

Rapport du miniprojet de Recherche opérationnelle

Groupe 9 : Nacim Bouia, Emma Beuze, Rémi Simon

1 Questions

1. Dans ce cas simplifié, on ne peut charger un camion qu'avec une pile d'emballage. Le flux d'emballage est alors égal au flux de camions et de plus, un camion ne pourra livrer qu'un fournisseur par route. Nous allons donc construire un graphe avec des sommets u_j (resp f_j), u_j^s (resp f_j^s) correspondant à la partie sans sur-coût de l'usine (resp fourni.) et à la partie sur-stock de l'usine (resp fourni.) au jour j . Nous introduisons également un sommet modélisant une centrale à cartons. Le principe est illustré figure 1.

2. Le problème d'optimisation (11) fait intervenir les variables s , x et z soit, (en considérant $|E|$ constant) en $O(J|F| + J|U| + JR_{sup})$ où R_{sup} est le max total de routes possibles : $R_{sup} = O((|U| + |F|)(|U| + |F| - 1)(|U| + |F| - 2)(|U| + |F| - 3))$. La PLNE qui dérive de ce problème d'optimisation rajoute $O(FJ + UJ)$ variables (termes en max). En tout, le nombre de variables est au plus polynomial en : $O(|F|J + |U|J + J(|U| + |F|)^4)$. C'est en pratique inextricable à cause du terme $O(J(|U| + |F|)^4)$.

3. Notations : e - emballage, k - camion, j - jour, v - sommet du graphe. Nous gardons les contraintes s et z du sujet. Nous introduisons 3 variables : a_{e,k,j,v_1,v_2} , quantité de e dans k à j transitant de v_1 à v_2 , ainsi que $b_{k,j,v_1,v_2,i}$, $i \leq 4$, variable binaire valant 1 si (v_1, v_2) est en i^e position de la tournée de k et $\delta_{j,k,i} \leq \max(0, n_{j,k} - i)$ variable binaire = 0 si $i \geq$ nombre d'arrêts de k le jour j , noté $n_{j,k} = \sum_{i,v_1,v_2} b_{j,k,v_1,v_2,i}$.

Il y a $(|U| + |F|)^2$ arcs dans le graphe, on en déduit qu'il y a au plus $|E|(|U| + |F|)^2 JK$ variables. Pour la fonction objectif, nous remplaçons la somme des $c_R x_R$ par :

$$c_{stop} \sum_{k,j} n_{k,j} + \sum_{v_1,v_2} \gamma d_{v_1,v_2} \sum_{i,k} b_{j,k,v_1,v_2,i} + c_{cam} \sum_{j,k,v_1,v_2} b_{k,j,v_1,v_2,1}$$

. Les autres termes restent inchangés (il faut linéariser les max, cf 2.2.4). Nous gardons les contraintes (1), (2), (6) et nous ajoutons les contraintes suivantes :

- Capacité du camion : $\sum_e a_{e,k,j,v_1,v_2} \leq L$ — Tournée f ne démarre pas¹ : $\sum_{f,v_2} b_{k,j,f,v_2,1} = 0$
- Au départ d'au plus un arc : $\sum_{v_2,i} b_{k,j,v,v_2,i} \leq 1$ — A l'arrivée d'au plus un arc : $\sum_{v_1,i} b_{k,j,v_1,v,i} \leq 1$
- u ne reçoit pas d'emballage : $\sum_{v_1,i} b_{k,j,v_1,u,i} = 0$
- Pas d'emballage transporté si le camion ne circule pas sur l'arc (v_1, v_2) : $a_{e,k,j,v_1,v_2} \leq L \sum_i b_{k,j,v_1,v_2,i}$
- Connexité (à linéariser par McCormick, produit variable binaire et variable bornée) :
 $\forall i \leq 3, \sum_{v_1} b_{k,j,v_1,v_2,i} \delta_{j,k,i+1} = \sum_{v_3} b_{k,j,v_2,v_3,i+1}$
- Expédition/Livraison : $z_{e,u,j}^- = \sum_{v_2,k} a_{e,k,j,u,v_2} - \sum_{v_1,k} a_{e,k,j,v_1,u}$ et $z_{e,f,j}^+ = \sum_{v_1,k} a_{e,k,j,v_1,f} - \sum_{v_2,k} a_{e,k,j,f,v_2}$

2 Algorithme

L'objectif central est de rendre le gros de la charge de calcul traitable automatiquement par solveur. Le 1.2. montre que le nombre de variables $(x_R)_R$ induites par la prise en compte de toutes les routes est prohibitif pour ce qui est des résolutions directes en machine.

1. Master Yoda, "A simple method to make reports fit the 4 page threshold", Springer-Verlag, 2020

Par ailleurs, les autres variables -uniquement considérées- restent intéressantes car elles se suffisent à elles même pour ce qui est de décrire la consommation et la production des fournisseurs. De plus leur nombre n'est pas prohibitif : $O(|F|J + |U|J)$.

Cette démarche induit une décomposition naturelle du problème principal. Une partie de gestion des stocks, et des quantités expédiées (resp. reçues) des usines (resp. des fournisseurs) traitable en machine, et une partie d'allocation de transports sur ce réseau de flux.

2.1 Calcul des flux par PL(NE)

2.1.1 Emballages et horizon temporel

L'allocation des flux est linéaire en les emballages. En effet sans restriction logistique, l'allocation des flux d'emballages sont des problèmes parallèles. De même, comme les trajets sont réalisés au jour le jour, le problème se décompose naturellement en travaillant à jour fixé et emballage fixé.

Pour réduire encore davantage le nombre de variables, on remarque que le problème est réductible en horizon temporel. En se plaçant à un jour donné, il est possible de ne considérer dans l'optimisation des flux que les offres et besoins futurs au terme d'un horizon réduit J_{red} . Le cas $J_{red} = 1$ correspondant au cas où les flux ne prennent pas en compte la situation du jour suivant. C'est à dire, considérant un jour j_0 donné, on considérera pour les variables, les indices de jours $j \in \llbracket j_0, j_0 + J_{red} - 1 \rrbracket \cap \llbracket 1, J \rrbracket$. On considérera par la suite que l'on travaille à e et j_0 fixé.

2.1.2 Fonction objectif

La fonction objectif du problème général doit être adaptée. Premièrement elle doit être linéarisée. Deuxièmement, la fonction objectif dans le cas réduit sans les variables $(x_R)_R$ doit être modifiée. Une façon de procéder serait d'assumer complètement le parti pris de la dissociation des problèmes de flux et des problèmes de logistique. Cela consiste à ne garder que les termes liés aux stocks usine, stocks fournisseur et excédents. Cela revient à prendre $c_R = 0$ dans la fonction objectif de l'énoncé. Cet angle d'attaque nous éloignerait cependant trop du problème et ferait exploser nos coûts routes.

Une meilleure approche serait d'anticiper la logistique et d'estimer les coûts de transport à priori. Cela nécessite d'estimer le coût de chargement qu'implique un flux d'emballage e au jour j de l'usine u au fournisseur f . On peut choisir pour cela, d'estimer le nombre de camions mobilisé par tas de e déplacé par $\frac{l_e}{L}$. On peut grossièrement suggérer qu'un flux de u à f induit une route de u à f . Ainsi le coût induit par le flux de u à f $y_{u,f}$ induit un coût logistique que l'on approxime par :

$$(c^{cam} + c^{stop} + \gamma d_{u,f}) \left(\frac{l_e}{L} y_{u,f} \right)$$

En réalité, ce coût approximé correspond au coût camion si l'on pouvait découper les camions en autant de parties que l'on souhaite. Ce nouveau problème étant moins contraignant que le problème réel, son coût associé propose un minorant du coût réel et ce malgré le fait qu'un camion "découpé" n'effectuera qu'un seul arrêt. Nous esquissons la preuve de cette propriété sur l'exemple suivant.

Afin de démontrer cette propriété, étudions le cas du graphe de la figure 2 en annexe et comparons le coût approximé dans le cas où les flux de deux fournisseurs f_1 et f_2 requièrent un demi-camion chacun partant de l'usine u et le cas où nous utilisons un seul camion rempli qui livrera d'abord f_1 puis f_2 .

La somme des coûts des 2 camions approximés est ici : $c_{app} = c^{cam} + c^{stop} + \gamma \frac{d_{u,f_1} + d_{u,f_2}}{2}$.

Le coût réel du camion rempli passant par f_1 puis f_2 est lui : $c_{reel} = c^{cam} + 2c^{stop} + \gamma(d_{u,f_1} + d_{f_1,f_2})$.

Finalement $c_{app} \leq c_{reel}$ par inégalités triangulaires et donc le coût approximé des deux flux est un minorant du coût réel. Ce cas particulier se généralise au cas général et nous emploierons cette propriété pour le calcul de la borne inférieure.

2.1.3 Variables

Les variables que l'on cherche principalement à déterminer dans le cas optimal sont **les flux** de e , de u à f au jour j : $y_{u,f,j}$. On utilise également s , z_+ et z_- introduites dans l'énoncé.

D'autres variables moins naturelles apparaissent de la **linéarisation** des fonctions *max*. On nomme ces variables $\alpha_{u,j}$, $\beta_{f,j}$, $\lambda_{f,j}$ et $\delta_{f,j}$ provenant respectivement des expressions (3), (7a), (7b), et (6) de l'énoncé.

2.1.4 Contraintes et fonction objective

On retrouve, en utilisation les notations de l'énoncé, les contraintes de stock 1., les flux liés aux émissions d'emballages 2. et les contraintes de linéarisation 3. :

$$\begin{aligned}
1. \quad & \forall u, j \quad s_{u,j} = s_{u,j-1} + b_{u,e,j-1}^+ - z_{u,j}^-, \quad s_{u,j} \geq 0 \\
& \forall f, j \quad s_{f,j} = \delta_{f,j} + z_{f,j}^+, \quad s_{f,j} \geq 0 \\
2. \quad & \forall u, j \quad z_{u,j}^- = \sum_f y_{u,f,j}, \quad z_{u,j}^- \geq 0 \\
& \forall f, j \quad z_{f,j}^+ = \sum_u y_{u,f,j}, \quad z_{f,j}^+ \geq 0 \\
3. \quad & \forall u, j \quad \alpha_{u,j} \geq s_{u,j} - r_{u,e,j}, \quad \alpha_{u,j} \geq 0 \quad (3) \\
& \forall f, j \quad \delta_{f,j} \geq s_{f,j-1} - b_{f,e,j}^-, \quad \delta_{f,j} \geq 0 \quad (6) \\
& \forall f, j \quad \gamma_{f,j} \geq b_{f,e,j}^- - s_{f,j-1}, \quad \lambda_{f,j} \geq 0 \quad (7a) \\
& \forall f, j \quad \beta_{f,j} \geq s_{f,j} - r_{f,e,j}, \quad \beta_{f,j} \geq 0 \quad (7b)
\end{aligned}$$

La fonction objective prendra alors elle la forme :

$$\sum_{u,f,j} (c^{cam} + c^{stop} + \gamma_{d_{u,f}}) \left(\frac{l_e}{L} y_{u,f,j} \right) + \sum_{u,j} c_{e,u}^s \alpha_{u,j} + \sum_{f,j} c_{e,f}^s \beta_{f,j} + \sum_{f,j} c_{e,f}^{exc} \lambda_{f,j}$$

2.1.5 Programmation Linéaire

Le problème se résout a priori en nombres entiers. Cependant, sa formulation est parfaite. Nous allons par conséquent pouvoir le relâcher, gagner un précieux temps de calcul tout en conservant des solutions entières. Cela nous a permis en particulier de résoudre le problème sur tout l'horizon temporel : $J_{red} = J$.

Nous résolvons ce problème d'optimisation linéaire avec le solveur Gurobi. Cela fixe alors les flux d'emballage chaque jour, entre chaque couple usine-fournisseur. Par conséquent, les stocks des usines et fournisseurs sont déjà déterminés : on a donc déjà accès à leur coût final. Cette approche nous permet également d'obtenir un minorant du coût de notre problème général puisque notre modélisation de coût de transport nous fait résoudre en réalité un problème moins contraignant. Notre PL permet ainsi d'obtenir ces premiers résultats :

Coût usine total	Coût fournisseur total	Minorant du coût total	Temps de résolution du PL
1 287	1 908 268	2 861 810	9 min 03 s

Les transferts d'emballages étant dorénavant fixés, il nous reste à construire les routes. Ce nouveau problème se divise en la résolution de chaque sous-problème par jour et par usine : il faut affecter les camions partant de u le jour j pour livrer ses fournisseurs désignés par la PL : $\{f \in F ; \exists e \in E; y_{eufj} > 0\}$.

2.2 Clustering des fournisseurs à jour et usine donné

Afin d'optimiser nos coûts de transports, nous allons diviser le problème spatialement par la mise en place d'un clustering géographique regroupant des fournisseurs proches. Cette méthode permettra par la suite de construire des camions livrant uniquement les fournisseurs d'un même cluster. Ce partitionnement doit être avantageux car, s'il est bien construit, les camions auront peu de distance à effectuer lors des livraisons intra-fournisseurs et le coût kilométrique sera très fortement réduit. Notre algorithme de partitionnement est illustré figure 3.

Algorithm 1 Clustering en groupe de n_f fournisseurs pour une usine u et un jour j

Require: L la liste des fournisseurs devant être livrés par u à j

Ensure: Clustering C de L en groupe de n_f fournisseurs

while $L \neq \emptyset$, il reste des fournisseurs **do**

$f \leftarrow \operatorname{argmax}_{f' \in L} \{d(u, f')\}$ le plus lointain fournisseur de u , ajoute f dans un nouveau cluster c' de C , et retire f de L

for i from 1 to $n_f - 1$ and $L \neq \emptyset$ **do**

$f' \leftarrow \operatorname{argmin}_{f' \in L} \{d(f, f')\}$ le plus proche fournisseur de f , ajoute f' dans c' et retire f' de L

end for

end while

return C

2.3 Construction des camions

Afin d'évaluer la qualité de notre partitionnement, nous devons implémenter un algorithme permettant la création de camions. Pour ce faire, dans un cluster donné, nous traitons les fournisseurs un par un. Pour chaque fournisseur, nous remplissons le premier camion possible par taille décroissante d'emballages (inspiration du first-fit decreasing). Lorsque toutes les livraisons d'un fournisseur ont été traitées, on passe au fournisseur suivant tandis que lorsque le camion est rempli, on en crée un nouveau.

2.4 Taille de cluster optimale

Afin de choisir une taille de cluster optimale, nous avons tracé le coût route en fonction de la taille des clusters (Annexe-Figure 4) pour l'algorithme de chargement des camions précédent. On note tout d'abord que les multiples de 4 se démarquent : c'est parfaitement normal puisque 4 est le nombre maximal de fournisseurs dans camion. Parmi ces multiples, la taille de cluster minimale (égale à 4) est la meilleure. Cela pourrait paraître étonnant puisque si les flux partants de l'usine pour un cluster sont faibles, on risque de construire un seul camion qui sera peu rempli. On pourrait alors penser qu'un clustering plus grand (8,12,16 ...) serait plus prudent afin d'avoir des flux plus homogènes, mais le prix à payer : trajet de camions plus long ; se révèle finalement être plus élevé que le gain en nombre de camions, nous privilégierons donc avoir des camions relativement moins remplis et donc de regrouper les fournisseurs par 4.

Une fois la méthode de construction des camions par cluster effectuée, il s'agit d'affecter l'ordre de livraison des fournisseurs pour chaque camion. La particularité du problème ($n \leq 4$ fournisseurs) nous permet de tester toutes les configurations possibles ($4!=24$ au maximum) et de finalement conserver la route de coût minimale. Appliquant cela à toutes nos routes, nous avons accès au coût de transport total :

Coût route	Coût camion fixe	Coût arrêt	Coût distance	Temps d'exécution (Clustering(s) + Routes)
2 782 931	539 550	472 200	1 771 281	3 min 35 s

2.5 Conclusion

En mettant nos résultats obtenus sur les flux en lien avec notre processus de construction des routes, notre méthode de résolution par PL puis clustering nous permet d'obtenir ces résultats :

Coût total	Meilleur minorant	Coût naïf	Temps d'exécution total
4 692 486	2 861 810	41 777 761	12 min 38 s

On remarque alors que notre solution a nettement fait diminuer le coût total en comparaison au coût naïf (aucune route) et que nous nous sommes grandement rapproché de notre minorant obtenu par notre problème linéaire de la partie 2.1. Ce minorant n'est d'ailleurs probablement pas une borne inférieure à notre problème au sens mathématiques et cette borne, est très probablement bien plus proche de notre coût total final.

En ce qui concerne les améliorations possibles, il nous semble difficile de progresser sur notre approche des flux car elle nous offre une solution exacte d'un problème très proche du problème initial. En revanche, notre approche sur la construction des routes et le clustering est loin d'être optimale, d'autres groupes ont par ailleurs de meilleurs résultats sur les routes en partant des même flux. Cependant, cette approche est très rapide et propose un coût honnête.

Nous avons toutefois essayé d'autres stratégies sur les routes. L'une d'elle consiste à faire de la recherche locale sur la liste de priorité des fournisseurs renseignée pour le remplissage des camions sans utiliser de clustering. Ici, l'accès au coût engendré par ces modifications était trop gourmand en complexité : génération de nouveaux camions, conversion en routes, puis calcul du coût des routes. Nous ne pouvions donc pas faire un nombre d'itérations suffisant pour à la fois intensifier et diversifier notre métaheuristique de voisinage. Une recherche locale sur les routes aurait été possible mais cela aurait risqué de déplacer certains flux qui avaient préalablement été calculés pour être optimaux. En ce sens, nous avons donc opté pour ne pas modifier ces flux. Nous avons donc moins de flexibilité dans notre construction des routes mais plus de garantie sur les coûts de stockage.

3 Annexe

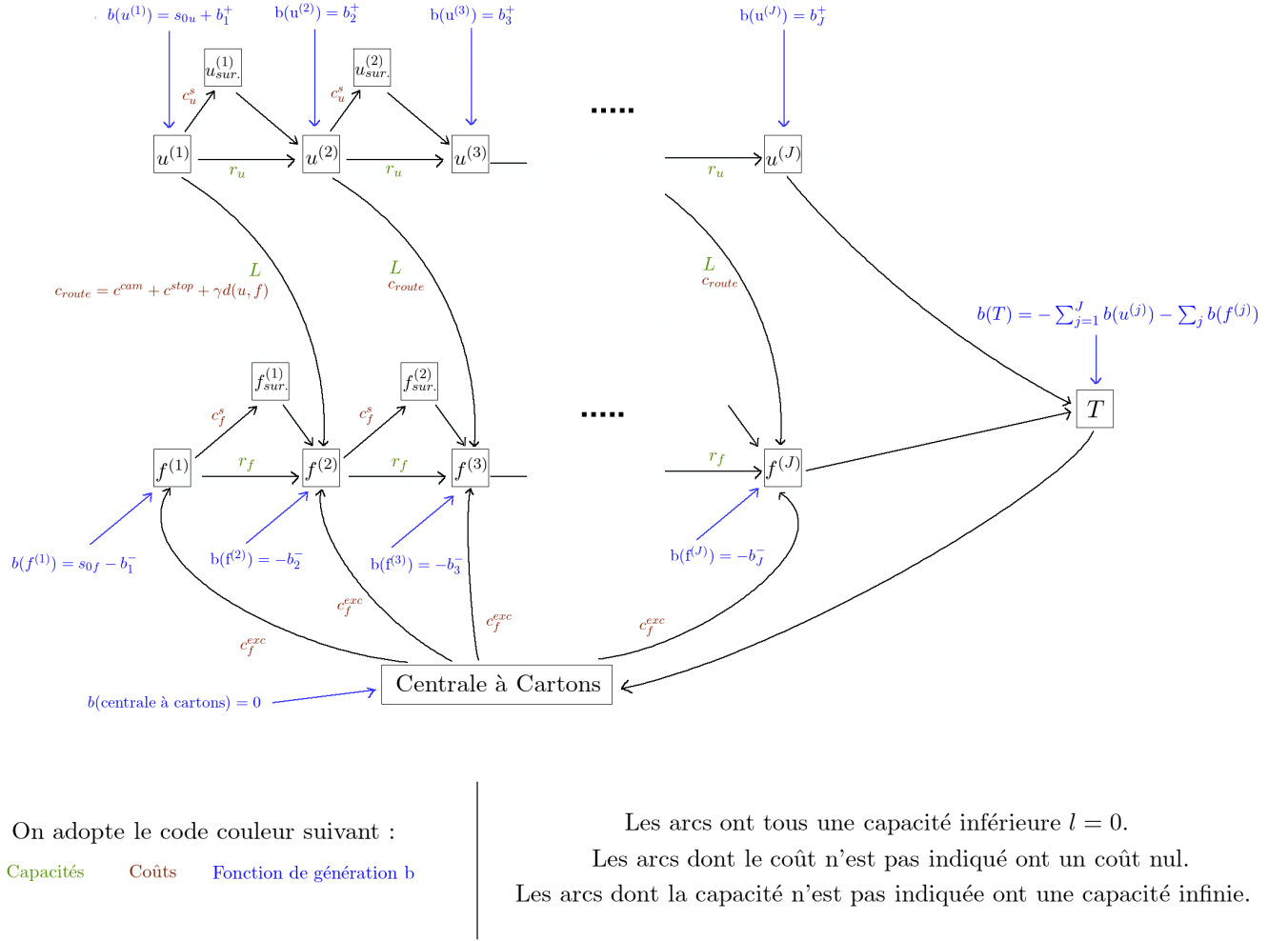


FIGURE 1 – Modélisation en problème de b-flot de la question pour un unique emballage e tel que $l_e \geq L$

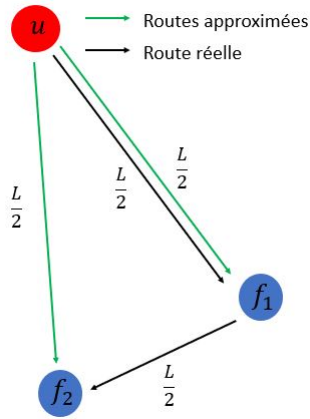


FIGURE 2 – Preuve du coût du paragraphe 2.2.2

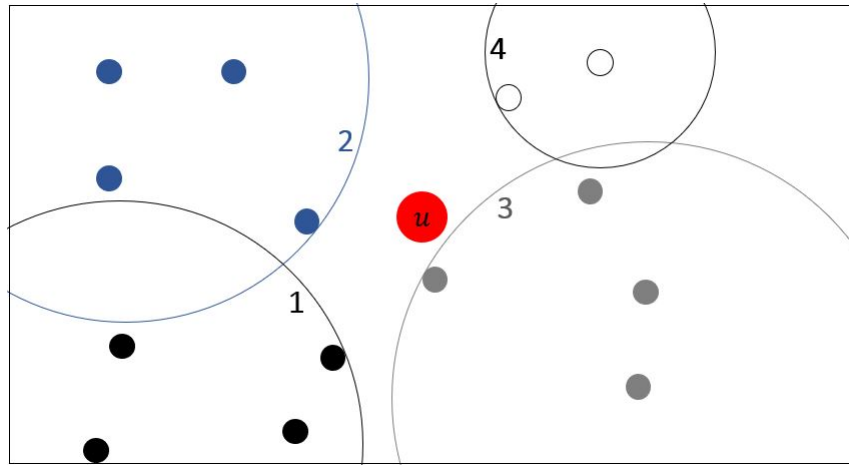


FIGURE 3 – Schéma de principe du clustering géographique utilisé pour des groupes de 4 fournisseurs, le point rouge représente l'usine, les autres points les fournisseurs devant être livré depuis l'usine

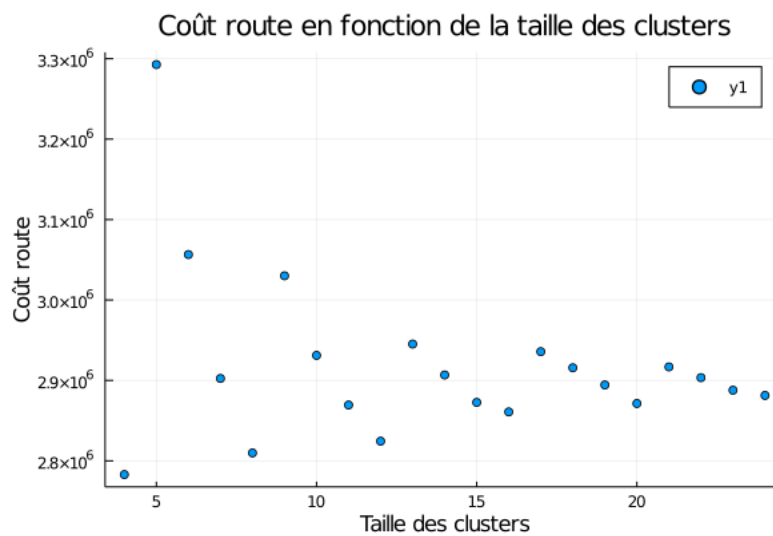


FIGURE 4 – Les multiples de 4 sont des minimums locaux, 4 est un minium global