ECMA

Pierrre Cortambert, Victor Spitzer

Décembre 2021

Table des matières

1 Modélisation papier

1.1 Question 1 : Modélisation du problème statique sous la forme d'un PLNE compact

On se place dans un graphe orienté $G = (V, A)$, et on cherche le chemin de durée la plus courte d'une source s vers une destination t dans ce graphe. On impose que la somme des poids de chaque sommet visité par ce chemin soit bornée par une constante fixée.

On propose une modélisation PLNE compacte du problème statique, c'est à dire en supposant fixées les durées de trajet d_{ij} et les poids de chaque sommet p_i. Pour cela, on s'inspire de la modélisation PLNE compacte d'un problème de flot, en posant x_{ij} le flot sur l'arc $(i, j) \in A$. Ainsi on a :

$$
\begin{aligned}
\min_x & \sum_{(i,j)\in A} x_{ij}.d_{ij} \\
& \sum_{k;(k,i)\in A} x_{ki} - \sum_{q;(i,q)\in A} x_{iq} = 0, \quad \forall i \in V \backslash \{s,t\} \\
& \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1 \\
& \sum_{(i,j)\in A} x_{ij}.p_i + p_t \leq S \\
& x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A
\end{aligned}
$$

1.2 Question 2 : Modélisation du problème robuste sous la forme d'un PLNE

On cherche à présent à formuler une modélisation du problème robuste. Désormais les valeurs que peuvent prendre les durées et les poids sont représentées par les ensembles $\mathcal{U}_1, \mathcal{U}_2$, et on cherche une solution optimale dans le pire des cas. On reprends la formulation précédente pour intégrer les variations δ^1_{ij} et δ^2_i sur les durées et poids sous la forme de variables sous contraintes du problème tel que :

$$
\begin{aligned}
\min_x \max_{\delta^1\in\mathcal{U}_1} & \sum_{(i,j)\in A} x_{ij}.d_{ij} + x_{ij}.d_ij.\delta^1_{ij} \\
& \sum_{k;(k,i)\in A} x_{ki} - \sum_{q;(i,q)\in A} x_{iq} = 0, \quad \forall i \in V \backslash \{s,t\} \\
& \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1 \\
& \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta^2_i\right) + p_t + \delta^2_t.\hat{p}_t \leq S, \quad \forall \delta^2 \in \mathcal{U}_2 \\
& x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A
\end{aligned}
$$

On en déduit la formulation PLNE du problème robuste en intégrant les contraintes sur les ensembles $\mathcal{U}_1$ et $\mathcal{U}_2$:

$$\begin{aligned}
\min_x \max_{\delta^1} & \sum_{(i,j)\in A} x_{ij}.d_{ij} + x_{ij}.d_{ij}.\delta^1_{ij} \\
& \sum_{(k,i)\in A} x_{ki} - \sum_{(i,q)\in A} x_{iq} = 0, \quad \forall i \in V \\
& \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1 \\
& \max_{\delta^2} \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta^2_i\right) + p_t + \delta^2_t.\hat{p}_t \leq S \\
& \sum \delta^1_{ij} \leq d_1 \text{ et } 0 \leq \delta^1_{ij} \leq D_{ij}, \quad \forall (i,j) \in A \\
& \sum \delta^2_i \leq d_2 \text{ et } 0 \leq \delta^2_i \leq 2, \quad \forall i \in V \\
& x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A
\end{aligned}$$

1.3 Question 3 : Resolution par plans coupants

Question 3.a

On réécrit le problème pour que la robustesse n'apparaisse que dans les contraintes :

$$\begin{aligned}
\min_{x,z} & \sum_{(i,j)\in A} x_{ij}.d_{ij} + z \\
& \sum_{(i,j)\in A} x_{ij}.d_{ij}.\delta^1_{ij} \leq z, \quad \forall \delta^1 \in \mathcal{U}_1 \\
& \sum_{k;(k,i)\in A} x_{ki} - \sum_{q;(i,q)\in A} x_{iq} = 0, \quad \forall i \in V\backslash\{s,t\} \\
& \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1 \\
& \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta^2_i\right) + p_t + \delta^2_t.\hat{p}_t \leq S, \quad \forall \delta^2 \in \mathcal{U}_2 \\
& x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A
\end{aligned}$$

Question 3.b

Le principe de l'algorithme de coupe est d'ignorer initialement les contraintes de borne supérieure sur les coordonnées des variables $\delta^1_{i,j}$ et δ^2_i car les valeurs sur ces coordonnées n'ont d'importance que si on étudie un chemin réalisable passant respectivement par l'arc ou le sommet correspondant. On ajoute donc de nouvelles bornes sur ces variables au fur et à mesure que de nouveaux chemins sont explorés, sans considérer en permanence tous les chemins possibles, réalisables ou non.

Pour cette raison, on définit les ensembles $\mathcal{U}_1^*$ et $\mathcal{U}_2^*$ des coupes initiales pour obtenir une première solution réalisable pour le problème statique, et ajouter de nouvelles coupes en fonction de cette première solution. On fixe donc initialement tous les deltas à une valeur nulle :

$$\begin{cases} \mathcal{U}_0^* = \{0_{|A|}\} \\ \mathcal{U}_1^* = \{0_{|V|}\} \end{cases}$$

Question 3.c

Soit x^* la solution optimale de l'itération courante du problème maître, le sous-problème sur l'objectif est défini ainsi :

$$\begin{aligned} \max_{\delta^1} & \sum_{(i,j)\in A} x_{ij}^*.d_{ij}.(1+\delta_{ij}^1) \\ & \sum_{i,j} \delta_{ij}^1 \leq d_1 \\ & \delta_{ij}^1 \leq D_{ij}, \quad \forall (i,j) \in A \\ & \delta^1 \in \mathbb{R}_+^{|A|} \end{aligned}$$

De la même manière, le sous-problème sur les contraintes est défini par :

$$\begin{aligned} \max_{\delta^2} & \sum_{(i,j)\in A} x_{ij}^*.\hat{p}_i.\delta_i^2 + \delta_t^2.\hat{p}_t \\ & \sum_{i} \delta_i^2 \leq d_2 \\ & \delta_i^2 \leq 2, \quad \forall i \in V \\ & \delta^2 \in \mathbb{R}_+^{|V|} \end{aligned}$$

Question 3.d

Pour qu'une solution du problème maître soit optimale, il est nécessaire que les solutions des sous-problèmes objectif et contrainte appartiennent respectivement aux ensembles $\mathcal{U}_1^*$ et $\mathcal{U}_2^*$ à l'itération courante, et qu'on aura construites de manière itérative. Cela signifiera que cette solution est réalisable et optimale lorsqu'on considère le pire cas sur les valeurs prises par les paramètres δ^1 et δ^2.

Autrement dit, pour x^* la solution optimale de l'itération courante du problème maître et u_0^*, u_1^* les solutions des deux sous-problèmes correspondant, la solution x^* est optimale si :

$$\begin{cases} u_0^* \in \mathcal{U}_0^* \\ u_1^* \in \mathcal{U}_1^* \end{cases}$$

Question 3.e

On cherche l'expression des coupes ajoutées par les deux sous-problèmes. Ces coupes sont ajoutés en intégrant aux ensembles $\mathcal{U}_1^*$ et $\mathcal{U}_2^*$ une solution optimale des sous-problèmes pour un x^* solution du problème maître à chaque itération.

Résoudre le premier sous-problème revient à résoudre un problème de sac-à-dos à variables réelles car on peut le reformuler ainsi :

$$\begin{aligned} \max_y \sum_{(i,j)\in x^*} d_{ij}.y_{ij} \\ \sum_{(i,j)\in x^*} y_{ij} \leq d_1 \\ y_{ij} \in [0, D_{ij}] \end{aligned}$$

D'où on déduit une solution optimale définie pour (i,j) arc de x^* par $\delta^1_{ij} = y_{ij}$ et $\delta^1_{ij} = 0$ sinon. Une solution optimale s'obtient en triant les variables concernées (sur le chemin x^*) dans l'ordre décroissant des d_{ij} et en fixant dans cet ordre les variables à leur borne supérieure D_{ij} tant que la contrainte de capacité est respectée, puis en imposant à la variable suivante la valeur permettant d'atteindre l'égalité sur la contrainte de capacité. On fixe à 0 les variables restantes.

On résout de la même manière le second sous-problème, en le reformulant comme un problème de sac-à-dos en variable réelle :

$$\begin{aligned} \max_y \sum_{i\in x^*} y_i.\hat{p}_i \\ \sum_{i\in x^*} y_i \leq d_2 \\ y_i \in [0, 2] \end{aligned}$$

D'où on déduit une solution optimale définie pour i sommet de x^* par $\delta^2_i = y_i$ et $\delta^2_i = 0$ sinon. Une solution optimale s'obtient en triant les variables concernées (sur le chemin x^*) dans l'ordre décroissant des $\hat{p}_i$ et en fixant dans cet ordre les variables à leur borne supérieure de valeur 2 tant que la contrainte de capacité est respectée, puis en imposant à la variable suivante la valeur permettant d'atteindre l'égalité sur la contrainte de capacité. On fixe à 0 les variables restantes.

1.4 Question 4 : Résolution par dualisation

Question 4.a

On réécrit la formualtion donnée à la question 2 correspondant au problème robuste, mais on isole ici le terme faisant intervenir les variables δ^1_{ij}.

$$
\begin{aligned}
&\min_x \sum_{(i,j)\in A} x_{ij}.d_{ij} + \max_{\delta^1\in\mathcal{U}_1} \sum_{(i,j)\in A} x_{ij}.d_{ij}.\delta^1_{ij}\\
&\quad \sum_{k;(k,i)\in A} x_{ki} - \sum_{q;(i,q)\in A} x_{iq} = 0, \quad \forall i \in V\backslash\{s,t\}\\
&\quad \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1\\
&\quad \max_{\delta^2} \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta^2_i\right) + p_t + \delta^2_t.\hat{p}_t \leq S\\
&\quad \sum \delta^1_{ij} \leq d_1 \text{ et } 0 \leq \delta^1_{ij} \leq D_{ij}, \quad \forall (i,j) \in A\\
&\quad \sum \delta^2_i \leq d_2 \text{ et } 0 \leq \delta^2_i \leq 2, \quad \forall i \in V\\
&\quad x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A
\end{aligned}
$$

Question 4.b

Le problème interne lié aux variables δ^1_{ij} est :

$$
\begin{aligned}
&\max_{\delta^1\in\mathcal{U}_1} \sum_{(i,j)\in A} x_{ij}.d_{ij}.\delta^1_{ij}\\
&\quad \sum_{(i,j)\in A} \delta^1_{ij} \leq d_1\\
&\quad 0 \leq \delta^1_{ij} \leq D_{ij}, \quad \forall (i,j) \in A
\end{aligned}
$$

Question 4.c

Le dual s'écrit :

$$
\begin{aligned}
\min_\alpha \quad & \alpha_0.d_1 + \sum_{(i,j)\in A} \alpha_{ij}.D_{ij} & \\
& \alpha_0 + \alpha_{i,j} \geq x_{ij}.d_{ij} & \forall (i,j) \in A\\
& 0 \leq \alpha &
\end{aligned}
$$

Question 4.d

Dans la contrainte robuste, on isole les termes liés à δ^2_{ij} :

$$
\begin{aligned}
&\max_{\delta^2} \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta_i^2\right) + p_t + \delta_t^2.\hat{p}_t \leq S \\
&\sum_{i\in V} \delta_i^2 \leq d_2 \\
&0 \leq \delta_i^2 \leq 2, \quad \forall i \in V
\end{aligned}
$$

Question 4.e

Le probème interne lié aux variables δ_i^2 ne prend pas en compte la contrainte inférieure ou égale à S. Par ailleurs le sous-problème a pour seule variable (δ_i^2) et on suppose x fixé. Ainsi le problème est le suivant :

$$
\begin{aligned}
&\max_{\delta^2} \sum_{i\in V} \Big(\sum_{j;\,(i,j)\in A} \Big).\hat{p}_i.\delta_i^2 + \delta_t^2.\hat{p}_t + \sum_{(i,j)\in A} (x_{ij}.p_i) \\
&\sum_{i\in V} \delta_i^2 \leq d_2 \\
&0 \leq \delta_i^2 \leq 2, \quad \forall i \in V
\end{aligned}
$$

Question 4.f

En dualisant la formulation précédente, on obtient :

$$
\begin{aligned}
\min_{\alpha} \quad & \beta_0.d_2 + 2.\sum_{i\in V} \beta_i + \sum_{(i,j)\in A} (x_{ij}.p_i) \\
& \beta_0 + \beta_i \geq \hat{p}_i. \sum_{j;\,(i,j)\in A} x_{ij} && \forall i \in V\backslash\{t\} \\
& \beta_0 + \beta_t \geq \hat{p}_t \\
& 0 \leq \beta
\end{aligned}
$$

Question 4.g

On reprend le problème avec les questions précédentes, et on obtient :

$$
\begin{aligned}
&\min_{x} \sum_{(i,j)\in A} x_{ij}.d_{ij} + \max_{\delta^1\in\mathcal{U}_1} \sum_{(i,j)\in A} x_{ij}.\delta^1_{ij}\\
&\quad \sum_{k;(k,i)\in A} x_{ki} - \sum_{q;(i,q)\in A} x_{iq} = 0, \quad \forall i \in V\backslash\{s,t\}\\
&\quad \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1\\
&\quad \max_{\delta^2} \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta^2_i\right) + p_t + \delta^2_t.\hat{p}_t \leq S\\
&\quad \sum \delta^1_{ij} \leq d_1 \text{ et } 0 \leq \delta^1_{ij} \leq D_{ij}, \quad \forall (i,j)\in A\\
&\quad \sum \delta^2_i \leq d_2 \text{ et } 0 \leq \delta^2_i \leq 2, \quad \forall i \in V\\
&\quad x_{ij} \in \{0,1\}, \quad \forall (i,j)\in A
\end{aligned}
$$

Tel que, par dualité, la valeur objectif du sous-problème relatif à δ_1 est donnée par :

$$
\begin{aligned}
\min_{\alpha} \quad & \alpha_0.d_1 + \sum_{(i,j)\in A} \alpha_{ij}.D_{ij}\\
& \alpha_0 + \alpha_{i,j} \geq x_{ij}.d_{ij} \qquad \forall (i,j)\in A\\
& 0 \leq \alpha
\end{aligned}
$$

On peut alors intégrer cette valeur-ci de l'objectif du sous-problème à l'objectif du problème maître, et reformuler celui-ci ainsi :

$$
\begin{aligned}
&\min_{x,\,\alpha} \sum_{(i,j)\in A} x_{ij}.d_{ij} + \alpha_0.d_1 + \sum_{(i,j)\in A} \alpha_{ij}.D_{ij}\\
&\quad \sum_{(k,i)\in A} x_{ki} - \sum_{(i,q)\in A} x_{iq} = 0, \quad \forall i \in V\backslash\{s,t\}\\
&\quad \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1\\
&\quad \alpha_0 + \alpha_{i,j} \geq x_{ij}.d_{ij} \qquad \forall (i,j)\in A\\
&\quad 0 \leq \alpha\\
&\quad \max_{\delta^2} \sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta^2_i\right) + p_t + \delta^2_t.\hat{p}_t \leq S\\
&\quad \sum \delta^2_i \leq d_2 \text{ et } 0 \leq \delta^2_i \leq 2, \quad \forall i \in V\\
&\quad x_{ij} \in \{0,1\}, \quad \forall (i,j)\in A
\end{aligned}
$$

De même, par dualité, la valeur objectif du sous-problème relatif à δ_2 est donnée par :

$$\begin{aligned}
\min_{\alpha} \quad & \beta_0.d_2 + 2.\sum_{i\in V}\beta_i + \sum_{(i,j)\in A}(x_{ij}.p_i) \\
& \beta_0 + \beta_i \geq \hat{p}_i.\sum_{j;\,(i,j)\in A} x_{ij} \qquad\qquad \forall i \in V\backslash\{t\} \\
& \beta_0 + \beta_t \geq \hat{p}_t \\
& 0 \geq \beta
\end{aligned}$$

Et la contrainte sur le sous-problème est que la valeur optimale soit inférieure à S. Un argument suffisant est qu'il existe une solution $(\tilde{\beta}_i)$ réalisable du dual pour ce sous-problème, et qui respecte l'inégalité : il vient immédiatement que la solution réalisable d'objectif minimale du sous-problème dual soit de valeur également inférieure à S :

$$\min_{\beta} \beta_0.d_2 + 2.\sum_{i\in V}\beta_i + \sum_{(i,j)\in A}(x_{ij}.p_i) \leq \tilde{\beta}_0.d_2 + 2.\sum_{i\in V}\tilde{\beta}_i + \sum_{(i,j)\in A}(x_{ij}.p_i) \leq S$$

On propose donc une dernière refomulation du problème maître en prenant en considération cet argument de dualité, et on obtient la formulation MILP standard suivante :

$$\begin{aligned}
\min_{x,\,\alpha,\,\beta} & \sum_{(i,j)\in A} x_{ij}.d_{ij} + \alpha_0.d_1 + \sum_{(i,j)\in A}\alpha_{ij}.D_{ij} \\
& \sum_{(k,i)\in A} x_{ki} - \sum_{(i,q)\in A} x_{iq} = 0, \quad \forall i \in V\backslash\{s,t\} \\
& \sum_{q;(s,q)\in A} x_{sq} - \sum_{q;(q,s)\in A} x_{qs} = 1 \text{ et } \sum_{k;(k,t)\in A} x_{kt} - \sum_{q;(t,q)\in A} x_{tq} = 1 \\
& \alpha_0 + \alpha_{i,j} \geq x_{ij}.d_{ij} \qquad\qquad \forall (i,j) \in A \\
& 0 \leq \alpha \\
& \beta_0.d_2 + 2.\sum_{i\in V}\beta_i + \sum_{(i,j)\in A}(x_{ij}.p_i) \leq S \\
& \beta_0 + \beta_i \geq \hat{p}_i.\sum_{j;\,(i,j)\in A} x_{ij} \qquad\qquad \forall i \in V\backslash\{t\} \\
& \beta_0 + \beta_t \geq \hat{p}_t \\
& 0 \leq \beta \\
& x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A
\end{aligned}$$

2 Résolution numérique

2.1 Heuristique

On propose un algorithme de programmation dynamique pour trouver le plus court chemin sous contrainte de poids sur les sommets, si les poids sur chaque sommet sont supposés entiers. Sans perte de généralités, on considère également que le poids sur le sommet destination t est nul. Enfin, on distingue le poids d'un sommet donné en entrée du problème, et le poids d'un chemin qui corresponds à la somme des poids de ses sommets.

Cas statique

Sur chaque sommet x_i, on cherche les valeurs d'un vecteur $v(x_i) \in \mathbb{R}^S$ tel que $v_k(x_i)$ soit le couple coût-chemin de x_i à t de poids k et de coût minimal :

$$v_k(i) = \begin{cases} \min_{j;(i,j)\in A} \left(v_{k-p_i}(y) + d_{ij}\right) & \text{si } k \geq p_i \\ +\infty & \text{sinon} \end{cases}$$

On peut alors procéder par programmation dynamique en fixant initialement $v(t) = 0$ et pour tout autre sommet x_i, $v(x_i) = +\infty$. On minimise l'objectif en retrouvant dans $v(s)$ la coordonnée de plus faible coût, et son chemin associé : cette solution respectera la contrainte sur les poids car le poids de la solution correspond à son indice dans le vecteur $v(s)$, donc est inférieur à S. L'algorithme est similaire à celui de Bellman, excepté qu'ici on retient S valeurs sur chaque sommet, et le chemin optimal associé à chaque valeur correspondante.

Cas robuste : variation des distances

Considérons pour l'instant uniquement la variation sur les distances d_{ij} pour des poids fixés. On introduit une fonction $c_k(y)$ qui indique le sur-coûts dans le pire cas sur le chemin optimal de y à t pour une somme de poids k. Celui-ci s'obtient polynomialement par la résolution du sous-problème objectif de la décomposition en plan coupant. On met à jour la fonction valeur de Bellman et ses sur-coûts d'objectif :

$$v_k(i) = \begin{cases} \min_{j;(i,j)\in A} \left(v_{k-p_i}(j) + d_{ij} + \text{spo}\left(\{i\} \cup x^*_{k-p_i}(j)\right)\right) & \text{si } k \geq p_i \\ +\infty & \text{sinon} \end{cases}$$

On revient alors à la résolution d'un algorithme de programmation dynamique comme dans le cas statique.

Cas robuste complet

Considérons désormais les variations à la fois sur les distances et les poids. Pour une solution réalisable dans le cas statique notée x^*, le sur-coût des poids dans le pire cas sera :

$$\min(d_2, \sum_{i\in V} 2x^*_i.\hat{p}_i)$$

Comme précédemment, on introduit une fonction $p_k(j)$ qui indique la somme des sur-coûts de poids sur le chemin optimal de x_j à t pour une somme de poids k. On met alors à jour une dernière fois la fonction valeur de Bellman :

$$v_k(i) = \begin{cases} \min_{j;(i,j)\in A}\left(v_{k-p_i-2\hat{p}_i}(j) + d_{ij} + \text{spo}\left(\{i\} \cup x^*_{k-p_i-2\hat{p}_i}(j)\right)\right) & \text{si } k \geq p_i + 2\hat{p}_i \text{ et } p_{k-p_i-2\hat{p}_i}(j) + 2\hat{p}_i \leq d_2 \\ \min_{j;(i,j)\in A}\left(v_{k-p_i}(j) + +d_{ij} + \text{spo}\left(\{i\} \cup x^*_{k-p_i}(j)\right)\right) & \text{si } k \geq p_i \text{ et } p_k(j) + p_i + 2\hat{p}_i > d_2 \\ +\infty & \text{sinon} \end{cases}$$

On revient encore une fois à la résolution d'un algorithme de programmation dynamique comme dans le cas statique.

Complexité et approche heuristique

Contrairement à un algorithme de Bellman, on a sur chaque sommet du graphe autant de valeurs et chemins associés que chaque valeur possible du cumul de poids sur un chemin optimal. Si on considère S borné et les poids entiers, le problème reste polynomial. Sinon, celui-ci est pseudo-polynomial numérique. On remarque par ailleurs que pour limiter la complexité de l'algorithme de programmation dynamique, on peut immédiatement appliquer :

$$S \longleftarrow \min(S, \sum_{i\in V} p_i + \hat{p}_i)$$

Le modèle PLNE défini plus haut indique que l'inégalité sur les variables par rapport à S est donnée par :

$$\sum_{(i,j)\in A} \left(x_{ij}.p_i + x_{ij}.\hat{p}_i.\delta_i^2\right) + p_t + \delta_t^2.\hat{p}_t \leq S, \quad \forall \delta^2 \in \mathcal{U}_2$$

On pourrait envisager de diviser S ainsi que tous les poids p_i et $\hat{p}_i$ par une même valeur pour que le domaine des solutions réalisables reste inchangé tout en ramenant S à une valeur S' fixée :

$$\sum_{(i,j)\in A} \left(x_{ij}.p'_i + x_{ij}.\hat{p}'_i.\delta_i^2\right) + p'_t + \delta_t^2.\hat{p}'_t \leq S', \quad \forall \delta^2 \in \mathcal{U}_2$$

Cependant il est probable que les poids résultants p'_i et $\hat{p}'_i$ ne soient plus entiers, et donc qu'on ne puisse pas appliquer immédiatement l'algorithme décrit ci-dessus. Une option serait d'approcher les poids par leur valeur entière supérieure et S' par une sa valeur entière inférieure :

$$\begin{cases} p'_i \approx \ulcorner p'_i \urcorner \\ \hat{p}'_i \approx \ulcorner \hat{p}'_i \urcorner \\ S' \approx \llcorner S' \lrcorner \end{cases}$$

Ce faisant, commes les variables x_{ij} et δ_i^2 sont positives, toute solution qui satisfait l'inégalité pour ces poids approchés satisfait également l'inégalité pour les poids exacts.