

Exercice 1

Les valuations entre crochets des arcs du graphe G_1 représentent des capacités.

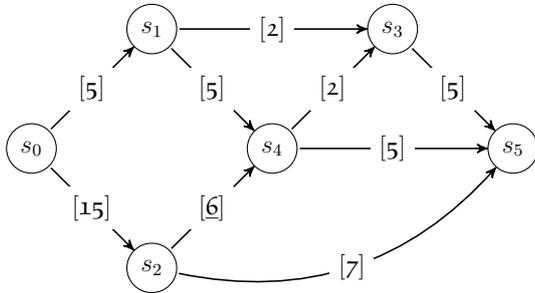


FIGURE 1 – G_1

- Déterminer sur le graphe G_1 un flot maximal du sommet s_0 au sommet s_5 . Quels arcs du graphe sont saturés ? Quelle est la coupe minimum obtenue par l'algorithme de Ford et Fulkerson ?

Exercice 2

Les valuations des arcs du graphe G_2 définissent un flot sur un graphe de capacité borné (les capacités des arcs sont indiquées entre crochets).

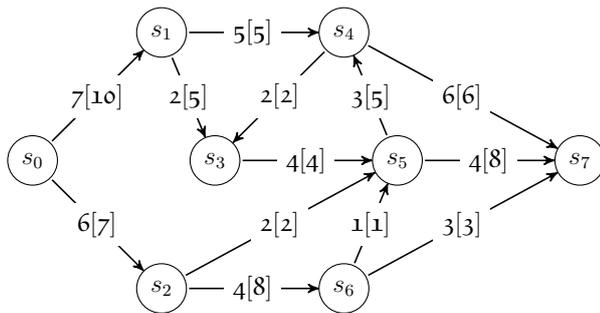


FIGURE 2 – G_2

- Le flot de s_0 à s_7 est-il maximum ? Si non, l'optimiser.
- Vous indiquerez une coupe minimum associée au flot maximal obtenu.

Exercice 3

Dans le graphe G_3 les valuations entre crochets des arcs représentent des capacités.

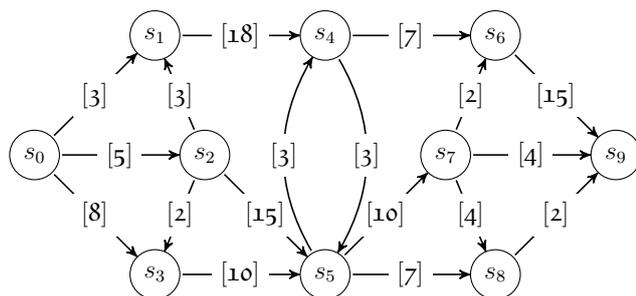


FIGURE 3 – G_3

- En utilisant l'algorithme de *Ford-Fulkerson*, déterminer sur le graphe G_3 un flot maximum du sommet s_0 au sommet s_9 (vous indiquerez les augmentations successives du flot, et vous mettrez en évidence le fait que le dernier flot obtenu est maximal).

Exercice 4

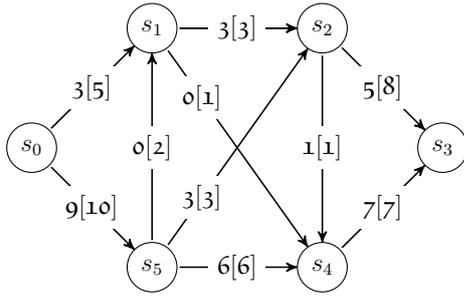


FIGURE 4 – G_4

Dans le graphe G_4 les valuations entre crochets des arcs représentent des capacités.

1. En utilisant l'algorithme de *Ford-Fulkerson*, déterminer sur le graphe G_4 un flot maximum du sommet s_0 au sommet s_3 (vous indiquerez les augmentations successives du flot, et vous mettez en évidence le fait que le dernier flot obtenu est maximum).

Exercice 5

(*) Les arcs du graphe G_5 sont valués par des nombres entre crochets (représentant leur capacité), et des nombres représentant un flot compatible avec les capacités.

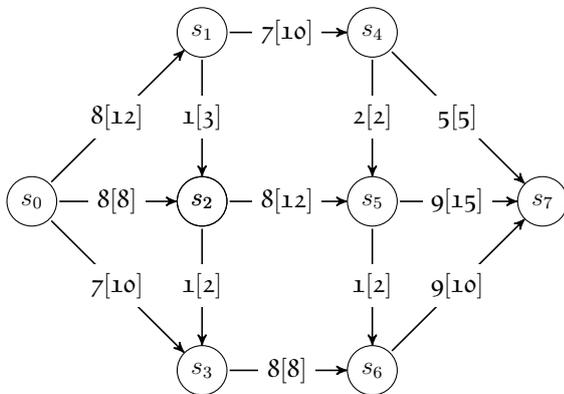


FIGURE 5 – G_5

1. Utiliser l'algorithme de *Ford-Fulkerson* pour augmenter le flot indiqué de façon à obtenir un flot maximum allant du sommet s_0 au sommet s_7 (vous montrerez que le flot obtenu est maximum).

Exercice 6

(*) Les arcs du graphe G_6 sont valués par des nombres entre crochets (représentant leur capacité), et des nombres représentant un flot compatible avec les capacités.

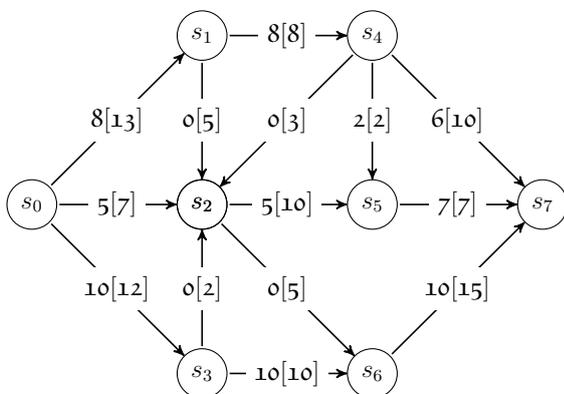


FIGURE 6 – G_6

1. En utilisant l'algorithme de *Ford-Fulkerson*, augmenter le flot du sommet s_0 au sommet s_7 . Vous ferez apparaître les différentes étapes de l'optimisation, et vous indiquerez une coupe minimum.

Algorithme de Ford et Fulkerson

G : graphe orienté

$c : E(G) \rightarrow \mathbb{R}^+$

s : source de G

t : puit de G

On considère un arc (t, s) de capacité $c(t, s)$ infinie. La valeur du flot sur cet arc sera la "valeur du flot de s à t ".

On note respectivement par $I(e)$ et $T(e)$ l'extrémité initiale et l'extrémité terminale d'un arc e .

Algorithme 1 FlotMax(G, c, s, t)

```
1: pour tout  $e$  de  $E(G)$  faire
2:    $f[e] \leftarrow 0$ 
3: fin pour
4: répéter
5:   Marquage( $G, c, f, s, t$ )
6:   si  $t \in Y$  alors
7:      $v \leftarrow t$ 
8:      $C^+ \leftarrow \{(t, s)\}$ 
9:      $C^- \leftarrow \emptyset$ 
10:    tant que  $v \neq s$  faire
11:       $e \leftarrow A[v]$ 
12:      si  $v = T[e]$  alors
13:         $C^+ \leftarrow C^+ \cup \{e\}$ 
14:         $v \leftarrow I[e]$ 
15:      sinon
16:         $C^- \leftarrow C^- \cup \{e\}$ 
17:         $v \leftarrow T[e]$ 
18:      fin si
19:    fin tant que
20:  fin si
21:  pour tout  $e \in C^+$  faire
22:     $f(e) \leftarrow f(e) + \delta[t]$ 
23:  fin pour
24:  pour tout  $e \in C^-$  faire
25:     $f(e) \leftarrow f(e) - \delta[t]$ 
26:  fin pour
27: jusqu'à  $t \notin Y$ 
```

Algorithme 2 Marquage(G, c, f, s, t)

```
1:  $Y \leftarrow \{s\}$ 
2:  $\delta(s) \leftarrow +\infty$ 
3:  $Max \leftarrow \text{faux}$ 
4: tant que  $t \notin Y$  et  $Max = \text{faux}$  faire
5:   si il existe  $e = (u, v)$  avec  $u \in Y, v \notin Y, f(e) < c(e)$  alors
6:      $Y \leftarrow Y \cup \{v\}$ 
7:      $A[v] \leftarrow e$ 
8:      $\delta[v] \leftarrow \min(\delta[u], c(e) - f(e))$ 
9:   sinon
10:    si il existe  $e = (u, v)$  avec  $v \in Y, u \notin Y, f(e) > 0$  alors
11:       $Y \leftarrow Y \cup \{u\}$ 
12:       $A[u] \leftarrow e$ 
13:       $\delta[u] \leftarrow \min(\delta[v], f(e))$ 
14:    sinon
15:       $Max \leftarrow \text{vrai}$ 
16:    fin si
17:  fin si
18: fin tant que
```
