

Contrôle continu

Algorithme de Dijkstra, une première approche

aucun document autorisé

1 Arborescence des plus courts chemins

1.1 Définition

Données

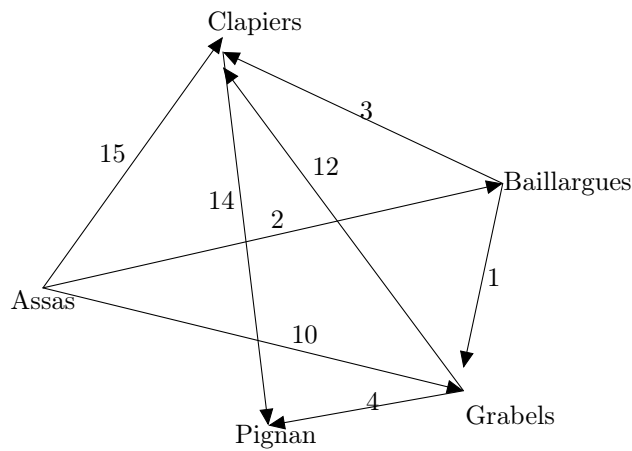
- Un graphe orienté $G = (X, U)$
- un sommet $s_0 \in X$ tel que de s_0 on puisse atteindre tout sommet de X par un chemin (orienté) dans G
- une fonction de coût $\omega : U \rightarrow \mathbb{N}$ (il s'agit du coût de chaque arc de U).

Résultat

- une arborescence orientée $T = (X, E)$ (avec $E \subseteq U$) de racine s_0
Cette arborescence est conservée grâce à un tableau $Pere$, indicé par les sommets de X , tel que $\forall x \in X$ (avec $x \neq s_0$), $Pere[x]$ est le père de x dans l'arborescence T ($Pere[s_0]$ n'est pas défini).
- une fonction de distance $\delta : X \rightarrow \mathbb{N}$ (il s'agit de la distance de s_0 au sommet x) telle que $\forall x \in X$, $\delta(x)$ soit à la fois
 - le coût de l'unique chemin de s_0 à x dans l'arborescence T
 - le coût minimal d'un chemin de s_0 à x dans le graphe G .Cette distance δ est conservée dans un tableau $Distance$ indicé par le sommets de X , tel que $\forall x \in X$ $Distance[x] = \delta(x)$

On notera Γ la fonction de x dans $\mathcal{P}(X)$ qui à chaque sommet fait correspondre l'ensemble de ses successeurs.

Exemple Soit le réseau de liaisons aériennes suivant, avec le prix (en milliers d'euros) de chaque trajet :



Question 1 *Pour vous faire comprendre le problème et les notations ; répondre sur la feuille de réponses, p. 1.*

- En indiquant la valeur de $\Gamma(\text{Grabels})$ compléter le tableau Γ , qui donne la fonction Γ .
- Pour savoir combien un homme d'affaire soucieux de voyager en avion (mais économe) partant d'Assas devra dépenser au minimum pour aller dans chacune des autres villes, compléter le tableau *Distance*.
- Compléter le tableau *Pere*, qui pour chaque ville, en indiquant l'étape précédente de celle de Pignan de cet homme d'affaire économe qui est parti d'Assas.
- Dans cet exemple qu'est ce que X , U , ω , E ? (vous pouvez donner à votre choix une réponse en extension ou en compréhension).
Vous pouvez, si vous le désirez, désigner chaque ville par son initiale, et chaque liaison aérienne par l'initiale de son origine et celle de son extrémité : ainsi CP représentera la liaison aérienne de Clapier à Pignan.

2 Algorithme de Dijkstra

Données:

- Un graphe orienté $G = (X, U)$
- un sommet $s_0 \in X$ tel que de s_0 on puisse atteindre tout sommet de X par un chemin (orienté) dans G
- une fonction de coût $\omega : X \times X \rightarrow \mathbb{N} : \omega(x, y)$ est le coût de l'arc $x \rightarrow y$.

Résultat:

- une arborescence orientée $T = (X, E)$ (avec $E \subseteq U$), de racine s_0 , conservée grace au tableau $Pere$ indicé par les sommets de X .
- un tableau $Distance$ indicé par les sommets de X .

pour $x \in X$ **faire** $Distance[x] \leftarrow +\infty$;

$Distance[s_0] \leftarrow 0$;

pour i de 2 à $|X|$ **faire**

choisir un sommet x tel que $Distance[x]$ soit minimum et qui n'a pas encore été choisi;

pour $y \in \Gamma(x)$ **faire**

si $Distance[y] > Distance[x] + \omega(x, y)$ **alors**

$Distance[y] \leftarrow Distance[x] + \omega(x, y)$;

$Pere[y] \leftarrow x$

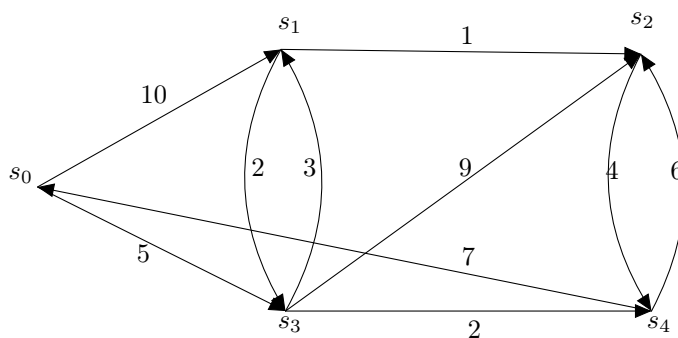
fin

fin

fin

2.1 trace de l'algorithme

Soit le graphe $G = (X, U)$



et la fonction ω de coût des arcs indiqué sur chaque arc

Question 2

En remplissant le rectangle du bas de la feuille de réponses p.2, continuer la trace de l'application de l'algorithme de Dijkstra sur le graphe ci dessus.

Il faudrait prouver qu'à la sortie de cet algorithme *Pere* représente une arborescence des plus courts chemins.
Ce n'est pas le but de ce contrôle, nous **admettrons** ce résultat.

2.2 Complexité de l'algorithme : approche naïve

On notera par n le nombre de sommets et par m le nombre d'arcs de G , et l'on établira tous les résultats demandé en fonction de l'un de ces deux (ou de ces deux) paramètres.
On considèrera que l'on a toujours $m \geq n$.

Question 3

Une fois que x a été choisi, combien de fois au plus passe-t-on dans la répétitive **Pour** $y \in \Gamma(x)$ **faire** ... ?

Question 4

On demande des classes Θ de complexité, pas des classes \mathcal{O} .
A quelle classe de complexité appartiennent les coûts

1. de l'exécution de *un pas* de la répétitive **Pour** $y \in \Gamma(x)$ **faire** ?
2. de l'exécution de *toute* la répétitive **Pour** $y \in \Gamma(x)$ **faire**
3. de l'exécution de *un pas* de la répétitive **Pour** i de 2 à $|X|$ **faire** ? (justifier)
4. de l'exécution de *toute* la répétitive **Pour** i de 2 à $|X|$ **faire** ?

2.3 Utilisation d'une file de priorité

Reprenons l'algorithme :

```
pour  $x \in X$  faire  $Distance[x] \leftarrow +\infty$ ;  
 $Distance[s_0] \leftarrow 0$ ;  
pour  $i$  de 2 à  $|X|$  faire  
    | choisir un sommet  $x$  tel que  $Distance[x]$  soit minimum et qui n'a pas encore été choisi ;  
    | pour  $y \in \Gamma(x)$  faire Modifier le cas échéant la valeur  $Distance[y]$  ;  
fin
```

On veut gérer le tableau *distance* grâce à un tas.

Question 5

1. Quel sera le contenu de chaque élément du tas ?
2. Quel est le nombre maximum d'éléments que peut contenir le tas ?

Question 6

A quelle classe de complexité appartient la partie *initialisation* de cet algorithme ? (une justification **concise** sera appréciée).

On supposera (sans justification dans un premier temps) que l'accès par son nom¹ à un sommet du tas se fait en $\Theta(1)$.

Pour analyser cet algorithme, on va utiliser une technique de **complexité amortie**.

Question 7

1. quel est le coût de choisir **un** sommet qui n'a pas encore été choisi ?
(une justification **concise** sera appréciée).
2. Quel est le coût global de choisir successivement tous les sommets ?

Question 8

Au cours de l'exécution de cet algorithme

1. combien de fois va-t-on examiner chaque arc du graphe ?
2. Quel va être le coût de **chaque** modification de la priorité d'un élément du tas ?
3. Quel va être le coût global de toutes les modifications de priorité ?

Question 9

L'utilisation d'un tas va-t-elle faire baisser la complexité

- en $\Theta((m + n)\log n)$?
- en $\Theta(m + n\log n)$?
- En $\Theta(n + m\log n)$?
- une autre ?

Question 10

Dans quel cas cette utilisation d'un tas n'est-elle pas intéressante ?

Question 11

Expliquez comment l'accès par son nom à un sommet du tas peut-il se faire en $\Theta(1)$. Préciser quelles seront les opérations de maintenance à prévoir et montrer que la complexité de l'algorithme n'en sera pas affectée.

1. Grabels ou Pignan