Bellman – MPM : Méthode des Potentiels Métra

1 Graphes Orientés

Vous devez tout d'abord définir des classes qui permettront de représenter un graphe orienté pondéré. Puis vous devez définir des classes qui permettront de représenter un graphe orienté pondéré et sans circuit (DAG) intégrant le tri topologique.

On veut pouvoir disposer de deux constructeurs. Le premier est un constructeur sans paramètre qui crée un graphe vide. Le second constructeur dispose d'un paramètre de type chaîne de caractères qui est le chemin vers un fichier texte contenant une description du graphe.

Exemples de fichiers (suffixé par `.gr`) :

digraph.gr	dag.gr
6 8	5 7
a b 3	1 2 7
a c 6	2 0 5
...	...

La première ligne contient le nombre de sommets et le nombre d'arcs. Un sommet est identifié par une chaîne de caractères alphanumériques. Chaque ligne à partir de la deuxième contient la description d'un arc : origine, but, poids.

2 Algorithme de Bellman

Il s'agit d'implémenter les 4 variantes suivantes de l'algorithme de Bellman :

- Recherche d'une arborescence de chemins de poids minimum (maximum) partant d'un sommet donné.
- Recherche d'une anti-arborescence de chemins de poids minimum (maximum) aboutissant en un sommet donné.

3 Méthode des Potentiels Métra (MPM)

Il s'agit d'implémenter la méthode MPM (cf. polycopié) en utilisant 2 des variantes de l'algorithme de Bellman. Les résultats fournis seront les dates de début au plus tôt, la durée globale du projet, les dates de début au plus tard, les marges libres, les marges totales, les taches critiques, un chemin critique.

Vous devez définir des classes qui permettront de représenter un graphe potentiel tâche.

Format de fichier contenant un jeu de données (suffixé par `.mpm`) :

graphe1.mpm	graphe2.mpm
6	5
a 3 b c	1 2
b 5	2 5 1 3
...	...

La première ligne contient le nombre de tâches. Chaque tâche est identifiée par une chaîne de caractères alphanumériques. Chaque ligne à partir de la deuxième décrit une tâche : le nom de la tâche, sa durée, les tâches qui la précèdent (s'il y en a).

4 Exemple

Un jeu de données

tâches	a	b	b	d	e	f	g	h	i
durée	3	8	2	7	5	2	4	3	4
tâches précédentes	–	–	a, b	a, b	b	b	c, d	d	e, f

TABLE 1 – Contraintes d'antériorité des tâches composant un projet

Graphe potentiel tâche représentant les données

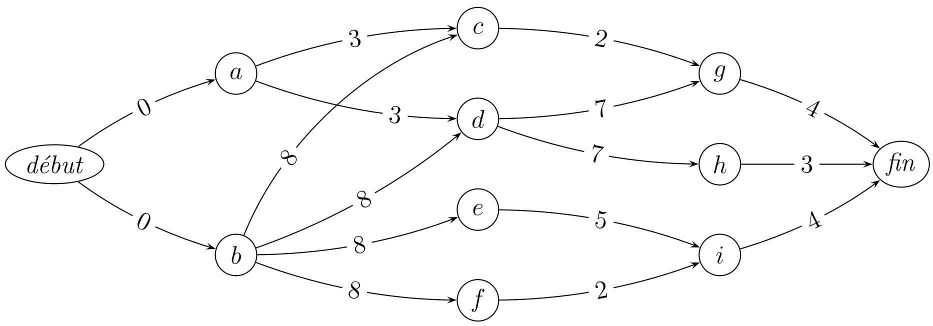

FIGURE 1 – Graphe potentiel tâche

Calcul des dates de début au plus tôt

On utilise l'algorithme de Bellman pour déterminer une arborescence de chemins de poids maximum de racine le sommet *début*.

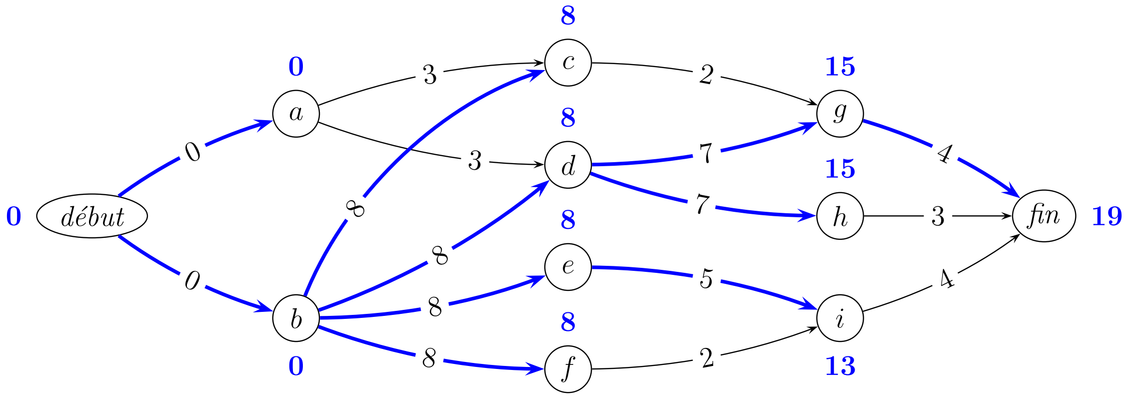

FIGURE 2 – Arborescence obtenue par algorithme de Bellman

t	*début*	a	b	c	d	e	f	g	h	i	*fin*
$\boldsymbol{tôt(t) = \Pi(t)}$	**0**	**0**	**0**	**8**	**8**	**8**	**8**	**15**	**15**	**13**	**19**
père : $\mathcal{P}(t)$	none	*début*	*début*	b	b	b	b	d	d	i	g

TABLE 2 – Dates de début au plus tôt

Durée globale du projet

On note $D_{\min}$ la durée globale (et optimale) du projet et : $D_{\min} = \textit{tôt}(\textit{fin})$. Soit 19 pour les données de l'exemple.

Calcul des dates de début au plus tard

On utilise l'algorithme de Bellman pour déterminer une anti-arborescence de chemins de poids maximum d'anti-racine le sommet *fin*.

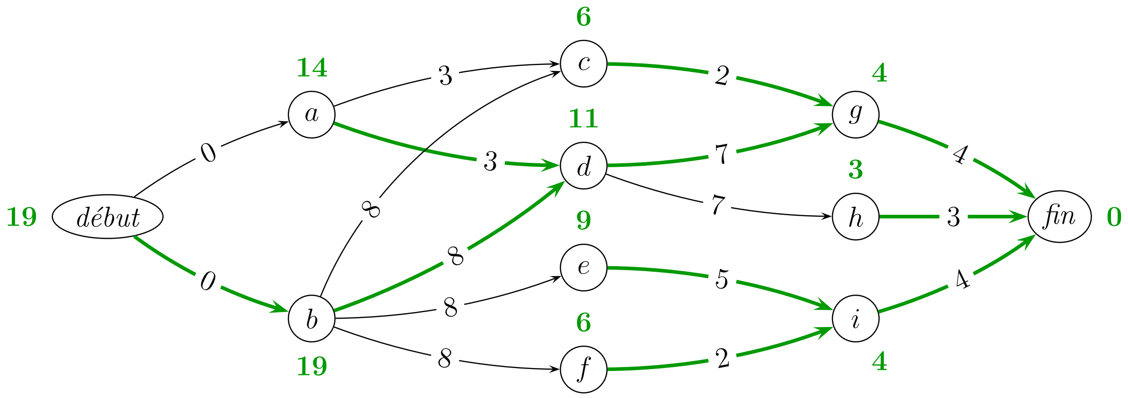

FIGURE 3 – Anti-arborescence obtenue par l'algorithme de Bellman

t	*début*	a	b	c	d	e	f	g	h	i	*fin*
$\Delta(t)$	19	14	19	6	11	9	6	4	3	4	0
fils : $\mathcal{F}(t)$	b	d	d	g	g	i	i	*fin*	*fin*	*fin*	none
$\boldsymbol{tard(t) = 19 - \Delta(t)}$	**0**	**5**	**0**	**13**	**8**	**10**	**13**	**15**	**16**	**15**	**19**

TABLE 3 – Dates de début au plus tard

Marges

La **marge libre** d'une tâche t est définie comme suit :

$$m(t) = \min_{x \in V^+(t)} \{ \textit{tôt}(x) - (\textit{tôt}(t) + \textit{durée}(t)) \} = \min_{x \in V^+(t)} \{ \textit{tôt}(x) \} - \textit{tôt}(t) - \textit{durée}(t)$$

La **marge totale** d'une tâche t est définie comme suit :

$$\forall t \in S, \ \ M(t) = \textit{tard}(t) - \textit{tôt}(t)$$

t	*début*	a	b	c	d	e	f	g	h	i	*fin*
$\boldsymbol{m(t)}$	**0**	**5**	**0**	**5**	**0**	**0**	**3**	**0**	**1**	**2**	**0**
$\boldsymbol{M(t)}$	**0**	**5**	**0**	**5**	**0**	**2**	**5**	**0**	**1**	**2**	**0**

TABLE 4 – Marges

Tâches critiques et chemin critique

Une tâche t est dite critique si sa marge totale est nulle, c.-à-d. si $M(t) = 0$. Un chemin critique est un chemin de poids maximum allant du sommet *début* au sommet *fin*. Les sommets d'un chemin critique représentent des tâches critiques.

Dans le cas de l'exemple (voir Table 4), les tâches critiques sont les tâches b, d et g. Il n'existe qu'un seul chemin critique (voir Figure 2 ou Figure 3) : $(\textit{début}, b, d, g, \textit{fin})$.