

Chapitre 3

GRAPHES : PLUS COURT CHEMIN

I	GRAPHE PONDÉRÉ	179
1	définition	179
2	longueur d'un chemin	179
II	ALGORITHME DE DIJKSTRA	179
	EXERCICES	182

La recherche du meilleur itinéraire que ce soit en distance, en temps ou en coût d'un point à un autre peut être modélisée par la recherche du plus court chemin dans un graphe.

Dans ce paragraphe, on s'intéresse à la recherche d'un plus court chemin dans un graphe entre deux sommets donnés.

I GRAPHE PONDÉRÉ

1 DÉFINITION

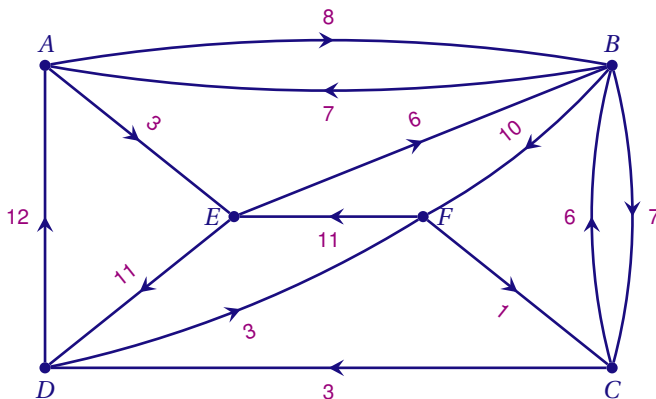
On appelle *graphe pondéré*, un graphe (orienté ou non) dont les arêtes ont été affectées d'un nombre appelé poids (ou coût).

Par analogie avec la matrice d'adjacence, on peut définir la matrice des poids $P(a_{i,j})$ du graphe, dont les coefficients $a_{i,j}$ correspondent aux poids des arêtes (ou des arcs dans le cas d'un graphe orienté) :

$$a_{i,j} = \begin{cases} 0 & \text{si } i = j \\ \infty & \text{s'il n'existe pas d'arêtes (ou d'arc) entre les sommets } x_i \text{ et } x_j \\ p_{ij} & \text{où } p_{ij} \text{ est le poids de l'arête (ou de l'arc) entre les sommets } x_i \text{ et } x_j \end{cases}$$

On utilise le symbole ∞ pour indiquer qu'il n'y a pas d'arêtes entre deux sommets.

EXEMPLE



Les sommets du graphe étant rangés dans l'ordre alphabétique :

$$P = \begin{pmatrix} 0 & 8 & \infty & \infty & 3 & \infty \\ 7 & 0 & 7 & \infty & \infty & 10 \\ \infty & 6 & 0 & 3 & \infty & \infty \\ 11 & \infty & \infty & 0 & \infty & 3 \\ \infty & 6 & \infty & 11 & 0 & \infty \\ \infty & \infty & 1 & \infty & 12 & 0 \end{pmatrix}$$

2 LONGUEUR D'UN CHEMIN

Soit $C(x,y)$ un chemin (ou une chaîne) dans un graphe pondéré G du sommet x vers le sommet y . La longueur de ce chemin est égale à la somme des poids de chacun arcs (ou de chacune des arêtes) qui le constituent.

REMARQUE

Cette définition généralise la définition de la longueur d'une chaîne dans un graphe non pondéré, il suffit d'attribuer un poids égal à 1 à chaque arête du graphe.

Dans l'exemple précédent, la longueur du chemin $AEBF$ est 19.

Si on souhaite déterminer le plus court chemin du sommet A au sommet F, on peut essayer d'énumérer tous les chemins ABF , $AEBF$, $AEDF$, $ABCDF$, $AEBCDF$ et calculer leurs longueurs. Mais avec un graphe de taille plus importante, ceci risque de devenir rapidement impossible.

Pour résoudre ce problème, on fait appel à des algorithmes.

En terminale ES, on n'étudie que le cas particulier où **les poids de tous les arcs sont des réels positifs**.

II ALGORITHME DE DIJKSTRA

E. W. Dijkstra (1930-2002) a proposé en 1959 un algorithme qui permet de calculer le plus court chemin entre un sommet particulier et tous les autres dans un graphe pondéré dont tous les poids sont positifs.

L'algorithme comporte une phase d'initialisation. À chaque sommet on attribue un poids qui vaut 0 pour le sommet de départ et infini pour les autres sommets.

Le traitement de l'algorithme consiste à examiner les sommets les uns après les autres et à sélectionner le sommet x auquel on a affecté la plus petite distance du sommet de départ jusqu'à x .
On recommence tant qu'il reste des sommets à sélectionner.

Soit G un graphe *connexe* dont les arêtes sont pondérées par des nombres *positifs*.

NOTATIONS :

S la liste des sommets du graphe

s_0 le sommet du graphe à partir duquel on veut déterminer les plus courts chemins aux autres sommets

$l(x,y)$ le poids de l'arête entre deux sommets x et y

$\delta_s(x)$ la longueur d'un chemin du sommets s_0 au sommet x

$V^+(x)$ la liste des successeurs du sommet x

$p(x)$ le prédécesseur du sommet x

X liste des sommets restant à traiter.

E liste des sommets déjà traités.

INITIALISATION :

POUR CHAQUE $x \in S$ FAIRE $\delta_s(x) = \infty$

On attribue un poids ∞ à chacun des sommets x

$\delta_s(s_0) = 0$

Le poids du sommet s_0 est nul

$X = S$

La liste des sommets restant à traiter est initialisée à S

$E = \emptyset$

La liste des sommets déjà traités vide

TRAITEMENT :

TANT_QUE $X \neq \emptyset$ FAIRE

Tant que la liste des sommets restant à traiter n'est pas vide

 Sélectionner dans la liste X le sommet x avec $\delta_s(x)$ minimum

 Retirer le sommet x de la liste X

 Ajouter le sommet x à la liste E

 POUR CHAQUE $y \in V^+(x) \cap X$ FAIRE

On examine tous les successeurs y du sommet x qui ne sont pas traités

 SI $\delta_s(y) > \delta_s(x) + l(x,y)$ ALORS

$\delta_s(y)$ prend la valeur $\delta_s(x) + l(x,y)$

La distance du sommet s_0 au sommet y est minimale

$p(y) = x$

le sommet x est le prédécesseur du sommet y

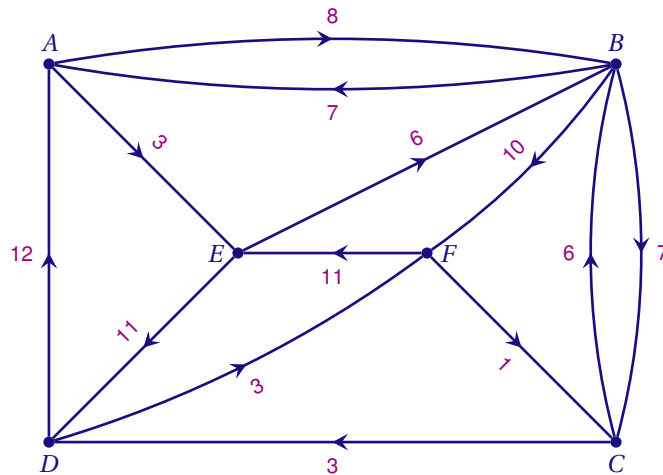
 FIN SI

 FIN POUR

FIN TANT_QUE

EXEMPLE

Considérons le graphe suivant :



On souhaite déterminer le plus court chemin du sommet A au sommet F.

A	B	C	D	E	F	Sommets sélectionnés
0	∞	∞	∞	∞	∞	A(0) Initialisation; $\delta(A) = 0$ A est sélectionné.
8(A)	∞	∞	∞	3(A)	∞	B et E sont les successeurs de A qui ne sont pas traités; $0 + 8 < \infty$ donc $\delta(B) = 8$ et $p(B) = A$; $0 + 3 < \infty$ donc $\delta(E) = 3$ et $p(E) = A$; $\delta_{min} = 3$, Le sommet E est sélectionné.
8(A)	∞	∞	14(E)		∞	B et D sont les successeurs de E qui ne sont pas traités; $3 + 6 > 8$ on ne change pas $\delta(B) = 8$ et $p(B) = A$; $3 + 11 < \infty$ donc $\delta(D) = 14$ et $p(D) = E$; $\delta_{min} = 8$, Le sommet B est sélectionné.
		15(B)	14(E)		18(B)	C et F sont les successeurs de B qui ne sont pas traités; $8 + 7 < \infty$ donc $\delta(C) = 15$ et $p(C) = B$; $8 + 10 < \infty$ donc $\delta(F) = 18$ et $p(F) = B$; $\delta_{min} = 14$, Le sommet D est sélectionné.
	15(B)				17(D)	F est le successeur de D qui n'est pas traité; $14 + 3 < 18$ donc $\delta(F) = 17$ et $p(F) = D$; $\delta_{min} = 15$, Le sommet C est sélectionné.
					17(D)	le successeur de C est déjà traité; Le sommet F est le dernier sommet traité.

- L'algorithme de Dijkstra fournit les longueurs des plus courts chemins du sommet origine aux différents sommets.
- Pour déterminer le plus court chemin du sommet origine à un sommet x , il suffit de remonter la liste des prédécesseurs en partant de x .

Ainsi, le plus court chemin de A à F est un chemin de longueur 17.

Dans la colonne F on lit que le prédécesseur de F est le sommet D. Le prédécesseur de D est E et, le prédécesseur de E est A.

Ainsi, la liste des prédécesseurs est $F \leftarrow D \leftarrow E \leftarrow A$.

Le plus court chemin de A à F est donc : A-E-D-F.