \Pi_{1n}T^n + \Pi_{1(n+1)}C \\ \vdots & \\ T'^m &= \Pi_{m1}T^1 + \dots + \Pi_{mn}T^n + \Pi_{m(n+1)}C \end{cases} \quad (1.1)$$

La matrice de projection Π peut être globale ou locale à un noeud, ou peut encore dépendre des arcs. La sélection des étiquettes efficaces basée, par exemple, uniquement sur le coût et ne tenant pas compte des autres composantes (T^1, \dots, T^n) (c'est à dire, $m = 1$, $\Pi_{1j} = 0$, $\forall j \in \{1, \dots, n\}$ et $\Pi_{1(n+1)} = 1$) permet effectivement de réduire le nombre d'étiquettes traitées en chaque noeud. Cependant, on risque d'éliminer en cours de traitement des chemins optimaux. Ainsi, malgré que la réalisabilité par rapport à toutes les ressources soit maintenue, l'optimalité n'est plus garantie. Une première amélioration consiste à définir, pour chaque noeud j , une matrice de projection locale $\Pi(j)$, qui dépend du noeud traité. Une seconde amélioration peut être apportée, en considérant une matrice de projection $\Pi(i, j)$ dont les coefficients dépendent des arcs (i, j) arrivant au noeud traité j . En d'autres termes, elle dépend du noeud traité et de ses prédécesseurs. Cependant, le nombre de coefficients à ajuster sera plus important.

Le but de ce rapport est de présenter un travail préliminaire pour améliorer la compréhension de ce problème.

Dans la littérature, l'étude des solutions efficaces et la réduction de l'espace des états ont fait l'objet de recherches surtout dans le cadre de la programmation multi-objective et également pour améliorer l'efficacité de certains algorithmes de programmation dynamique (Christofides et al. 1981, White 1982, Henig 1985, Warburton 1987, Kolen et al. 1987, Solomon 1987, Abdul-Razak et Potts 1988, Bianco et Ricciardelli 1997). Cet article propose donc d'explorer cette voie, à savoir la réduction de l'espace des étiquettes, dans le cadre de la résolution du problème de plus court chemin avec contraintes de ressources par un algorithme de type programmation dynamique. On se propose de formuler mathématiquement le modèle qui correspond à la procédure de dominance (section 3). On peut ainsi mieux comprendre ce que représente la dominance par rapport à un sous-ensemble de ressources, et étudier les propriétés des solutions induites. Une formulation mathématique permet également de généraliser la notion de dominance sur un espace projeté quelconque et d'identifier ceux qui fournissent la solution optimale globale (section 4). Une comparaison entre différentes approches ainsi que leurs limites sont données dans la section 5. Afin de contourner les difficultés liées à ces approches, la section 6 propose une alternative basée sur la relaxation lagrangienne. Cette procédure peut être combinée avec l'algorithme de réoptimisation de Desrochers et Soumis (1988b) pour bâtir des chemins réalisables. Afin de mieux illustrer les différentes idées, nous nous plaçons dans le cas à une seule ressource dans les sections 2 à 7. Une généralisation au cas de plusieurs ressources est présentée dans la section 8. La section 9 présente un algorithme générique de génération de colonnes (GENCOL) incluant la nouvelle procédure de résolution du sous-problème.

2 Problème de plus court chemin avec fenêtres de temps

Le problème de plus court chemin avec fenêtres de temps (PCC-FT), se formule comme suit :

$$\min \sum_{i,j} c_{ij} x_{ij} \quad (2.1)$$

$$s.c. \quad \sum_i x_{ij} - \sum_i x_{ji} = e_j = \begin{cases} -1 & \text{si } j = o \\ 0 & \text{si } j \in \mathcal{N} \\ 1 & \text{si } j = d \end{cases} \quad (2.2)$$

$$x_{ij} \geq 0 \quad \forall (i,j) \in \mathcal{A} \quad (2.3)$$

$$x_{ij} (T_i + t_{ij} - T_j) \leq 0 \quad \forall (i,j) \in \mathcal{A} \quad (2.4)$$

$$T_j \in [a_j, b_j] \quad \forall j \in \mathcal{V} \quad (2.5)$$

Ce problème peut être résolu par un algorithme de type programmation dynamique. Considérons un graphe acyclique dont les noeuds sont classés dans un ordre topologique. L'algorithme de programmation dynamique traite séquentiellement les noeuds de la source jusqu'à la destination. En chaque noeud, il génère des étiquettes en prolongeant les chemins correspondant aux étiquettes efficaces présentes aux noeuds prédécesseurs. Les prolongements qui fournissent des chemins illégaux sont ensuite supprimés. Puis il applique la règle de dominance pour éliminer tous les chemins correspondants à des étiquettes non efficaces.

Soit $F(S, j, t)$ le coût minimum du chemin allant de l'origine o au noeud j , $j \in \mathcal{N} \cup \{d\}$, passant par tous les noeuds de $S \subseteq \mathcal{N} \cup \{d\}$ et arrivant au noeud j au temps t ou plus tard. Le calcul du coût $F(S, j, t)$ se fait en résolvant les relations de récurrence suivantes :

$$\begin{aligned} F(\emptyset, o, a_o) &= 0 \\ F(S, j, t) &= \min_{\substack{i|(i,j) \in \mathcal{A} \\ \forall S \subseteq \mathcal{N} \cup \{d\}, j \in S, a_j \leq t \leq b_j}} \{F(S - \{j\}, i, t') + c_{ij} \mid i \in S - \{j\}, t' \leq t - t_{ij}, a_i \leq t' \leq b_i\}, \end{aligned}$$

La solution optimale est donnée par

$$\min_{S \subseteq \mathcal{N} \cup \{d\}} \min_{a_d \leq t \leq b_d} F(S, d, t).$$

Ainsi dans un algorithme de programmation dynamique de type "pulling", la génération des étiquettes se fait comme suit. Pour un noeud j donné, des étiquettes sont créées en prolongeant celles réalisables et présentes aux noeuds i , tels que $(i, j) \in \mathcal{A}$. Plus précisément, une nouvelle étiquette (T_j, C_j) donnée par

$$\begin{aligned} T_j &= \max\{a_j, T_i + t_{ij}\} \\ C_j &= C_i + c_{ij} \end{aligned}$$

est créée au noeud j si $T_i + t_{ij} \leq b_j$.

3 Dominance dans l'espace projeté

Dans le cas d'une seule ressource, l'espace des états est de dimension deux. La matrice de projection $\Pi(j)$ au noeud j devient $\Pi(j) = (\Pi_{11}(j), \Pi_{12}(j))$. En considérant que le coefficient $\Pi_{12}(j)$ (celui associé au coût) doit être non nul, la matrice $\Pi(j)$ peut s'écrire sous la forme $(\pi_j, 1)$, $\pi_j \in \mathbb{R}$. La qualité de la solution obtenue en dominant par rapport à la combinaison "coût + $\pi_j \times$ temps" dépend du choix du coefficient π_j . En le choisissant par exemple égal à zéro, la dominance revient à garder en chaque noeud l'étiquette de meilleur coût.

En considérant que tous les prédécesseurs du noeud $j \in \mathcal{N}$ sont déjà traités, la dominance dans l'espace projeté peut être interprétée comme la résolution de

$$\phi_j(\pi_j) = \min_i \{C_i + c_{ij} + \pi_j \times \max\{a_j, (T_i + t_{ij})\} \mid (i, j) \in \mathcal{A}, T_i + t_{ij} \leq b_j\} \quad (3.1)$$

Ainsi, on ne garde qu'une seule étiquette efficace par noeud. On peut restreindre le domaine de π_j à \mathbb{R}^+ , sans perdre de généralité, car $\pi_j = 0$ produit une solution qui domine celles produites par $\pi_j < 0$.

La dominance par rapport au coût uniquement, revient à remplacer en chaque noeud $j \in \mathcal{N}$, le calcul de $\phi_j(\pi_j)$ par celui de $\phi_j(0)$, soit

$$\phi_j(0) = \min_i \{C_i + c_{ij} \mid (i, j) \in \mathcal{A}, T_i + t_{ij} \leq b_j\}. \quad (3.2)$$

Par ailleurs, il faut noter que pour le noeud origine o , il n'y a pas de chemin à traiter et qu'au noeud destination d on prend l'étiquette de coût minimum. Par suite, le vecteur $\pi = (\pi_j)_{j \in \mathcal{N}}$ n'est défini que pour les éléments de \mathcal{N} . Rappelons que le but est d'optimiser le coût au noeud terminal d . Soit $\Phi(\pi)$ le coût du plus court chemin obtenu en dominant en chaque noeud $j \in \mathcal{N}$, par rapport à la combinaison "coût + $\pi_j \times$ temps". La meilleure approximation de la valeur optimale du problème PCC-FT fournie par cette approche est obtenue en résolvant

$$\begin{cases} \min_{\pi} & \Phi(\pi) \\ \text{s.c.} & \pi_j \geq 0, \quad \forall j \in \mathcal{N} \end{cases} \quad (3.3)$$

qui consiste à déterminer le meilleur vecteur π . Le chemin ainsi obtenu correspond à une solution primale réalisable, mais pas forcément optimale globale.

Pour étudier les propriétés d'une telle solution, exprimons les conditions d'optimalité de premier ordre de Kuhn et Tucker.

4 Interprétation par les conditions de Kuhn et Tucker

Les conditions de Kuhn et Tucker associées au problème de plus court chemin avec fenêtres de temps peuvent être obtenues en dualisant les contraintes (2.2) et (2.4). Mais auparavant, et afin d'obtenir des variables duales par noeud, récrivons le problème PCC-FT sous la forme équivalente suivante :

$$\min \quad \sum_{i,j} c_{ij} x_{ij} \quad (4.1)$$

$$\text{s.c.} \quad \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1 & \text{si } i = o \\ 0 & \text{si } i \in \mathcal{N} \\ -1 & \text{si } i = d \end{cases} \quad (4.2)$$

$$x_{ij} \geq 0 \quad \forall (i, j) \in \mathcal{A} \quad (4.3)$$

$$\sum_i x_{ij} (T_i + t_{ij} - T_j) \leq 0 \quad \forall j \in \mathcal{N} \cup \{d\} \quad (4.4)$$

$$T_j \in [a_j, b_j] \quad \forall j \in \mathcal{V} \quad (4.5)$$

Le système d'optimalité de Kuhn et Tucker, obtenu en dualisant les contraintes (4.2) et (4.4), peut s'écrire comme suit :

$$c_{ij} + \lambda_i - \lambda_j + \sigma_j(T_i + t_{ij} - T_j) \geq 0 \quad (4.6)$$

$$x_{ij}(c_{ij} + \lambda_i - \lambda_j + \sigma_j(T_i + t_{ij} - T_j)) = 0 \quad (4.7)$$

$$\lambda_j \left(\sum_i x_{ij} - \sum_i x_{ji} - e_j \right) = 0 \quad (4.8)$$

$$\sum_i \sigma_j x_{ij}(T_i + t_{ij} - T_j) = 0 \quad (4.9)$$

$$x_{ij} \geq 0 \quad (4.10)$$

$$\sigma_j \geq 0 \quad (4.11)$$

$$\lambda_j \in \mathbb{R} \quad (4.12)$$

$$T_j \in [a_j, b_j] \quad (4.13)$$

où $\lambda_j \in \mathbb{R}$, $j \in \mathcal{V}$, sont les variables duales associées aux contraintes (4.2), et $\sigma_j \geq 0$, $j \in \mathcal{N} \cup \{d\}$, sont celles associées aux contraintes (4.4).

Remarque 1 Les contraintes (4.4) étant non convexes, les conditions d'optimalité (4.6)-(4.13) ne caractérisent pas l'optimum global, comme le montre l'exemple 1.

Remarque 2 Dans un graphe acyclique, si on traite les noeuds dans l'ordre topologique, résoudre séquentiellement les équations (3.1) revient à résoudre le système (4.6)-(4.13) avec en plus les contraintes primales (4.2) et (4.4).

Théorème 1 Si \bar{x} est une solution du problème PCC-FT obtenue en dominant par rapport à la nouvelle ressource "coût + $\pi \times$ temps", alors il existe des vecteurs $\bar{\lambda}$, \bar{T} et $\bar{\sigma}$ tels que $(\bar{x}, \bar{\lambda}, \bar{T}, \bar{\sigma})$ soit solution du système (4.6)-(4.13)

Preuve : Soit \bar{x} une solution du problème PCC-FT obtenue en dominant par rapport à la combinaison "coût + $\pi \times$ temps". \bar{x} est donc solution des équations (3.1), où $\pi = (\pi_j)_{j \in \mathcal{N}}$ est un vecteur réel positif donné. Soient $\bar{\lambda} = C$ et \bar{T} les vecteurs des variables coût et temps associées à \bar{x} . Montrons qu'il existe $\bar{\sigma}$ telle que $(\bar{x}, \bar{\lambda}, \bar{T}, \bar{\sigma})$ soit solution du système (4.6)-(4.13). Soit $(i, j) \in \mathcal{A}$, distinguons deux cas :

1er cas : $x_{ij} = 1$.

Par construction, on a, $\bar{\lambda}_j = \bar{\lambda}_i + c_{ij}$ et $\bar{T}_j \geq \bar{T}_i + t_{ij}$. Il suffit de prendre $\bar{\sigma}_j = 0$, pour satisfaire le système (4.6)-(4.13).

2ème cas : $x_{ij} = 0$.

les équations (4.7)-(4.13) sont vérifiées indépendamment de la valeur de $\bar{\sigma}_j$. Il reste à vérifier l'inéquation (4.6). Puisqu'on a dominé par rapport à la combinaison "coût + $\pi \times$ temps", l'étiquette $(\bar{\lambda}_j, \bar{T}_j)$ est dominante, s'exprime de la manière suivante : $\bar{\lambda}_j + \pi_j \bar{T}_j \leq \bar{\lambda}_i + c_{ij} + \pi_j \times \max \{a_j, (\bar{T}_i + t_{ij})\}$. Trois cas sont possibles :

- $a_j \leq \bar{T}_i + t_{ij} \leq b_j$: il suffit alors de prendre $\bar{\sigma}_j = \pi_j$ pour satisfaire (4.6).
- $\bar{T}_i + t_{ij} < a_j$: en choisissant $\bar{\sigma}_j = \pi_j \left(1 - \frac{a_j - \bar{T}_i + t_{ij}}{\bar{T}_j - \bar{T}_i + t_{ij}}\right)$, l'équation (4.6) est vérifiée.
- $\bar{T}_i + t_{ij} > b_j$: en prenant $\bar{\sigma}_j = \max \left\{0, -\frac{\bar{\lambda}_i + c_{ij} - \bar{\lambda}_j}{\bar{T}_i + t_{ij} - \bar{T}_j}\right\}$, l'équation (4.6) est satisfaite.

⊠

Exemple 1

Considérons le problème de plus court chemin avec une contrainte de ressource, représenté dans la figure 1. La solution optimale globale correspond au chemin (X_1, Y_1) .

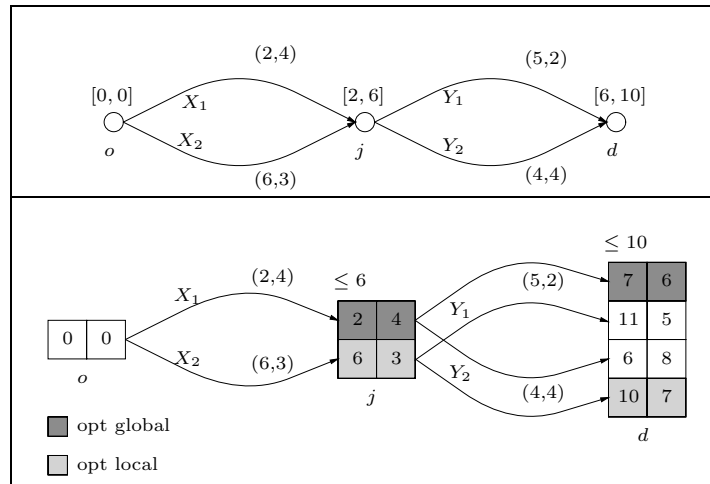


FIG. 1 – Exemple 1 - ensemble des étiquettes dans le plan (ressource, coût)

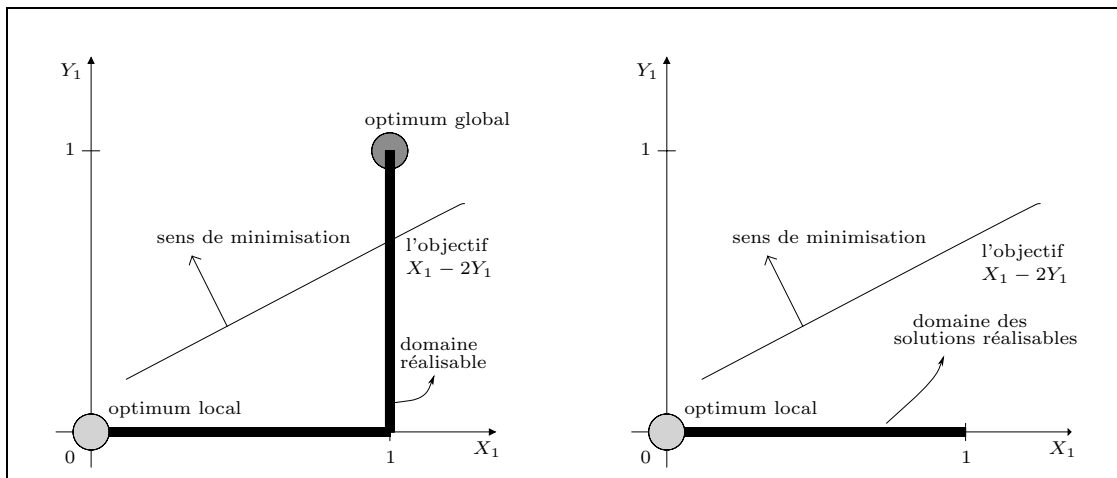


FIG. 2 – Exemple 1 - représentation dans le plan (X_1, Y_1) du domaine réalisable pour $T_j \in [2, 6[$ et $T_j \geq 6$

Par contre, en dominant uniquement sur le coût, $\pi_j = 0$, l'algorithme de programmation dynamique fournit la solution (X_2, Y_2) , qui correspond à un optimum local (figure 2).

On peut vérifier aisément que la solution (X_2, Y_2) vérifie les conditions de Kuhn et Tucker (4.6)-(4.13), avec les valeurs des variables $\lambda_o = 0$, $\lambda_j = 3$, $\lambda_d = 7$, $T_o = 0$, $T_j = 6$, $T_d = 10$, $\sigma_o = \sigma_i = \sigma_d = 0$.

Par ailleurs, la dominance par rapport à la combinaison "coût + $\pi_j \times$ temps", avec $\pi_j > 1/4$ fournit la solution optimale globale (X_1, Y_1) . Cependant, dans le cas général, un tel π_j peut ne pas exister comme le montre l'exemple 2 suivant.

Exemple 2

Parmi les étiquettes du noeud terminal d données dans la figure 3, l'étiquette $(10, 6)$ qui correspond à la solution optimale globale, représente le quatrième chemin arrivant au noeud d , soit $(i^{(3)}, j^{(4)}, d^{(4)})$ (l'indice en exposant correspond à l'ordre de l'étiquette au noeud considéré dans la figure 3). Un algorithme de programmation dynamique de type "pulling" où on domine uniquement par rapport au coût, fournit une solution de coût 9. Elle correspond à deux étiquettes $(2, 9)$ (chemin $(j^{(1)}, d^{(1)})$) et $(10, 9)$ (chemin $(i^{(4)}, d^{(9)})$). Dans un tel cas, on préférera la première étiquette puisqu'elle correspond au chemin qui consomme le moins.

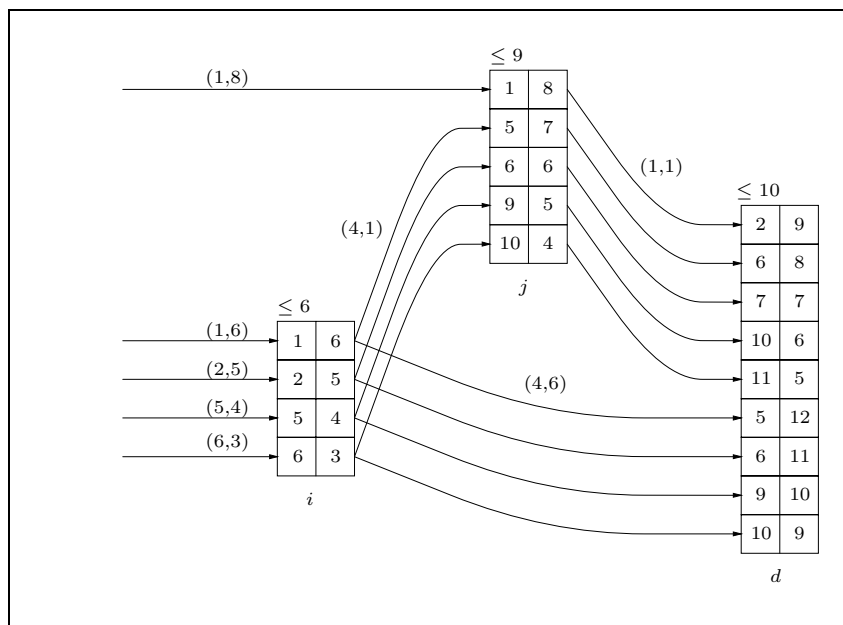


FIG. 3 – Exemple 2 - ensemble des étiquettes dans le plan (ressource, coût)

Comme on peut le voir sur la figure 4, la dominance par rapport à la combinaison "coût + $\pi_i \times$ temps", conduit aux étiquettes $(6, 3)$, $(2, 5)$ et $(1, 6)$ respectivement selon les intervalles des valeurs de π_i respectifs $[0, 1/2]$, $[1/2, 1]$ et $[1, +\infty[$. Il n'existe donc pas de

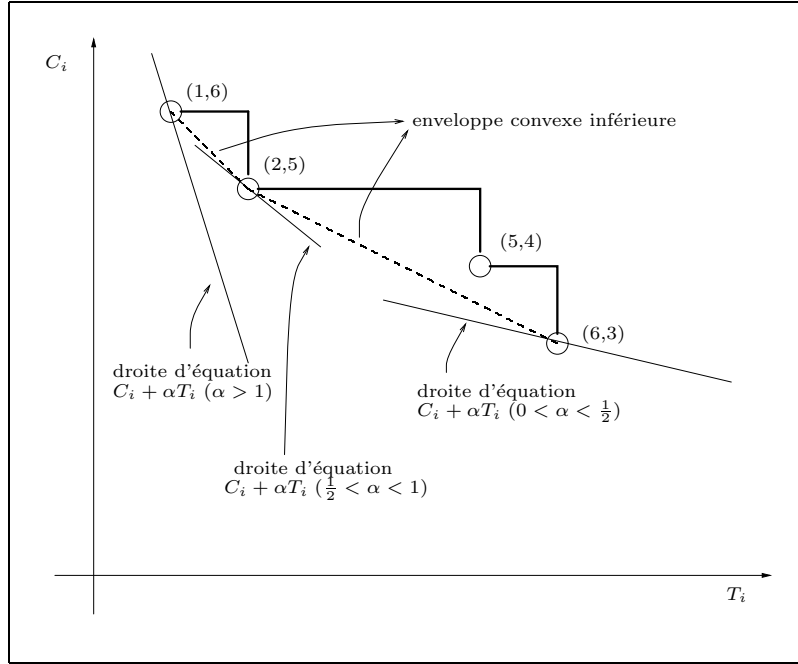


FIG. 4 – Exemple 2 - représentation des étiquettes au noeud i dans le plan (ressource, coût)

valeur de π_i permettant de récupérer l'étiquette (5, 4) au noeud i , et par suite la solution optimale globale.

Par contre, en considérant un coefficient de projection qui dépend, en plus du noeud traité, du noeud prédécesseur, il existe toujours une instanciation de ces coefficients qui permet de récupérer la solution optimale globale, comme le montre le théorème 2. En effet, supposons que tous les noeuds i prédécesseurs de j sont traités. La résolution en dominant dans l'espace correspondant à une telle projection, peut être interprétée comme la résolution de

$$\tilde{\phi}_j(\tilde{\pi}_{.j}) = \min_i \{C_i + c_{ij} + \tilde{\pi}_{ij} \times \max \{a_j, T_i + t_{ij}\} \mid (i, j) \in \mathcal{A}, T_i + t_{ij} \leq b_j\} \quad (4.14)$$

où les coefficients de projection $\tilde{\pi}_{ij}$ dépendent cette fois-ci du noeud traité j et de ses prédécesseurs, soit des arcs (i, j) qui aboutissent au noeud j .

Soit $\tilde{\Phi}(\tilde{\pi})$ le coût du plus court chemin obtenu en dominant en chaque noeud $j \in \mathcal{N}$ par rapport à la combinaison "cout + $\tilde{\pi}_{.j}$ × ressource". La meilleure valeur associée à la solution du problème PCC-FT fournie par cette approche est donnée par

$$\begin{cases} \min_{\tilde{\pi}} & \tilde{\Phi}(\tilde{\pi}) \\ \text{s.c.} & \tilde{\pi}_{ij} \geq 0, \quad \forall (i, j) \in \mathcal{A}. \end{cases} \quad (4.15)$$

Théorème 2

$$\min_{\tilde{\pi} \geq 0} \tilde{\Phi}(\tilde{\pi}) = v^*$$

où v^* est la valeur optimale du problème PCC-FT.

Preuve : Soit (o, j_1) le premier arc du chemin optimal global. En prenant π_{ij_1} suffisamment grand pour tout $(i, j_1) \in \mathcal{A}$ tel que $i \neq o$ et $\pi_{oj_1} = 0$, la solution de (4.14) au noeud j_1 correspondra au chemin (o, j_1) . Puis on réitère le même procédé jusqu'au noeud d . \square

Remarque 3 C'est un résultat surprenant dans le sens où le nombre de chemins est exponentiel et pourtant on peut exhiber le chemin optimal au moyen d'un nombre polynômial $(|\mathcal{A}|)$ de variables.

Une interprétation par les conditions de Kuhn et Tucker, analogue à celle donnée dans la section 4, peut être obtenue en considérant le problème PCC-FT écrit sous la forme (2.1)-(2.5) et dans lequel on dualise les contraintes (2.2) et (2.4) :

$$c_{ij} + \lambda_i - \lambda_j + \tilde{\sigma}_{ij}(T_i + t_{ij} - T_j) \geq 0 \quad (4.16)$$

$$x_{ij}(c_{ij} + \lambda_i - \lambda_j + \tilde{\sigma}_{ij}(T_i + t_{ij} - T_j)) = 0 \quad (4.17)$$

$$\lambda_j \left(\sum_i x_{ij} - \sum_i x_{ji} - e_j \right) = 0 \quad (4.18)$$

$$\tilde{\sigma}_{ij} x_{ij} (T_i + t_{ij} - T_j) = 0 \quad (4.19)$$

$$x_{ij} \geq 0 \quad (4.20)$$

$$\tilde{\sigma}_{ij} \geq 0 \quad (4.21)$$

$$\lambda_j \in \mathbb{R} \quad (4.22)$$

$$T_j \in [a_j, b_j] \quad (4.23)$$

où $\lambda_j \in \mathbb{R}$, $j \in \mathcal{V}$, sont les variables duales associées aux contraintes (2.2), et $\tilde{\sigma}_{ij} \geq 0$, $(i, j) \in \mathcal{A}$, sont celles associées aux contraintes (2.4). Notons que ces dernières variables portent sur les arcs alors que dans la formulation précédente elles portaient sur les noeuds.

5 Comparaison des bornes

Bien que les deux formulations primales (2.1)-(2.5) et (4.1)-(4.5) soient équivalentes, les systèmes de Kuhn et Tucker associés ne le sont pas. En fait, le système (4.6)-(4.13) peut être vu comme un cas particulier du système (4.16)-(4.23) correspondant à prendre $\tilde{\sigma}_{ij} = \sigma_j$ pour tout i tel que $(i, j) \in \mathcal{A}$. Ainsi la dominance en un noeud j par rapport à une combinaison linéaire du coût et de la ressource ($\text{coût} + \pi \times \text{ressource}$), dépendamment du choix de la matrice de projection, revient à résoudre : (a) l'équation (4.14) si la projection dépend du noeud traité et de ses prédécesseurs, (b) l'équation (3.1) si la projection est locale et ne dépend que du noeud traité. La dominance par rapport au coût uniquement correspond à résoudre (4.14) avec $\tilde{\pi}_{ij} = 0$, pour tout i , tel que $(i, j) \in \mathcal{A}$, ou bien (3.1) avec $\pi_j = 0$. D'où le résultat suivant :

Théorème 3

$$v^* = \min_{\tilde{\pi}} \tilde{\Phi}(\tilde{\pi}) \leq \min_{\pi} \Phi(\pi) \leq \tilde{\Phi}(0) = \Phi(0)$$

où $v^* = v(\text{PCC} - \text{FT})$ est la valeur optimale du problème PCC-FT.

Rappelons que le but est de minimiser C_d , le coût au noeud terminal d . La valeur $\min_{\tilde{\pi}} \tilde{\Phi}(\tilde{\pi})$ est meilleure que $\min_{\pi} \Phi(\pi)$ mais son calcul est beaucoup plus coûteux. En effet, la variable $\tilde{\pi}$ est de dimension $|\mathcal{A}|$ alors que π est de dimension $|\mathcal{N}|$. Néanmoins, la détermination du meilleur vecteur π demeure difficile, dû au comportement de $\Phi(\pi)$ qui n'est ni monotone, ni convexe, ni concave en fonction de chaque composante π_j (voir exemple 2) bien que la fonction $\phi_j(\pi_j)$ soit linéaire par morceaux, croissante et concave. En effet, soit \mathcal{E}_j l'ensemble de toutes les étiquettes efficaces dans le plan (*ressource, coût*) présentes au noeud j . Soit $\left\{ (T_j^{(1)}, C_j^{(1)}), (T_j^{(2)}, C_j^{(2)}), \dots, (T_j^{(q)}, C_j^{(q)}) \right\} \subset \mathcal{E}_j$ l'ensemble des étiquettes appartenant à l'enveloppe convexe inférieure de \mathcal{E}_j , telles que $T_j^{(1)} < T_j^{(2)} < \dots < T_j^{(q)}$ et $C_j^{(1)} > C_j^{(2)} > \dots > C_j^{(q)}$. On peut exprimer analytiquement la fonction $\phi_j(\pi_j)$ comme suit :

$$\phi_j(\pi_j) = \begin{cases} C_j^{(q)} + \pi_j T_j^{(q)} & \text{si } \pi_j \in \left[0, \frac{C_j^{(q-1)} - C_j^{(q)}}{T_j^{(q)} - T_j^{(q-1)}} \right] \\ C_j^{(k)} + \pi_j T_j^{(k)} & \text{si } \pi_j \in \left] \frac{C_j^{(k)} - C_j^{(k+1)}}{T_j^{(k+1)} - T_j^{(k)}}, \frac{C_j^{(k-1)} - C_j^{(k)}}{T_j^{(k)} - T_j^{(k-1)}} \right], \quad k \in \{2, \dots, q-1\} \\ C_j^{(1)} + \pi_j T_j^{(1)} & \text{si } \pi_j \in \left] \frac{C_j^{(1)} - C_j^{(2)}}{T_j^{(2)} - T_j^{(1)}}, +\infty \right[\end{cases}$$

qui est bien linéaire par morceaux, croissante et concave.

Considérons l'exemple 2. En notant que dans le plan (*ressource, coût*), l'intersection de l'ensemble $\{(1, 6), (2, 5), (6, 3)\}$ des étiquettes générées au noeud i avec son enveloppe convexe inférieure est lui même, on a :

$$\phi_i(\pi_i) = \begin{cases} 3 + 6\pi_i & \text{si } \pi_i \in \left[0, \frac{1}{2} \right] \\ 5 + 2\pi_i & \text{si } \pi_i \in \left] \frac{1}{2}, 1 \right] \\ 6 + \pi_i & \text{si } \pi_i \in]1, +\infty[\end{cases}$$

En maintenant les autres π_j invariables, par exemple en prenant $\pi_j = 0$, on obtient les valeurs suivantes pour $\Phi(\pi)$ en fonction de π_i :

$$\Phi(\pi) = \begin{cases} 9 & \text{si } \pi_i \in [0, \frac{1}{2}] \\ 7 & \text{si } \pi_i \in]\frac{1}{2}, 1] \\ 8 & \text{si } \pi_i \in]1, +\infty] \end{cases}$$

Ce qui montre que $\Phi(\pi)$ est une fonction non convexe, non concave et non monotone de la variable π_i .

Dans ces conditions, l'utilisation d'une méthode de recherche locale serait inefficace et la détermination d'une direction et un pas de déplacement pour ajuster la valeur de π_j est aussi farfelu que difficile.

6 Relaxation lagrangienne

Le comportement chaotique de la fonction $\Phi(\pi)$ est essentiellement dû à la contrainte de ressource. Afin de contourner cette difficulté, considérons le dual lagrangien du problème de plus court chemin avec fenêtres de temps, obtenu en relâchant la contrainte de borne supérieure sur la consommation de ressource, $T_j \leq b_j$. Avant de la dualiser, on la réécrit sous une forme équivalente, permettant de relier la consommation T en la ressource et la variable de flot x . Deux formulations sont possibles.

Formulation mathématique équivalente I

En utilisant la formulation (2.1)-(2.5), le problème de plus court chemin avec fenêtres de temps peut être réécrit sous la forme équivalente suivante :

$$\min \sum_{i,j} c_{ij} x_{ij} \quad (6.1)$$

$$s.c. \quad \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1 & \text{si } i = o \\ 0 & \text{si } i \in \mathcal{N} \\ -1 & \text{si } i = d \end{cases} \quad (6.2)$$

$$x_{ij} \geq 0 \quad \forall (i,j) \in \mathcal{A} \quad (6.3)$$

$$x_{ij} (T_i + t_{ij} - T_j) \leq 0 \quad \forall (i,j) \in \mathcal{A} \quad (6.4)$$

$$T_j \geq a_j \quad \forall j \in \mathcal{V} \quad (6.5)$$

$$x_{ij} (T_i + t_{ij} - b_j) \leq 0 \quad \forall (i,j) \in \mathcal{A} \quad (6.6)$$

En dualisant les contraintes (6.6), en leur associant les multiplicateurs de lagrange $\tilde{u}_{ij} \geq 0$, on obtient la fonction de lagrange :

$$\mathcal{L}_1(\tilde{u}) = \min_{(6.2) - (6.5)} \sum_{i,j} (c_{ij} + \tilde{u}_{ij}(T_i + t_{ij} - b_j)) x_{ij}.$$

Le dual lagrangien associé est donc :

$$(DR1) \quad \max_{\tilde{u} \geq 0} \mathcal{L}_1(\tilde{u})$$

Formulation mathématique équivalente II

Une seconde réécriture équivalente du problème de plus court chemin avec fenêtres de temps peut être obtenue en sommant les contraintes (6.6) sur tous les noeuds i prédécesseurs du noeud j :

$$\min \quad \sum_{i,j} c_{ij} x_{ij} \quad (6.7)$$

$$s.c. \quad \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1 & \text{si } i = o \\ 0 & \text{si } i \in \mathcal{N} \\ -1 & \text{si } i = d \end{cases} \quad (6.8)$$

$$x_{ij} \geq 0 \quad \forall (i, j) \in \mathcal{A} \quad (6.9)$$

$$x_{ij} (T_i + t_{ij} - T_j) \leq 0 \quad \forall (i, j) \in \mathcal{A} \quad (6.10)$$

$$T_j \geq a_j \quad \forall j \in \mathcal{V} \quad (6.11)$$

$$\sum_i x_{ij} (T_i + t_{ij} - b_j) \leq 0 \quad \forall j \in \mathcal{N} \cup \{d\} \quad (6.12)$$

En dualisant les contraintes (6.12), en leur associant les multiplicateurs de lagrange $u_j \geq 0$, on obtient la fonction de lagrange suivante :

$$\mathcal{L}_2(u) = \min_{(6.8) - (6.11)} \sum_{i,j} (c_{ij} + u_j (T_i + t_{ij} - b_j)) x_{ij}$$

et le dual lagrangien associé :

$$(DR2) \quad \max_{u \geq 0} \mathcal{L}_2(u)$$

Résolution du problème lagrangien

En procédant de la même façon qu'à la section 4, la résolution du problème lagrangien peut être vue comme la résolution en chaque noeud j , du problème suivant :

$$\tilde{\varphi}_j(\tilde{u}_{.j}) = \min_i \{C_i + c_{ij} + \tilde{u}_{ij} (T_i + t_{ij} - b_j) \mid (i, j) \in \mathcal{A}\} \quad (6.13)$$

dans le cas de la formulation I, et

$$\varphi_j(u_j) = \min_i \{C_i + c_{ij} + u_j (T_i + t_{ij} - b_j) \mid (i, j) \in \mathcal{A}\} \quad (6.14)$$

en utilisant la formulation équivalente II.

Comparaison des bornes

Contrairement à l'approche décrite dans la section 4, la résolution des problèmes $(DR1)$ et $(DR2)$ peut fournir des solutions non réalisables, et leurs valeurs $v(DR1)$ et $v(DR2)$ sont des minorants de la valeur optimale v^* du problème PCC-FT.

Bien que les deux formulations primales (6.1)-(6.5) et (6.7)-(6.11) soient équivalentes, le problème dual $(DR1)$ domine $(DR2)$, et fournit une meilleure borne inférieure de la valeur du problème PCC-FT. En effet, $(DR2)$ est un cas particulier de $(DR1)$ qui correspond à prendre $\tilde{u}_{ij} = u_j$, pour tout i tel que $(i, j) \in \mathcal{A}$. D'où le résultat suivant :

Théorème 4

$$v^* \geq v(DR1) \geq v(DR2).$$

Les fonctions de lagrange $\tilde{u} \mapsto \mathcal{L}_1(\tilde{u})$ et $u \mapsto \mathcal{L}_2(u)$ étant définies comme des minimums d'une collection de fonctions affines, sont concaves et linéaires par morceaux. Les programmes duaux $(DR1)$ et $(DR2)$ peuvent être résolus en utilisant un algorithme de type sous-gradient.

7 Relations entre les solutions

Le corollaire suivant découle directement des théorèmes 3 et 4.

Corollaire 1

$$\max_u \mathcal{L}_2(u) \leq \max_{\tilde{u}} \mathcal{L}_1(\tilde{u}) \leq v^* = \min_{\tilde{\pi}} \tilde{\Phi}(\tilde{\pi}) \leq \min_{\pi} \Phi(\pi).$$

Notons que les inégalités du corollaire 1 peuvent être strictes comme le montre l'exemple 2, dont la valeur optimale est $v^* = 6$. On a d'une part

$$v(DR1) = \max_{\tilde{u}} \mathcal{L}_1(\tilde{u}) = 5.5 \quad \text{et} \quad v(DR2) = \max_u \mathcal{L}_2(u) = 5.4.$$

D'autre part

$$\min_{\tilde{\pi}} \tilde{\Phi}(\tilde{\pi}) = 6, \quad \min_{\pi} \Phi(\pi) = 7 \quad \text{et} \quad \tilde{\Phi}(0) = \Phi(0) = 9.$$

Après avoir obtenu une borne inférieure en résolvant $\mathcal{L}_2(u)$ (resp. $\mathcal{L}_1(\tilde{u})$), avec un u (resp. \tilde{u}) donné, on peut construire une solution réalisable, solution de $\Phi(u)$ (resp. $\tilde{\Phi}(\tilde{u})$), en utilisant l'algorithme de réoptimisation développé par Desrochers et Soumis (1988b).

8 Problème de plus court chemin avec contraintes de ressources

Dans cette section, nous généralisons les différents résultats développés dans les sections précédentes pour le problème de plus court chemin avec plusieurs contraintes de ressources.

Formulation mathématique

Soit le problème de plus court chemin avec contraintes de ressources PCC-CR suivant :

$$\min \sum_{i,j} c_{ij} x_{ij} \quad (8.1)$$

$$s.c. \quad \sum_i x_{ij} - \sum_i x_{ji} = e_j = \begin{cases} -1 & \text{si } j = o \\ 0 & \text{si } j \in \mathcal{N} \\ 1 & \text{si } j = d \end{cases} \quad (8.2)$$

$$x_{ij} \geq 0 \quad \forall (i, j) \in \mathcal{A} \quad (8.3)$$

$$x_{ij} (T_i^r + t_{ij}^r - T_j^r) \leq 0 \quad \forall (i, j) \in \mathcal{A}, \forall r \in \mathcal{R} \quad (8.4)$$

$$T_j^r \in [a_j^r, b_j^r] \quad \forall j \in \mathcal{V}, \forall r \in \mathcal{R} \quad (8.5)$$

La généralisation au cas de plusieurs ressources de l'équation (4.14) s'écrit :

$$\phi_j(\pi_{\cdot j}) = \min_i \left\{ C_i + c_{ij} + \sum_r \pi_{ij}^r \times \max \{ a_j^r, T_i^r + t_{ij}^r \} \mid T_i^r + t_{ij}^r \leq b_j^r, (i, j) \in \mathcal{A}, r \in \mathcal{R} \right\}$$

À l'instar de l'étude présentée dans la section 4, on peut considérer une approximation plus facile à résoudre en considérant des variables π qui ne dépendent que du noeud traité pour obtenir l'équivalent de l'équation (3.1), avec une interprétation analogue. Par ailleurs, la non dominance par rapport à une ressource r donnée revient à prendre $\pi^r = 0$.

Projection de l'espace des ressources

En remplaçant l'ensemble des ressources \mathcal{R} , par son projeté $\tilde{\mathcal{R}} = \tilde{\Pi}(\mathcal{R})$, on obtient le problème de plus court chemin suivant avec le nouveau vecteur de ressources $\tilde{T} = \tilde{\Pi}(T)$:

$$\min \sum_{i,j} c_{ij} x_{ij} \quad (8.6)$$

$$s.c. \quad \sum_i x_{ij} - \sum_i x_{ji} = e_j = \begin{cases} -1 & \text{si } j = o \\ 0 & \text{si } j \in \mathcal{N} \\ 1 & \text{si } j = d \end{cases} \quad (8.7)$$

$$x_{ij} \geq 0 \quad \forall (i, j) \in \mathcal{A} \quad (8.8)$$

$$x_{ij} (\tilde{T}_i^p + \tilde{t}_{ij}^p - \tilde{T}_j^p) \leq 0 \quad \forall (i, j) \in \mathcal{A}, \forall p \in \tilde{\mathcal{R}} \quad (8.9)$$

$$\tilde{T}_j^p \in [\tilde{a}_j^p, \tilde{b}_j^p] \quad \forall j \in \mathcal{V}, \forall p \in \tilde{\mathcal{R}} \quad (8.10)$$

où $\tilde{T}_j^p = \sum_r \tilde{\pi}_{pr} T_j^r$, $\tilde{a}_j^p = \sum_r \tilde{\pi}_{pr} a_j^r$, $\tilde{b}_j^p = \sum_r \tilde{\pi}_{pr} b_j^r$ et $\tilde{t}_{ij}^p = \sum_r \tilde{\pi}_{pr} t_{ij}^r$.

Si la matrice de projection est en escalier (c'est à dire, pour chaque p , il n'y a qu'un seul coefficient $\tilde{\pi}_{pr}$ non nul) alors cette projection consiste à établir une partition de l'ensemble des ressources avant de les combiner. Elle correspond ainsi à la relaxation "surrogate" du problème (8.1)-(8.5), où les contraintes (8.9) correspondent à l'agrégation de (8.4) et les contraintes (8.10) à celle de (8.5).

9 Algorithme

Un problème de fabrication d'horaires peut être modélisé comme un problème de partitionnement (Desaulniers et al. 1998). Sa résolution par une technique de génération de colonnes aboutit à des sous-problèmes de plus court chemins avec contraintes de ressources. Un tel algorithme de génération de colonnes résout alternativement un problème maître et des sous-problèmes (PCC-CR). Ces derniers servent à générer les colonnes et leurs solutions doivent donc être primales réalisables. Pour pouvoir traiter des problèmes de taille réelle, une approche utilisée pour résoudre les sous-problèmes PCC-CR consiste à dominer sur un sous-ensemble prédéfini de ressources et générer ainsi des colonnes primales réalisables mais non nécessairement optimales. La modification proposée consiste à résoudre d'abord le dual lagrangien du problème de plus court chemin avec contraintes de ressources. Ce qui permet d'avoir une borne inférieure et des sous-gradients pour ajuster les multiplicateurs de lagrange. Puis, partant de cette solution duale, construire une solution primale réalisable (voir figure 5) en utilisant une adaptation au cas de plusieurs ressources de l'algorithme de réoptimisation de Desrochers et Soumis (1988b).

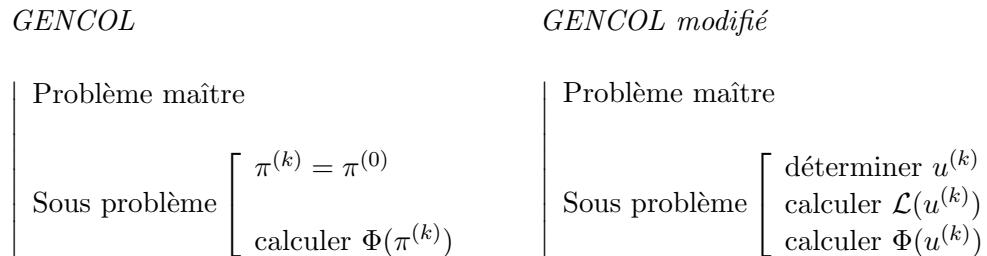


FIG. 5 – Modification proposée à *GENCOL*

Il faut noter qu'en considérant un $u^{(k)}$ constant pour toutes les itérations k de la génération de colonnes, on retrouve la version classique de *GENCOL*. Dans la version *GENCOL modifié*, plusieurs stratégies sont possibles. À chaque itération k de *GENCOL*, on peut déterminer le multiplicateur de lagrange $u^{(k)}$ optimal, ou bien on peut effectuer une seule mise à jour du multiplicateur de lagrange, avant de réoptimiser pour déterminer une solution réalisable. Des expérimentations sont en cours pour déterminer la meilleure stratégie. Elles ont également pour but d'estimer le saut de dualité $\Phi(u^{(k)}) - \mathcal{L}(u^{(k)})$ et d'apprécier les différences entre les bornes supérieures $\Phi(\pi^{(0)})$ et $\Phi(u^{(k)})$.

10 Conclusion

Dans l'algorithme de programmation dynamique, la dominance dans un espace projeté tout en gardant la réalisabilité par rapport aux ressources d'origine, permettrait en utilisant un espace mémoire et un temps moindres de générer des solutions primales réalisables.

Cependant, l'existence d'optimums locaux et le comportement chaotique de la valeur de l'objectif en fonction des coefficients de la matrice de projection ne permet pas d'élaborer une procédure d'ajustement de ces coefficients. Par contre, en ne conservant que la réalisabilité par rapport aux ressources agrégées, le problème induit correspond au dual lagrangien et/ou "surrogate" du problème primal, et peut être ainsi résolu par un algorithme classique de résolution d'un programme dual. Dans un algorithme de génération de colonnes où les sous-problèmes sont des problèmes de plus court chemins avec contraintes de ressources (Desrosiers et al. 1995), cette approche peut être complétée par l'algorithme de réoptimisation de Desrochers et Soumis (1988b) pour générer des solutions primales réalisables. De plus, l'obtention de bornes supérieure et inférieure permet l'utilisation d'heuristiques pour la génération de coupes dans le cadre de la méthode des centres analytiques.

Remerciements

Cette recherche a été supportée par le CRSNG (Conseil de Recherche en Sciences Naturelles et Génie), le Réseau RCM2 et un projet R. et D. Coop. Les auteurs souhaitent également remercier Guy Desaulniers et Ahmed Hadjar qui ont bien voulu réviser le texte.

Références

- [1] T. S. Abdul-Razak et C. N. Potts (1988), Dynamic Programming State-Space Relaxation for Single-Machine Scheduling, *Journal of Operational Research Society*, vol. 39, n. 2, p. 141-152.
- [2] Y. P. Aneja, V. Aggarwal et K. P. K. Nair (1983), Shortest Chain Subject to Side Constraints, *Networks*, 13, p. 295-302.
- [3] L. Bianco et S. Ricciardelli (1997), Dynamic Programming Strategies for the Traveling Salesman Problem with Time Window and Precedence Constraints, *Operations Research*, vol. 45, n. 3, p. 365-377.
- [4] N. Christofides, A. Mingozzi et P. Toth (1981), Exact Algorithms for the Vehicle Routing Problem Based on Spanning Tree and Shortest Path Relaxations. *Mathematical Programming*, 20, p. 255-282.
- [5] G. Desaulniers et D. Villeneuve (1997), The Shortest Path Problem with Time Windows and Linear Waiting Costs, *Les Cahiers du GERAD*, G-97-21, Avril 1997, révisé Août 1998.
- [6] G. Desaulniers, J. Desrosiers, M. Gamache et F. Soumis (1998), Crew Scheduling in Air Transportation, *Fleet Management and Logistics*, T. Crainic et G. Laporte (eds.), Kluwer, p. 169-185.
- [7] M. Desrochers (1986), La fabrication d'horaires de travail pour les conducteurs d'autobus par une méthode de génération de colonnes, *Thèse de Ph.D.*, Université de Montréal, Montréal, Canada.

- [8] M. Desrochers et F. Soumis (1988a), A Generalized Permanent Labelling Algorithm for the Shortest Path Problem with Time Windows, *INFOR* 26, p. 191-212.
- [9] M. Desrochers et F. Soumis (1988b), A Reoptimization Algorithm for the Shortest Path Problem with Time Windows, *European Journal of Operational Research*, 35, p. 242-254.
- [10] J. Desrosiers, P. Pelletier et F. Soumis (1983), Plus court chemin avec contraintes d'horaires, *RAIRO Recherche Opérationnelle*, vol. 17, n. 4, p. 357-377.
- [11] J. Desrosiers, Y. Dumas, M. M. Solomon et F. Soumis (1995), Time Constrained Routing and Scheduling, *M.O. Ball et al. (eds), Network Routing, Handbook in Operations Research and Management Science 8, Elsevier Science, Amsterdam*, p. 35-139.
- [12] G. Y. Handler et I. Zang (1980), A Dual Algorithm for the Constrained Shortest Path Problem, *Networks*, 10, p. 293-310.
- [13] M. I. Henig (1985), The Shortest Path Problem with Two Objective Functions, *European Journal of Operational Research*, 25, p. 281-291.
- [14] J. M. Jaffe (1984), Algorithms for Finding Paths with Multiple Constraints, *Networks*, 14, p. 95-116.
- [15] A. W. Kolen, A. H. G. Rinnooy Kan et H. W. J. M. Trienekens (1987), Vehicle Routing with Time Windows, *Operations Research*, vol. 35, n. 2, p. 266-273.
- [16] N. Maculan et J. J. C. Salles (1991), A Lower Bound for the Shortest Hamiltonian Path in Directed Graphs, *OR Spektrum*, 13, p. 99-102.
- [17] M. Minoux (1975), Plus court chemin avec contraintes : algorithmes et applications, *Annales des télécommunications*, tome 30, p. 326-329.
- [18] M. M. Solomon (1987), Algorithms for the Vehicle Routing and Scheduling Problem with Time Windows Constraints, *Operations Research*, vol. 35, n. 2, p. 254-265.
- [19] A. Warburton (1987), Approximation of Pareto Optima in Multiple-Objective Shortest-Path Problems, *Operations Research*, vol. 35, n. 1, p. 70-79.
- [20] D. J. White (1982), The Set of Efficient Solutions for Multiple Objective Shortest Path Problems, *Comput. and Operations Research*, vol. 9, n. 2, p. 101-107.