

Algorithmique et complexité Projet Partie II : Remplissage d'un réservoir

Romain Duc

2 janvier 2022

1 Introduction

Ce projet a été réalisé par Romain Duc en première année de Master Informatique de Nancy.

L'objectif de ce projet est d'implémenter en programmation dynamique l'algorithme de l'optimisation du nombre de pleins suivant un itinéraire avec un ensemble de stations et choisir de s'y arrêter ou non.

2 optimisation du problème des stations-essence

2.1

Montrer que le problème (P1) : un départ de s avec μ_s unités dans le réservoir, et le problème (P2) : un départ de s avec un réservoir vide sont équivalents. Pour la démonstration, on vous propose d'ajouter un nouveau noeud s' relié directement à s . Précisez le coût de l'essence au sommet s' et la distance $d(s', s)$, puis trouver, à l'optimum, le lien entre le nombre d'arrêts dans (P1) et le nombre d'arrêts dans (P2)

Réponse :

Pour montrer que les problèmes (P1), et (P2) sont équivalents, on peut ajouter un noeud s' qui est directement relié à s . Le coût de l'essence au noeud s' doit être égal au coût d'essence au noeud s , et la distance $d(s', s)$ doit être égale à la quantité d'essence μ_s initialement dans le réservoir.

Pour trouver la solution optimale au problème (P1) avec cette configuration, on peut d'abord trouver la solution optimale au problème (P2) en commençant au noeud s' avec un réservoir vide. Si la solution optimale au problème (P2) implique de faire K arrêts à des stations, alors on peut simplement soustraire K au nombre d'arrêts dans la solution optimale au problème (P1).

Par exemple, supposons que la solution au problème (P2) implique d'effectuer 3 arrêts à des stations. Cela signifie qu'on a besoin de remplir le réservoir 3 fois afin d'atteindre notre destination depuis le noeud s' . Si on commence au noeud s avec μ_s unités d'essence dans le réservoir, alors on a seulement besoin de faire $K - 3 = 0$ arrêts aux stations-service afin d'atteindre notre destination.

D'où, on peut conclure que le nombre d'arrêts dans la solution optimale au problème (P1) est égale au nombre d'arrêts dans la solution optimale au problème (P2) moins K . Cela démontre bien que les problèmes (P1) et (P2) sont équivalents.

2.2

Montrer que la stratégie optimale consiste à arriver à destination t avec un réservoir vide.

Réponse :

Pour montrer que la stratégie optimale consiste à arriver à destination t avec un réservoir vide, nous allons utiliser un raisonnement par l'absurde.

Supposons que la stratégie optimale consiste à arriver à destination t avec une quantité d'essence q ($q > 0$) dans le réservoir. Nous allons maintenant montrer que cette stratégie n'est pas optimale, ce qui contredit notre hypothèse initiale.

Puisque nous avons une quantité q d'essence dans le réservoir en arrivant à destination t , nous pouvons continuer notre voyage de t à un autre nœud v à une distance de q km, en utilisant l'essence dans le réservoir. Nous devons donc acheter de l'essence à v pour poursuivre notre voyage.

Si nous avons utilisé au maximum K arrêts pour remplir le réservoir, nous avons encore un arrêt disponible pour remplir le réservoir à v . Si nous remplissons le réservoir à v , nous avons maintenant une quantité U d'essence dans le réservoir et nous pouvons continuer notre voyage jusqu'à un autre nœud w à une distance de U km. Nous devons donc acheter de l'essence à w pour poursuivre notre voyage.

Nous pouvons continuer ce processus jusqu'à ce que nous atteignons notre destination finale. Cela signifie que nous avons utilisé un arrêt de plus pour remplir le réservoir par rapport à notre stratégie initiale, qui consistait à arriver à destination t avec un réservoir vide. Cela signifie que notre stratégie initiale est plus avantageuse que la stratégie consistant à arriver à destination t avec une quantité d'essence q dans le réservoir, ce qui contredit notre hypothèse initiale selon laquelle cette stratégie était optimale.

Par conséquent, la stratégie optimale consiste à arriver à destination t avec un réservoir vide.

2.3

Exprimer le coût de la solution optimale en fonction de l'état final du programme dynamique en fixant les valeurs de u , q et g dans la fonction coût $C[u, q, g]$.

Réponse :

Pour trouver la solution au problème des stations-service en utilisant la programmation dynamique, on peut utiliser la fonction de coût $C[u, q, g]$ définie comme le coût minimal d'un parcours du nœud u au nœud t s'arrêtant q

fois pour remplir le réservoir, sachant que l'on arrive au noeud u avec g unités d'essence (km).

Pour exprimer le coût de la solution optimale en termes d'état final de l'algorithme de programmation dynamique, nous pouvons fixer les valeurs de u , q et g dans la fonction de coût $C[u, q, g]$. Par exemple, si nous voulons trouver la solution optimale au problème de remplissage aux stations-service débutant au noeud s et s'achevant au noeud t avec 3 arrêts pour remplir le réservoir, on peut affecter les variables comme suit :

- $u = t$;
- $q = 3$;
- $g = 0$; dans la fonction de coût $C[u, q, g]$.

Cela va nous donner le coût minimal d'un trajet de s à t utilisant 3 arrêts pour remplir le réservoir, sachant que nous arrivons au noeud t avec 0 unité d'essence dans le réservoir.

En variante, nous pouvons fixer les valeurs de u , q et g dans la fonction de coût pour trouver le coût de la solution optimale à différentes étapes du voyage. Par exemple, si nous voulons trouver le coût de la solution optimale au noeud u avec 2 arrêts pour remplir le réservoir, et avec g unités d'essence dans le réservoir, on peut affecter comme suit :

- $q = 2$;
- $u = u$ et $g = g$; dans la fonction $C[u, q, g]$.

Cela va nous donner le coût minimal d'un trajet de s à t utilisant 2 arrêts pour remplir le réservoir, sachant que nous sommes actuellement au noeud u avec g unités d'essence dans le réservoir.

Plus généralement, le coût de la solution optimale peut être exprimée en termes d'état final de l'algorithme de programmation dynamique par la fixation des valeurs u , q et g dans la fonction de coût $C[u, q, g]$.

2.4

Donner l'expression de la récurrence de $C[u, q, g]$ en fonction de $C[v, q-1, h]$.

Réponse :

La fonction de coût $C[u, q, g]$ peut être définie en utilisant l'expression de récurrence suivante :

$$C[u, q, g] = \min(C[v, q-1, h] + d(u, v) + c(v) * (U - g)), \forall v \in V$$

Cette relation de récurrence établit que le coût d'un voyage du noeud u au noeud t utilisant q arrêts pour remplir le réservoir et arriver au noeud u avec g unités d'essence est égal au coût minimum d'un trajet du noeud v au noeud t utilisant $q-1$ arrêts pour remplir le réservoir et arriver au noeud

v avec h unités d'essence, plus le coût de voyage de u à v et le coût de remplissage du réservoir au noeud v . Le minimum est pris parmi tous les noeuds $v \in V$ dans le graphe G .

Cette relation nous permet de calculer le coût de la solution optimale au "problème de ravitaillement à des stations-service" utilisant la programmation dynamique. En appliquant alternativement cette relation de récurrence, nous pouvons calculer le coût de la solution optimale à chaque noeud u et pour chaque nombre d'arrêts q et chaque quantité d'essence g . Cela nous permet de déterminer le coût minimal d'un trajet de s à t utilisant q arrêts pour remplir le réservoir et arriver au noeud t avec 0 unité d'essence.

2.5

Proposer une initialisation de la fonction coût $C[u, q, g]$

Réponse :

Pour initialiser la fonction $C[u, q, g]$ en utilisant la prog. dynamique, on peut affecter la valeur de $C[u, q, g]$ à $l'\infty$ pour tous les noeuds u du graphe et toutes les valeurs de q et g . Cela garantit que nous n'utilisons pas accidentellement une valeur de $C[u, q, g]$ non initialisée lors du calcul de la solution optimale au problème des stations services.

On peut alors définir la valeur de $C[s, 0, 0]$ à 0, puisque le coût minimum d'un trajet de s à t utilisant 0 arrêts pour remplir le réservoir et arriver à s avec 0 unités d'essence est 0.

Pour les autres valeurs de u, q et g , on peut définir la valeur de $C[u, q, g]$ comme le coût minimal d'un trajet de u à t utilisant Q arrêts pour remplir le réservoir et arriver à u avec g unités d'essence. Cela peut être calculé en utilisant la relation de récurrence de la Question 4.

Plus généralement, les valeurs initiales de la fonction de coût $C[u, q, g]$ peuvent être définies comme suit :

- $C[u, q, g] = \infty$, $\forall u \in V$ et $\forall q, \forall g$
- $C[s, 0, 0] = 0$
- $C[u, q, g] = \min(C[v, q-1, h] + d(u, v) + c(v) * (U - g))$, $\forall u \in V, q > 0, g > 0$, et $\forall v \in V$

Cette initialisation garantit que nous avons un point de départ valable pour la programmation dynamique.

2.6

Donner la complexité de ce programme dynamique

Réponse :

La fonction coût $C[u, q, g]$ dépend de trois variables : le nœud u , le nombre q d'arrêts pour remplir le réservoir et la quantité g d'essence dans le réservoir. Pour chaque nœud u et chaque valeur de q et g , nous devons calculer le coût minimum du voyage pour aller de u à t en utilisant au plus q arrêts pour remplir le réservoir et en arrivant à u avec g unités d'essence.

*Il y a n nœuds dans le graphe, U valeurs possibles pour la quantité d'essence dans le réservoir et K valeurs possibles pour le nombre d'arrêts pour remplir le réservoir. Nous devons donc calculer la fonction coût $C[u, q, g]$ pour toutes les combinaisons possibles de u , q et g , soit $n * U * K$ combinaisons.*

Par conséquent, la fonction de coût C a une complexité de $O(nKU)$. Pour calculer $C[u, q, g]$ pour tous les nœuds u , tous les arrêts possibles q et toutes les quantités possibles d'essence g , nous devons effectuer $O(nKU)$ itérations.

2.7

Expliquer pourquoi même avec la discrétisation des valeurs de g l'utilisation de l'état $C[u, q, g]$ mène à un programme dynamique de complexité pseudo-polynomial.

Réponse :

Une façon d'éviter le besoin de considérer toutes les valeurs possibles de g lorsque l'on utilise la prog. dynamique pour résoudre le problème de remplissage aux stations, est de discrétiser les valeurs que g peut prendre. Cela signifie qu'au lieu de considérer toutes les valeurs possibles de g dans l'intervalle $[0, U]$, on considère uniquement un ensemble discret de valeurs pour g .

Par exemple, on pourrait diviser la plage $[0, U]$ en un ensemble de N intervalles égaux, et ne considérer que les valeurs de g qui sont contenues dans ces intervalles. Cela peut permettre de réduire le nombre de valeurs possibles de g que nous devons tenir compte lorsque l'on utilise la prog. dynamique, rendant ainsi l'algorithme plus efficace.

Cependant, même avec la discrétisation des valeurs de g , l'utilisation de l'état $C[u, q, g]$ en prog. dynamique mène à un algorithme avec une complexité pseudo-polynomiale. Cela s'explique par le fait que le nombre de

valeurs possibles de u , q et g demeure exponentiel en la taille de l'entrée, et la complexité en temps de l'algorithme est dépendante du nombre de ces valeurs dont il faut tenir compte. La discrétisation des valeurs de g ne modifie pas la dépendance du programme envers la taille du réservoir U et le nombre de nœuds V dans le graphe.

Par exemple, si nous utilisons N intervalles pour discrétiser les valeurs de g , alors le nombre de valeurs possibles de g est $O(N)$. Le nombre de valeurs possibles de u et q est aussi exponentiel en la taille de l'entrée, ainsi le nombre total d'états dans l'algorithme de prog. dynamique est $O(N * 2^n)$, où n est la taille de l'entrée. C'est une complexité pseudo-polynomiale, car la complexité en temps de l'algorithme est dépendante à la fois de la taille de l'entrée et de la taille des entiers utilisés pour représenter l'entrée.

2.8

Montrer que la stratégie suivante est optimale pour décider de la quantité d'essence à remplir à chaque arrêt.

1. A l'arrêt u_l , remplir juste assez d'essence pour atteindre t avec un réservoir vide ;
2. Pour $j < l$
 - i) Si $c(u_j) < c(u_{j+1})$, alors à u_j faire le plein
 - ii) Si $c(u_j) \geq c(u_{j+1})$, alors à u_j mettre juste assez d'essence pour atteindre u_{j+1}

Réponse :

Pour prouver que la stratégie décrite ci-dessus est optimale pour décider de la quantité d'essence à remplir à chaque arrêt, nous pouvons utiliser une récurrence.

Premièrement, considérons le cas de base où $l = 1$, c'est-à-dire qu'il n'y a qu'un arrêt de remplissage du réservoir. Dans ce cas, la stratégie optimale est de remplir le réservoir à l'arrêt u_l juste suffisamment pour atteindre t avec un réservoir vide. C'est parce que, par définition, u_l est le dernier arrêt dans la solution optimale, que nous voulons être sûrs que nous avons assez d'essence pour atteindre t sans qu'il soit nécessaire d'effectuer aucun autre arrêt.

Maintenant, considérons le cas où $l > 1$, et supposons que la stratégie optimale pour décider de la quantité d'essence à remplir à chaque arrêt lorsqu'il y a $(l - 1)$ arrêts est de :

(i) remplir le réservoir à l'arrêt u_{l-1} juste suffisamment pour atteindre u_l , si $c(u_{l-1}) \geq c(u_l)$

ou de

(ii) remplir le réservoir à l'arrêt u_{l-1} à la capacité maximale si $c(u_{l-1}) < c(u_l)$

Nous allons démontrer que cette hypothèse est valable également pour l .

Si $c(u_l) < c(u_{l+1})$, alors la stratégie optimale est de remplir le réservoir à l'arrêt u_l à la capacité maximale. En effet, remplir le réservoir à la capacité maximale à u_l coûtera moins que de le remplir à une capacité moindre et ensuite le remplir à nouveau à u_{l+1} , où le prix de l'essence est plus élevé.

Si $c(u_l) \geq c(u_{l+1})$, alors la stratégie optimale est de remplir le réservoir à l'arrêt u_l juste suffisamment pour atteindre u_{l+1} . En effet, remplir le réservoir à une capacité moindre à l'arrêt u_l coûtera moins que de le remplir à la capacité maximale, et nous aurons toujours assez d'essence pour atteindre u_{l+1} .

Ce fonctionnement permet de faire le plein lorsque le coût est le plus bas, et donc de potentiellement économiser sur l'achat d'essence sur le(s) prochain(s) arrêts lorsque les prix de ces derniers sont plus élevés.

Par conséquent, par le principe de récurrence, nous avons montré que la stratégie décrite ci-dessus est optimale pour décider de la quantité d'essence à remplir à chaque arrêt du trajet.

2.9

Expliquer votre compréhension des formules de récurrence (initialisation, récurrence, solution optimale).

Réponse :

La stratégie optimale décrite ci-dessus vise à trouver le chemin le moins coûteux pour atteindre la destination finale t en faisant au plus l arrêts de remplissage d'essence. Pour chaque arrêt u , la stratégie optimale détermine la quantité d'essence à mettre dans le réservoir en fonction de la position précédente w et du coût de l'essence à l'arrêt u .

La formule d'initialisation définit la valeur de $C[u, 1, g]$ pour le premier arrêt de remplissage u . Si g est inférieur ou égal à la distance entre u et t et la distance est inférieure ou égale à la capacité maximale du réservoir U , alors $C[u, 1, g]$ est égal à la distance entre u et t moins g multiplié par le coût de l'essence à l'arrêt u . Sinon, $C[u, 1, g]$ est égal à ∞ , ce qui signifie qu'il n'est pas possible de remplir suffisamment d'essence pour atteindre la destination finale en faisant un seul arrêt de remplissage.

La formule de récurrence définit la valeur de $C[u, q, g]$ pour un arrêt de remplissage u en fonction des arrêts de remplissage précédents v et de la quantité d'essence g dans le réservoir lors de l'arrivée à u . Si le coût de l'essence à l'arrêt v est inférieur ou égal au coût de l'essence à l'arrêt u , alors $C[u, q, g]$ est égal au minimum de $C[v, q - 1, 0]$ plus la distance entre u et v moins g multipliée par le coût de l'essence à l'arrêt u , où v est un arrêt de remplissage tel que la distance entre u et v est inférieure ou égale à U . Si le coût de l'essence à l'arrêt v est supérieur au coût de l'essence à l'arrêt u , alors $C[u, q, g]$ est égal au minimum de $C[v, q - 1, U - d(u, v)]$ plus U moins g multiplié par le coût de l'essence à l'arrêt u , où v est un arrêt de remplissage tel que la distance entre u et v est inférieure ou égale à U .

La formule de solution optimale indique que la solution optimale est le minimum de $C[s, l, 0]$ pour l allant de 1 à k , où s est le point de départ et k est le nombre maximum d'arrêts de remplissage autorisés. Cela signifie que la solution optimale est le coût minimum pour atteindre la destination finale t à partir du départ s en faisant au plus l arrêts de remplissage, avec un réservoir vide.

En résumé, cette formule de récurrence utilise une approche de programmation dynamique pour résoudre le problème de station-service en utilisant des valeurs précédemment calculées pour trouver la solution optimale. Elle s'appuie sur l'idée que le coût minimal pour atteindre une ville u à partir d'une ville précédemment visitée v dépend de la quantité d'essence restante dans le réservoir et du coût de l'essence dans chaque ville. Grâce à ces informations, elle trouve une solution optimale en étudiant chacune des villes et en sélectionnant la valeur la plus basse parmi les différentes options de remplissage du réservoir.