Algorithmique des graphes

David Pichardie

11 Avril 2018

Bilan du CM5

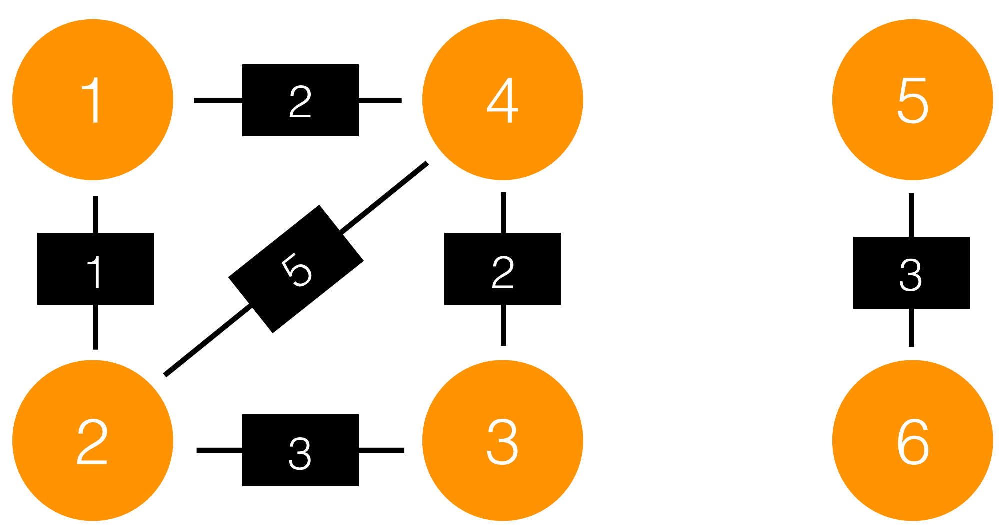
- Fermeture transitive
  - Explication de l'algorithme de Warshall
  - Déroulage de l'algorithme de Warshall
- Composantes fortement connexes
  - Application à la fermeture transitive
  - Algorithme de Tarjan

Graphe pondéré

Définition

Un *graphe non-orienté pondéré* est un graphe non-orienté muni d'une fonction de pondération qui associe une valeur réelle à chaque arête.

Exemple

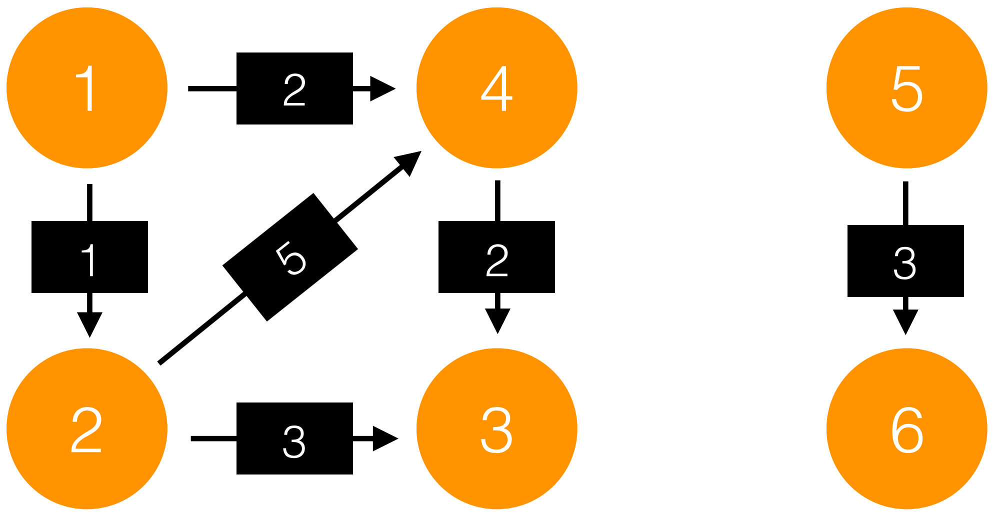


Graphe pondéré

Définition

Un *graphe orienté pondéré* est un graphe orienté muni d'une fonction de pondération qui associe une valeur réelle à chaque arc.

Exemple



Applications

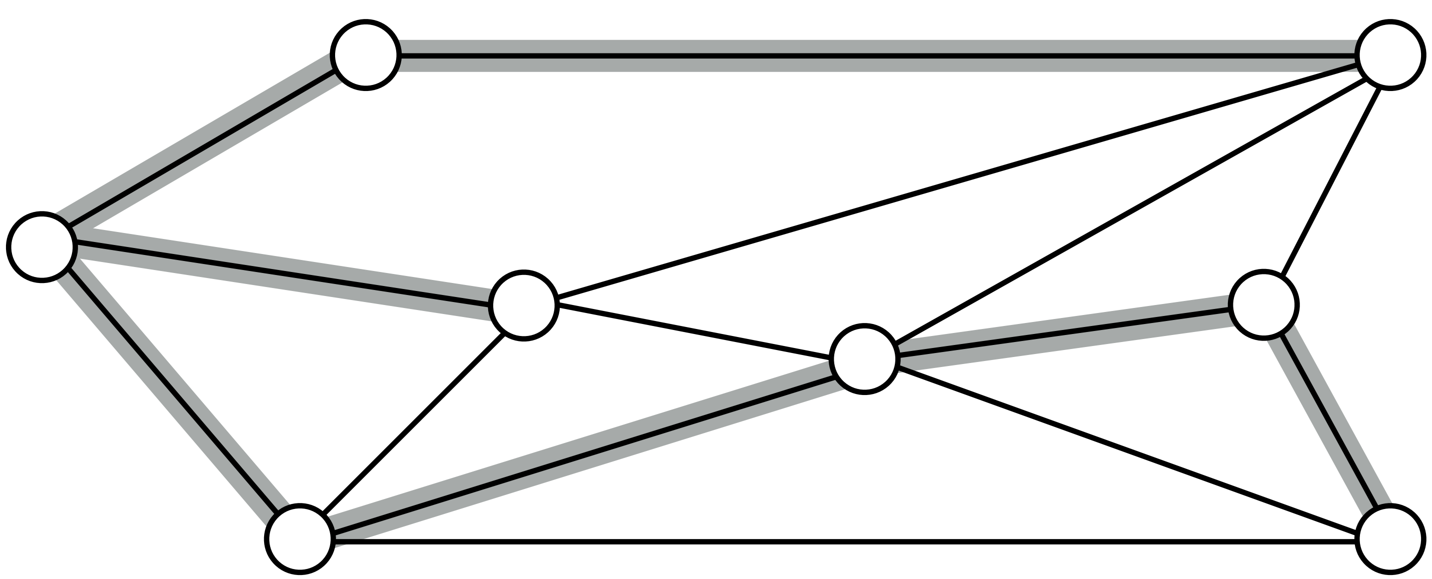
- Distances entre villes
  - Calcul de plus courts chemins
  - Arbre couvrant minimaux
- Contraintes entre tâches
  - Ajout de contraintes numériques (temporelles)

Arbre couvrant

Définition

Dans un graphe G non-orienté (pondéré ou non) et connexe, un arbre couvrant est un sous-graphe connexe acyclique (arbre) incluant tous les sommets de G.

Exemple

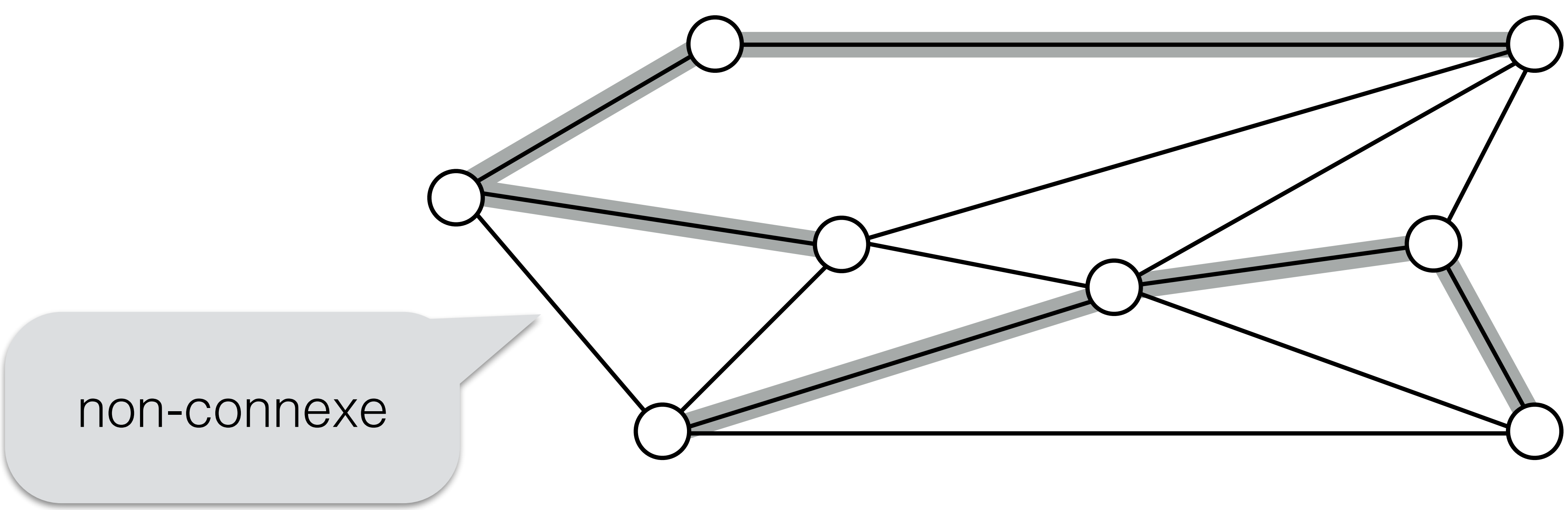


Arbre couvrant

Définition

Dans un graphe G non-orienté (pondéré ou non) et connexe, un arbre couvrant est un sous-graphe connexe acyclique (arbre) incluant tous les sommets de G.

Contre-exemple

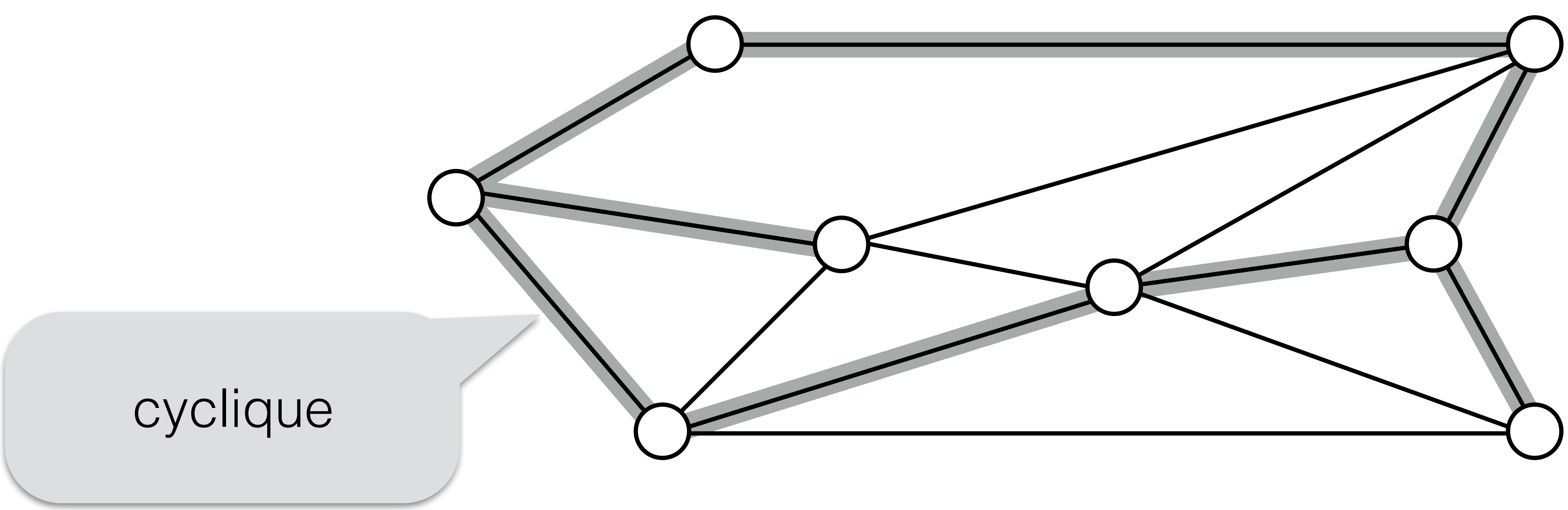


Arbre couvrant

Définition

Dans un graphe G non-orienté (pondéré ou non) et connexe, un arbre couvrant est un sous-graphe connexe acyclique (arbre) incluant tous les sommets de G.

Contre-exemple

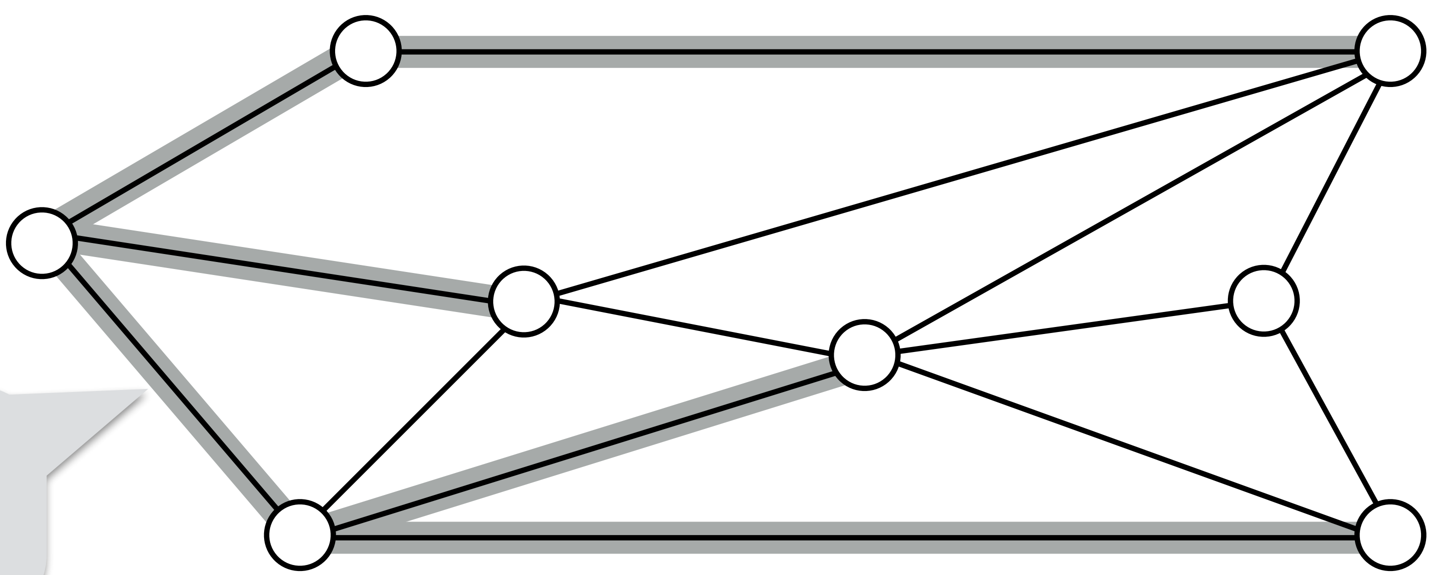


Arbre couvrant

Définition

Dans un graphe G non-orienté (pondéré ou non) et connexe, un arbre couvrant est un sous-graphe connexe acyclique (arbre) incluant tous les sommets de G.

Contre-exemple



non couvrant

Question

Connaissez vous un algorithme capable de construire un tel arbre ?

Question

Connaissez vous un algorithme capable de construire un tel arbre ?

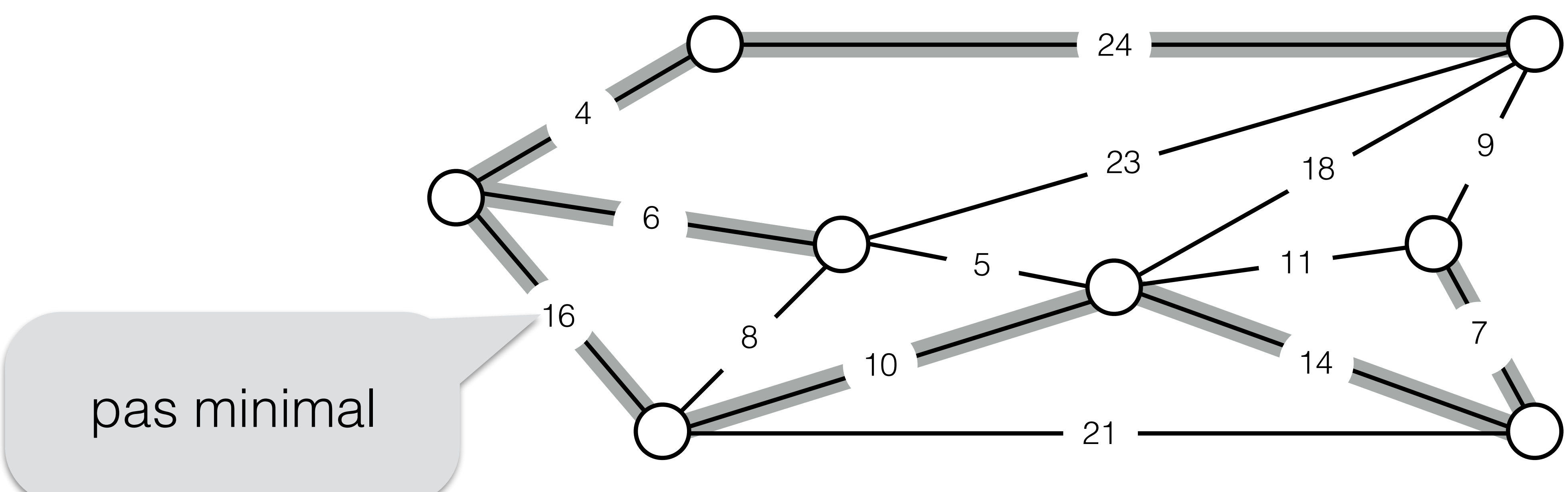
tous les algorithmes de parcours que nous avons vu

Problème de l'arbre couvrant minimal

Définition

Dans un graphe G non-orienté pondéré et connexe, un arbre couvrant minimal est un arbre couvrant dont la somme des arêtes est minimale.

Contre-exemple



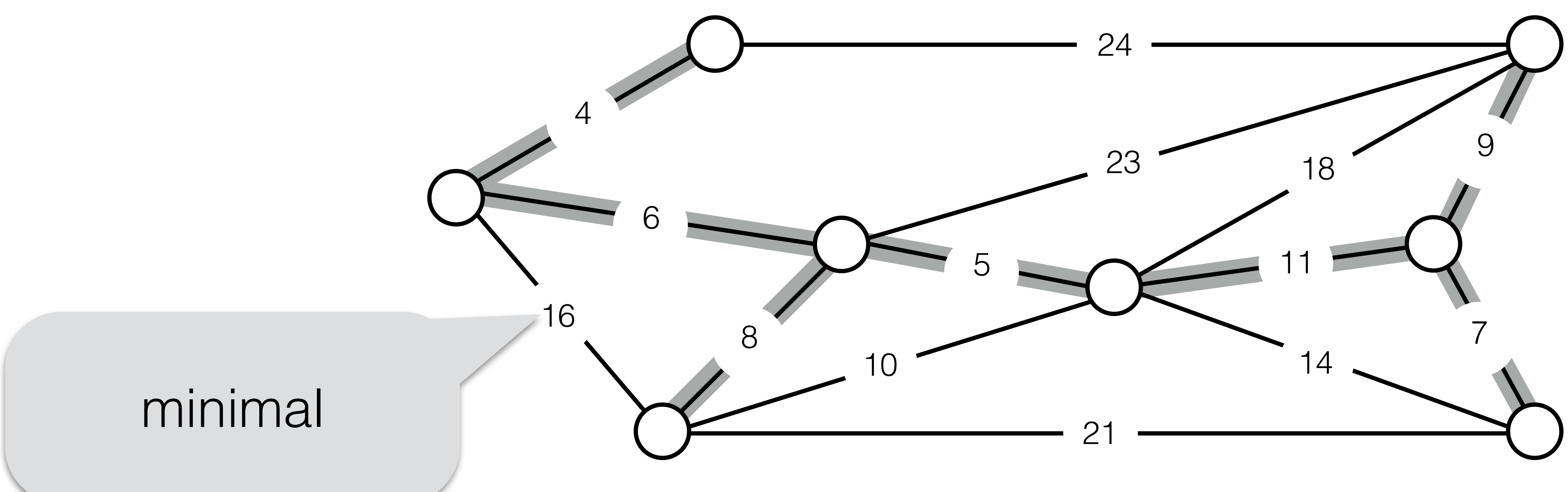
Coût total = 24 + 4 + 6 + 16 + 10 + 14 + 7 = 81

Problème de l'arbre couvrant minimal

Définition

Dans un graphe G non-orienté pondéré et connexe, un arbre couvrant minimal est un arbre couvrant dont la somme des arêtes est minimale.

Exemple



Coût total = 4 + 6 + 8 + 5 + 11 + 7 + 9 = 50

Question

Allons nous devoir énumérer tous les arbres couvrants pour trouver un arbre de coût minimal ?

Applications

- Minimiser la longueur de cable nécessaire pour connecter un réseaux, en réutilisant les gaines disponibles

- Utilisé pour certaines heuristiques dans les problèmes d'optimisation

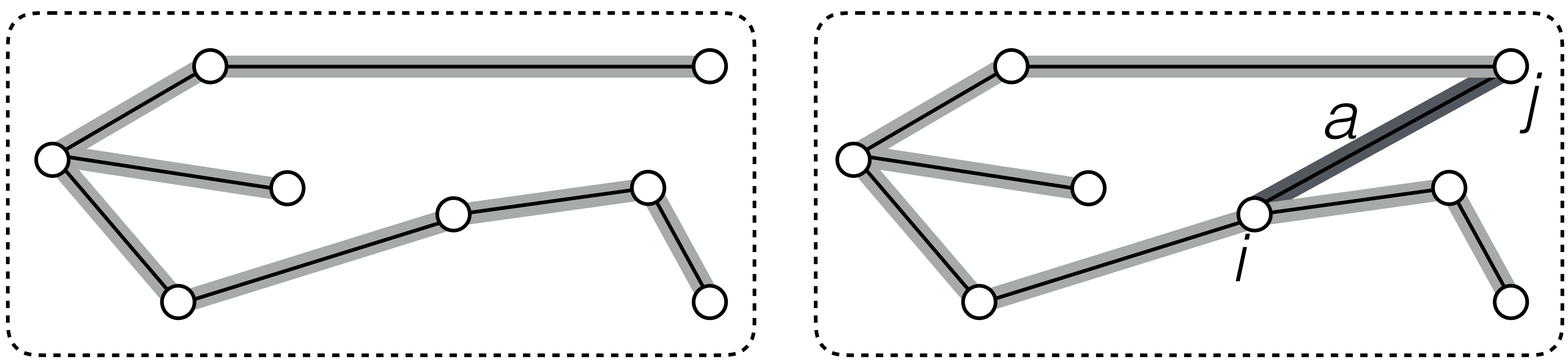
    - problème du voyageur de commerce

trouver un cycle de longueur totale minimale qui passe exactement une fois par chaque sommet du graphe

NP-complet

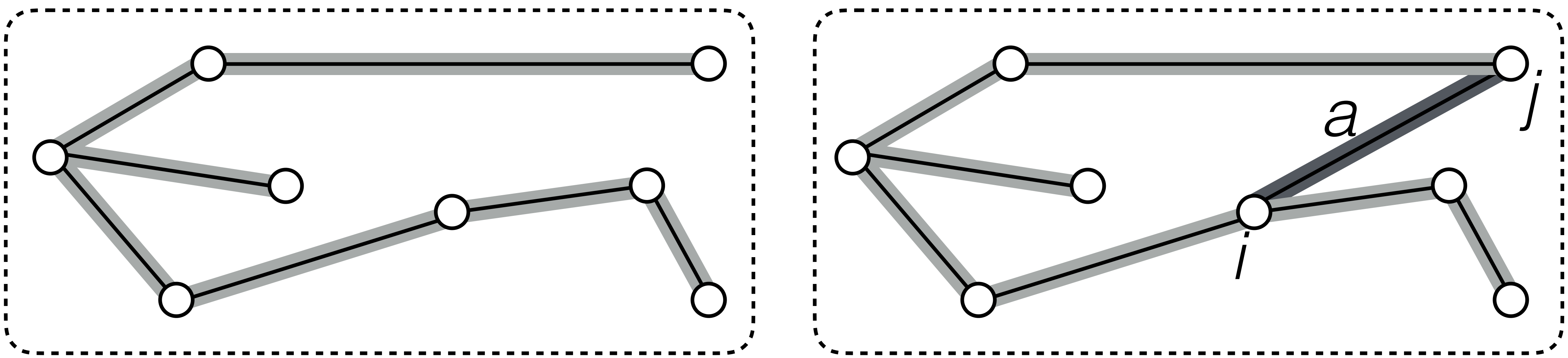
Propriété

Ajouter une arête à un graphe non-orienté connexe, crée un cycle



Propriété

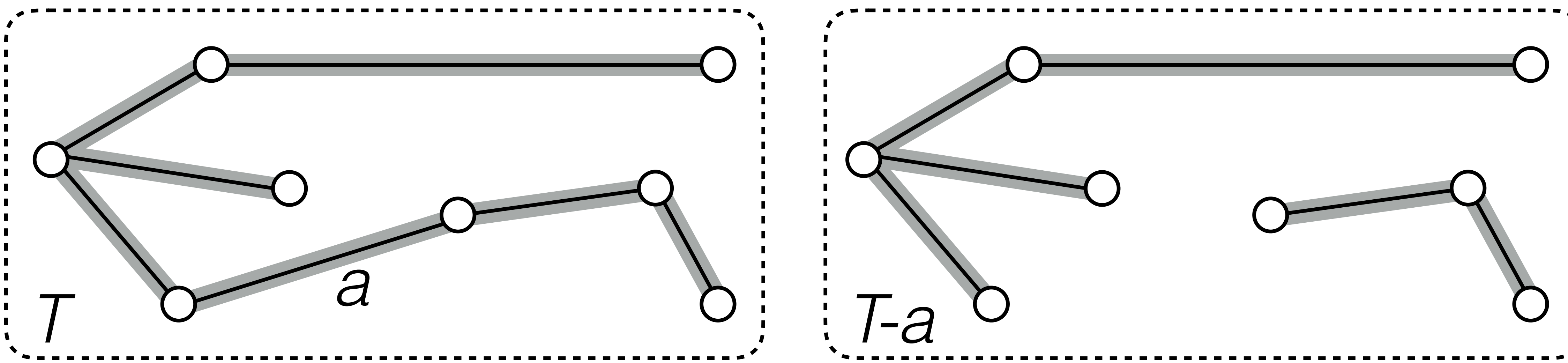
Ajouter une arête à un graphe non-orienté connexe, crée un cycle



- L'arête *a* ajoutée connecte deux sommets *i* et *j.* Dans le graphe initial, *i* et *j* étaient déjà connecté par un chemin. Ajouter *a* à ce chemin crée un cycle.

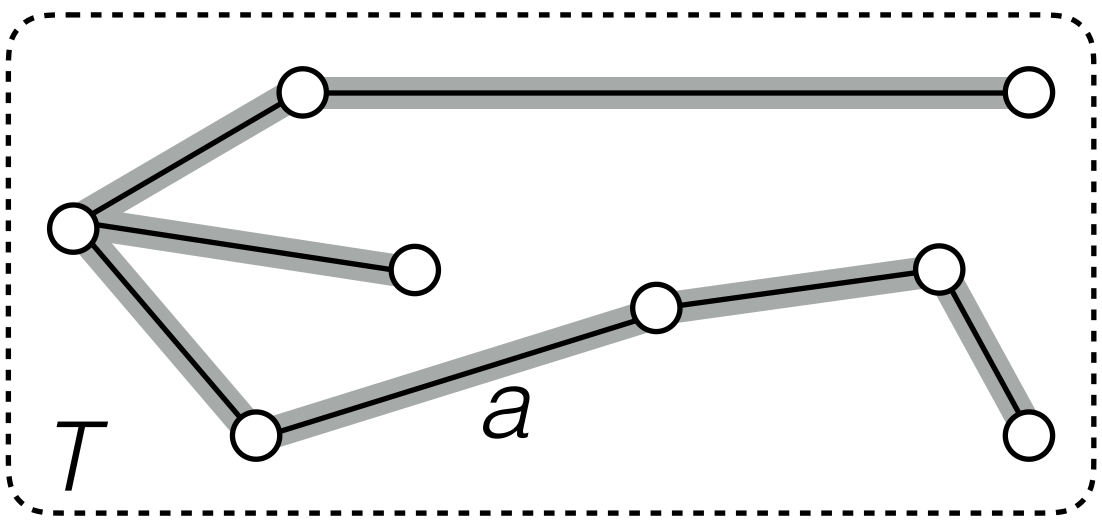
Propriétés sur les arbres

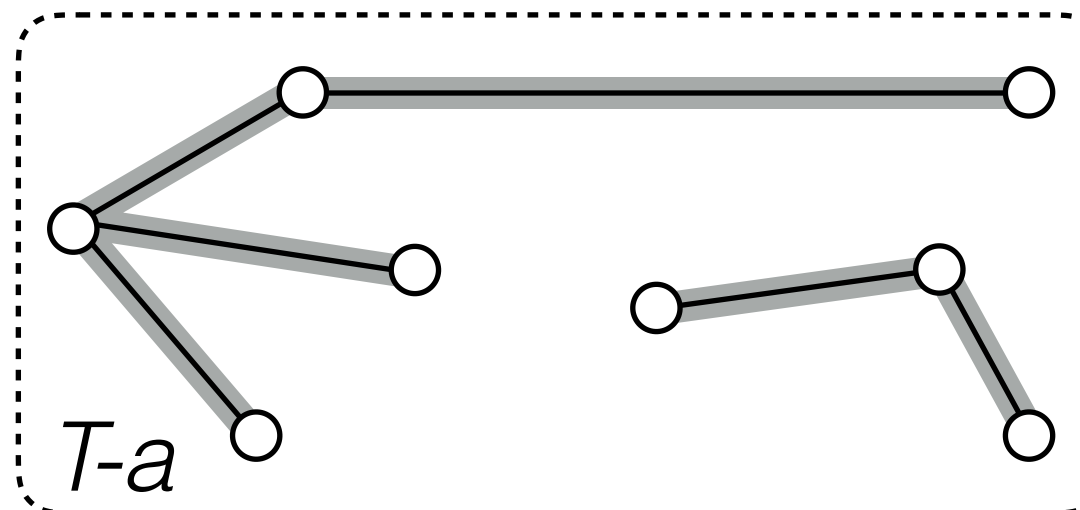
Enlever une arête à un arbre le rend non-connexe.



Propriétés sur les arbres

Enlever une arête à un arbre le rend non-connexe.

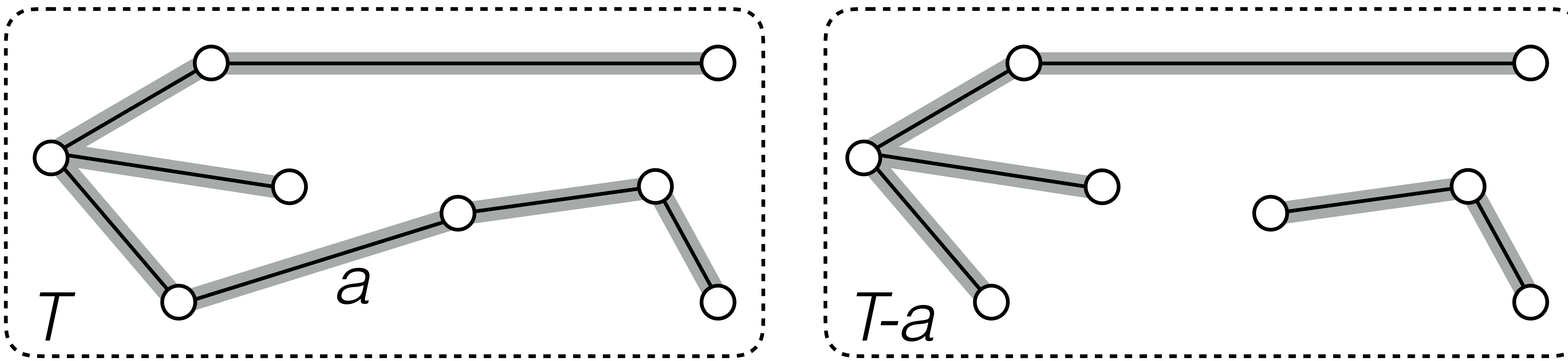




- On note *T-a* l'arbre privé d'une de ses arêtes *a*. Et on suppose, par l'absurde qu'il est connexe.

Propriétés sur les arbres

Enlever une arête à un arbre le rend non-connexe.

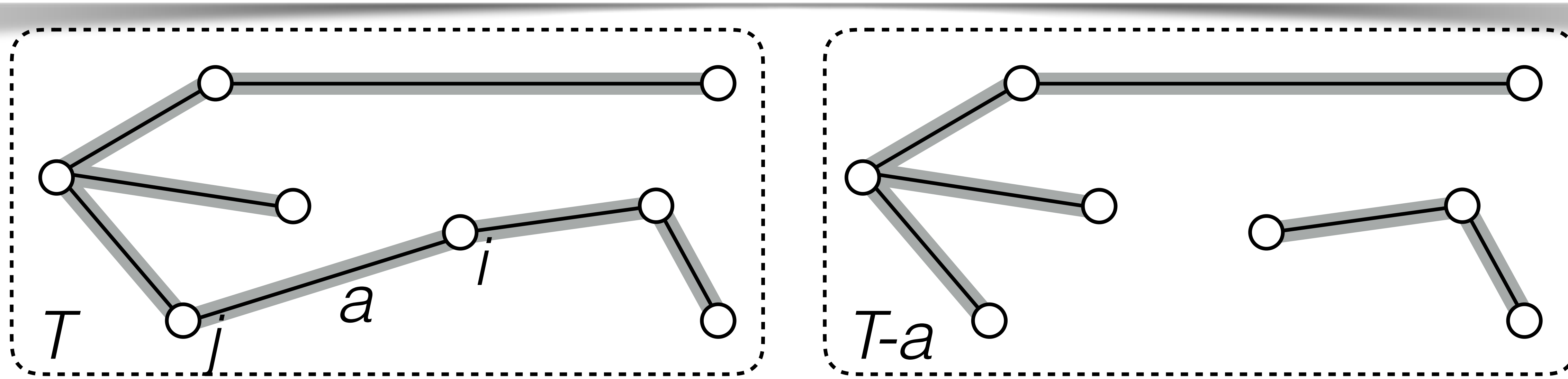


- On note *T-a* l'arbre privé d'une de ses arêtes *a*. Et on suppose, par l'absurde qu'il est connexe.

- D'après la propriété précédente, T contient un cycle, ce qui est absurde puisque T est un arbre.

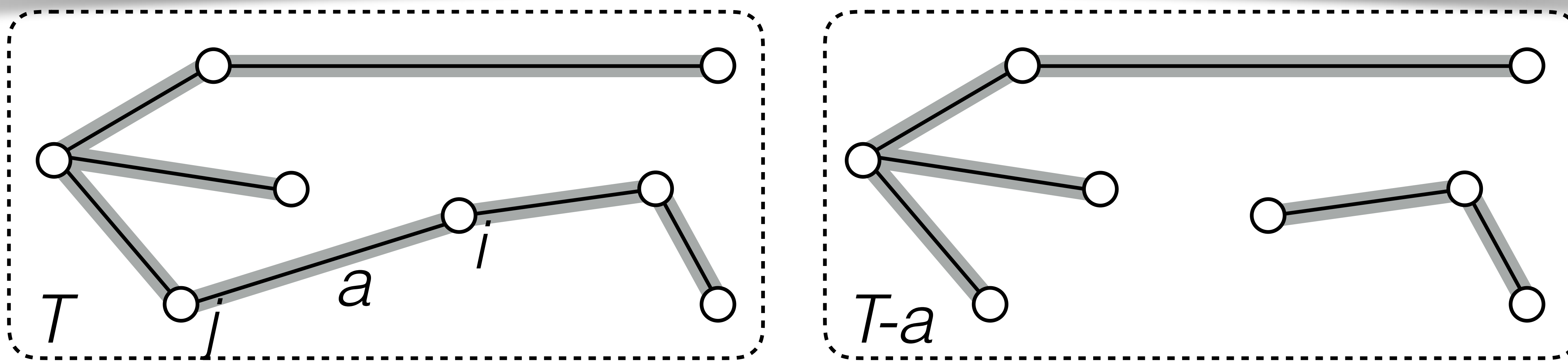
Propriétés sur les arbres

Enlever une arête à un arbre le rend non-connexe. Le graphe obtenu comporte exactement deux composantes connexes.



Propriétés sur les arbres

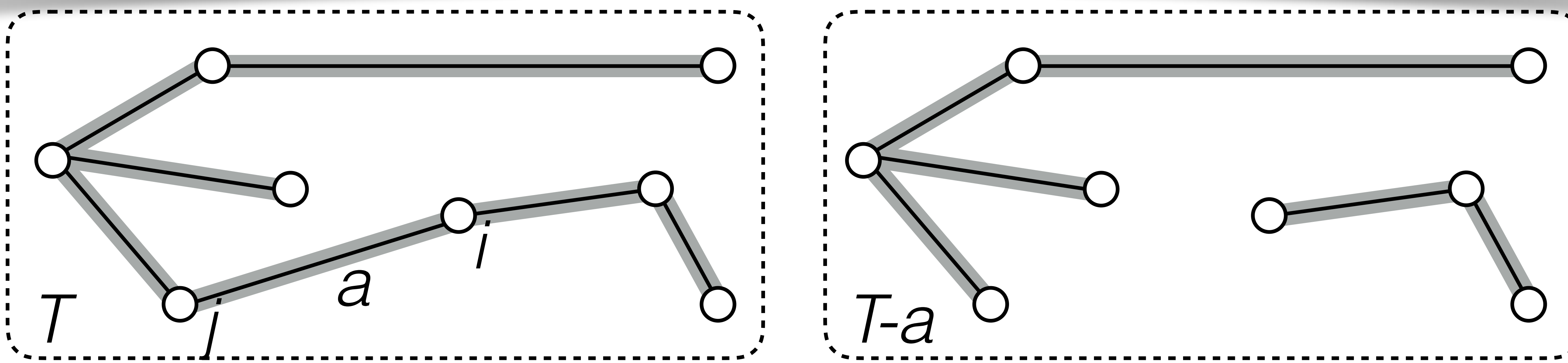
Enlever une arête à un arbre le rend non-connexe. Le graphe obtenu comporte exactement deux composantes connexes.



- On note *T-a* l'arbre privé d'une de ses arêtes *a*.

Propriétés sur les arbres

Enlever une arête à un arbre le rend non-connexe. Le graphe obtenu comporte exactement deux composantes connexes.



- On note *T-a* l'arbre privé d'une de ses arêtes *a*.

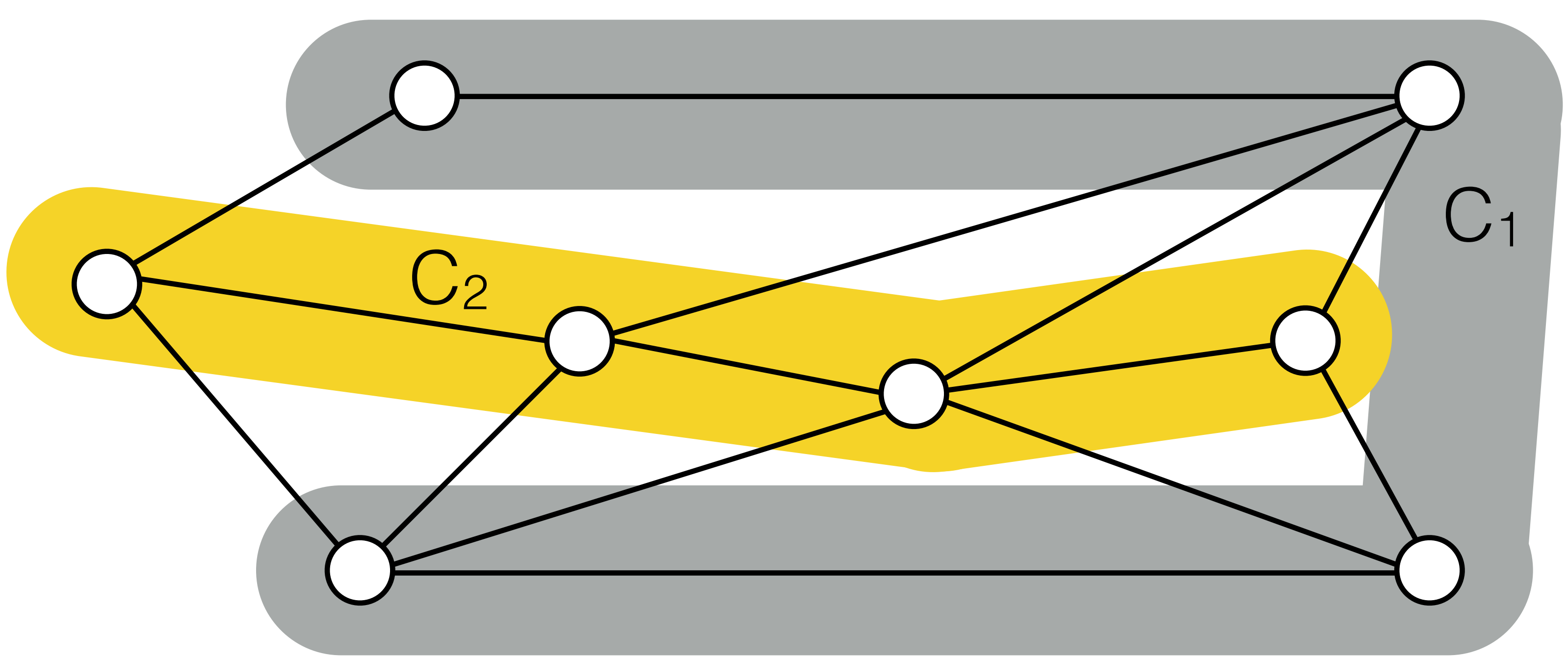
- Les extrémités *i* et *j* de l'arête *a* appartiennent chacune à des composantes connexes distinctes de *T-a.* Ajouter *a* fusionne ces deux composantes mais ne diminue que de 1 le nombre de composantes. Puisque *T* ne contient qu'une composante, *T'* en contient 2.

Coupure

Définition

Dans un graphe orienté ou non, une *coupure* est une partition (C_1, C_2) de l'ensemble des sommets en deux ensembles disjoints. Une arête *traversante* est une arête qui passe d'un ensemble à l'autre

Exemple

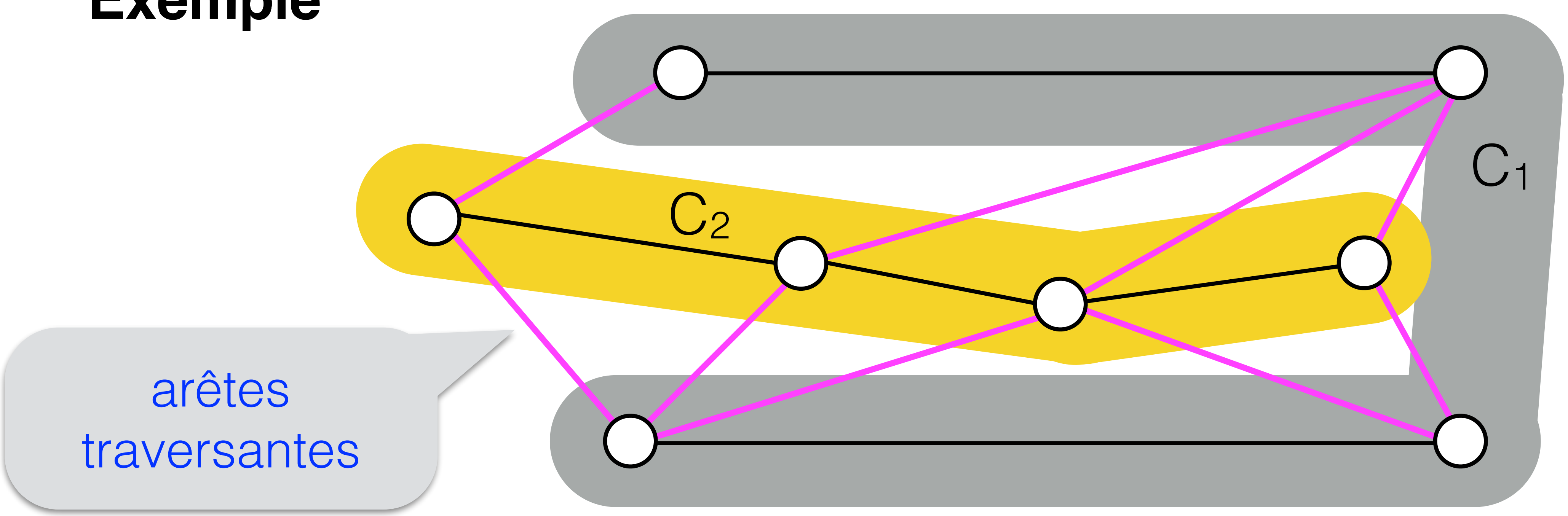


Coupure

Définition

Dans un graphe orienté ou non, une *coupure* est une partition (C_1, C_2) de l'ensemble des sommets en deux ensembles disjoints. Une arête *traversante* est une arête qui passe d'un ensemble à l'autre

Exemple



Hypothèse

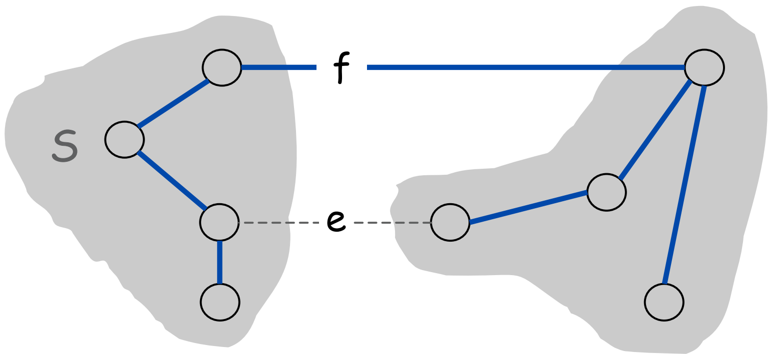
On considère une graphe non-orienté pondéré connexe. Nous supposons que tous ses arêtes ont des poids différents.

Propriété (admise) : cela assure l'unicité d'un arbre couvrant minimal ACM

Les algorithmes vus dans ce cours sont corrects sans cette hypothèse mais l'arbre couvrant minimal n'est plus unique

Propriété des cycles

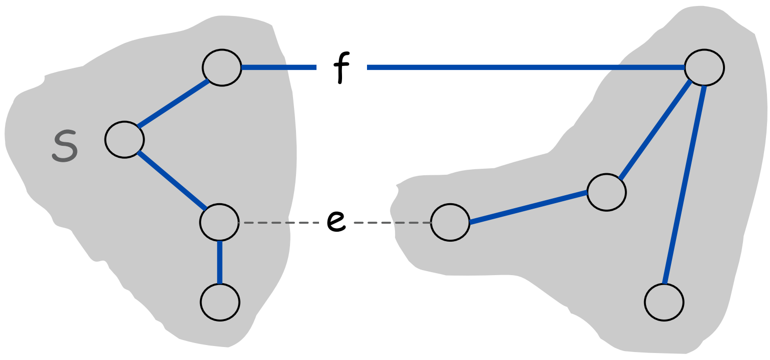
Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.



Propriété des cycles

Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.

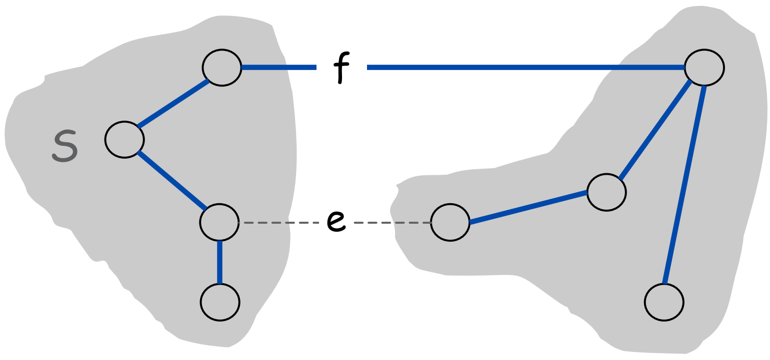
- Supposons, par l'absurde, que f appartienne à ACM.



Propriété des cycles

Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.

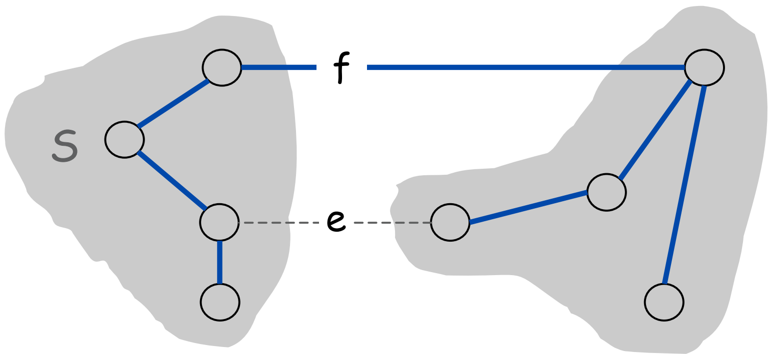
- Supposons, par l'absurde, que f appartienne à ACM.
- Enlever f de ACM le coupe en 2 composantes connexes.



Propriété des cycles

Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.

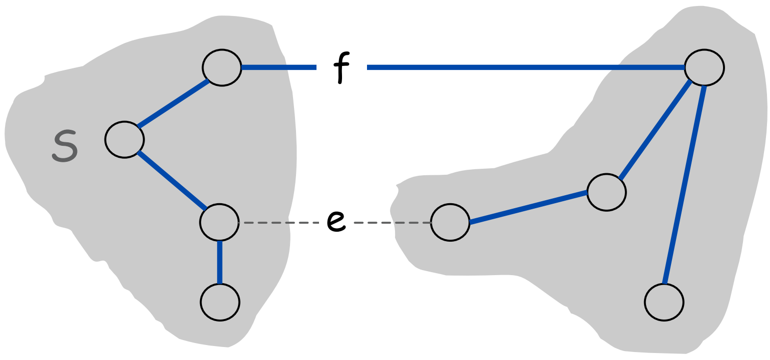
- Supposons, par l'absurde, que f appartienne à ACM.
- Enlever f de ACM le coupe en 2 composantes connexes.
- Il existe une arête e du cycle C qui peut re-connecter les deux composantes.



Propriété des cycles

Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.

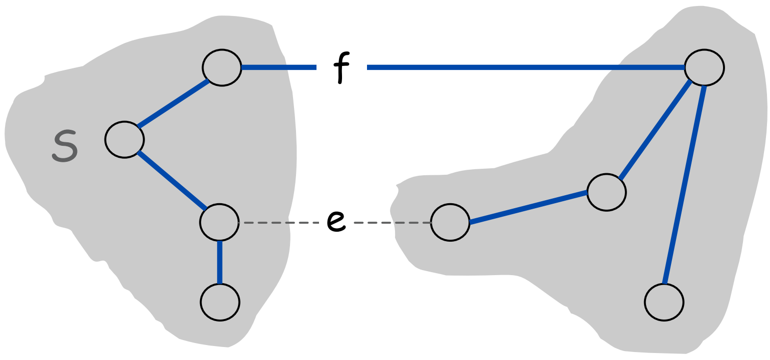
- Supposons, par l'absurde, que f appartienne à ACM.
- Enlever f de ACM le coupe en 2 composantes connexes.
- Il existe une arête e du cycle C qui peut re-connecter les deux composantes.
- ACM-{f}+{e} est un nouveau arbre couvrant.



Propriété des cycles

Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.

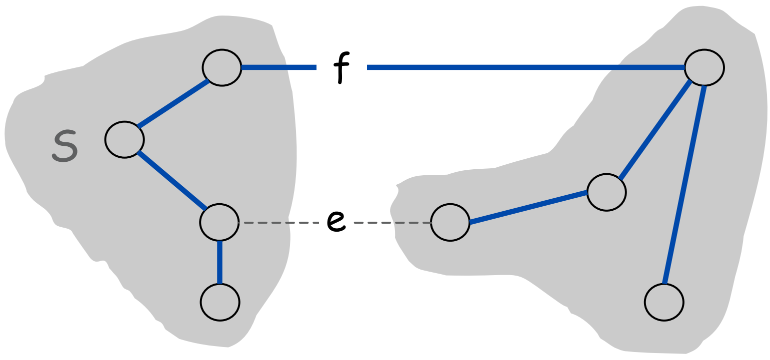
- Supposons, par l'absurde, que f appartienne à ACM.
- Enlever f de ACM le coupe en 2 composantes connexes.
- Il existe une arête e du cycle C qui peut re-connecter les deux composantes.
- ACM-{f}+{e} est un nouveau arbre couvrant.
- Son poid total est strictement plus petit que celui de ACM.



Propriété des cycles

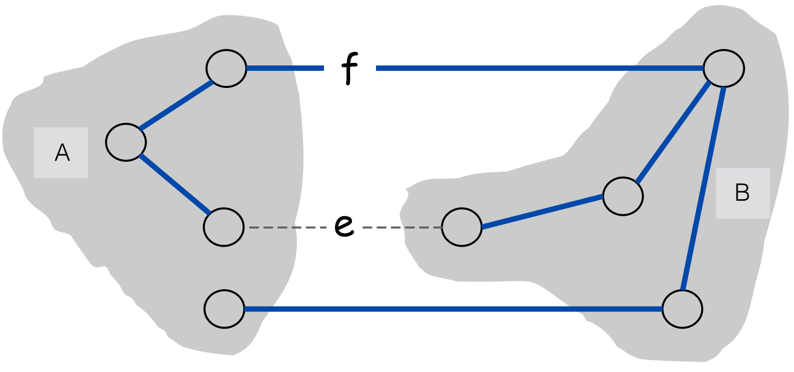
Soit C un cycle dans G. Soit f une arête de C de poids maximum dans ce cycle. Alors f n'appartient pas à ACM.

- Supposons, par l'absurde, que f appartienne à ACM.
- Enlever f de ACM le coupe en 2 composantes connexes.
- Il existe une arête e du cycle C qui peut re-connecter les deux composantes.
- ACM-{f}+{e} est un nouveau arbre couvrant.
- Son poid total est strictement plus petit que celui de ACM.
- Contradiction



Propriété de la coupure

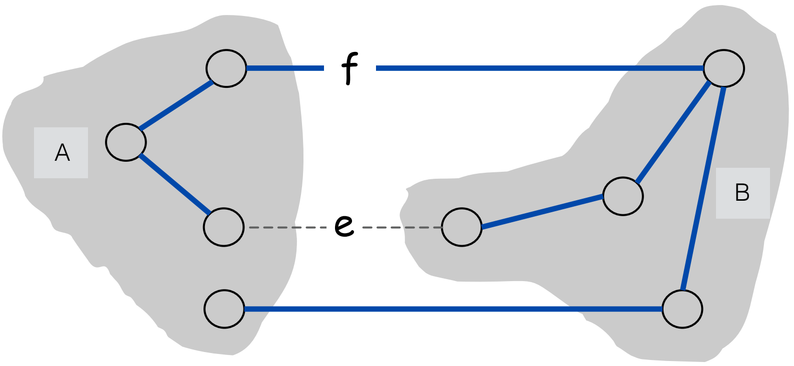
Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.



Propriété de la coupure

Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.

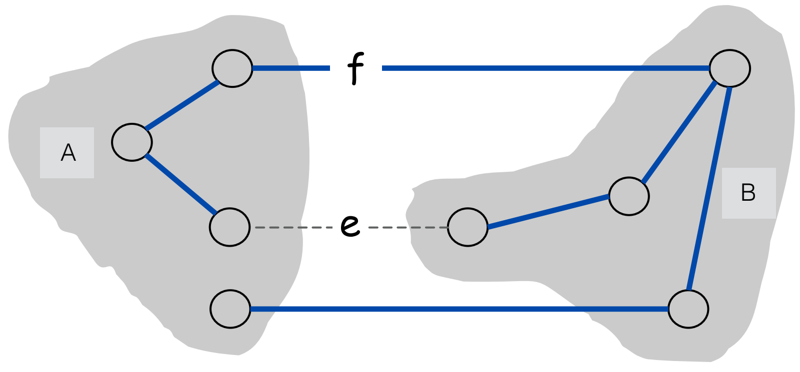
- Supposons, par l'absurde, que e n'appartienne pas à ACM.



Propriété de la coupure

Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.

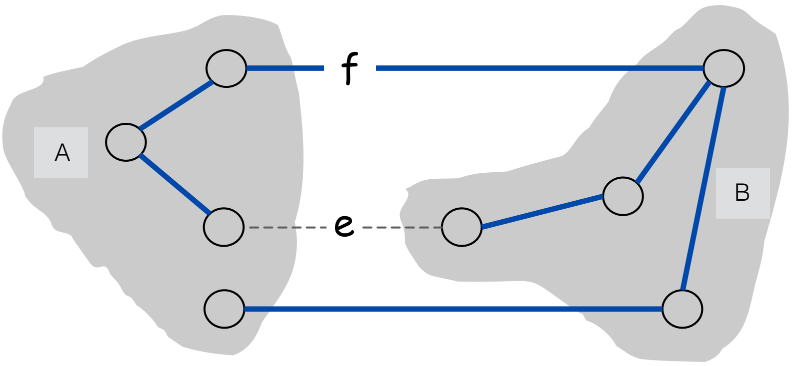
- Supposons, par l'absurde, que e n'appartienne pas à ACM.
- Ajouter e à ACM crée un cycle.



Propriété de la coupure

Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.

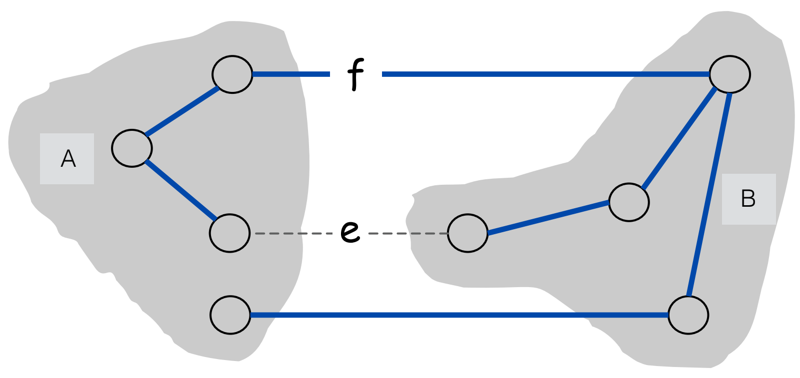
- Supposons, par l'absurde, que e n'appartienne pas à ACM.
- Ajouter e à ACM crée un cycle.
- Il existe une autre arête traversante f qui sépare ACM en deux composantes.



Propriété de la coupure

Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.

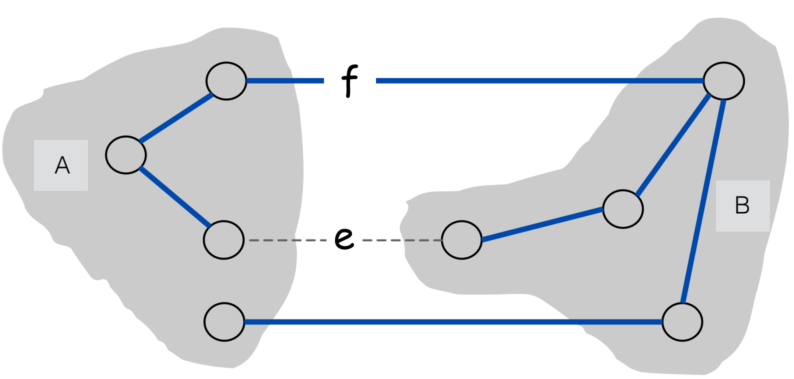
- Supposons, par l'absurde, que e n'appartienne pas à ACM.
- Ajouter e à ACM crée un cycle.
- Il existe une autre arête traversante f qui sépare ACM en deux composantes.
- ACM-{f}+{e} est un nouveau arbre couvrant.



Propriété de la coupure

Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.

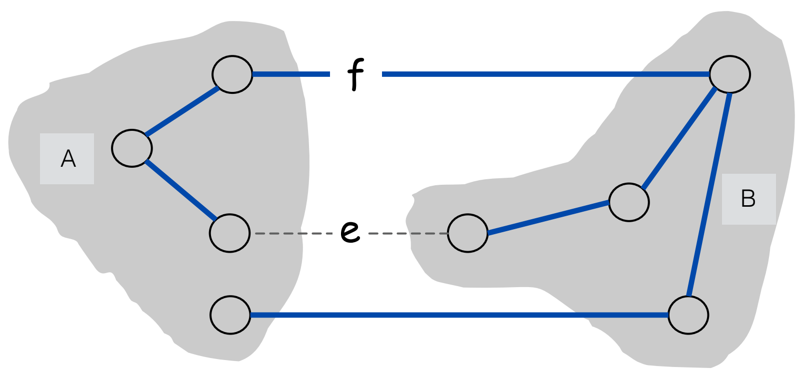
- Supposons, par l'absurde, que e n'appartienne pas à ACM.
- Ajouter e à ACM crée un cycle.
- Il existe une autre arête traversante f qui sépare ACM en deux composantes.
- ACM-{f}+{e} est un nouveau arbre couvrant.
- Son poid total est strictement plus petit que celui de ACM.



Propriété de la coupure

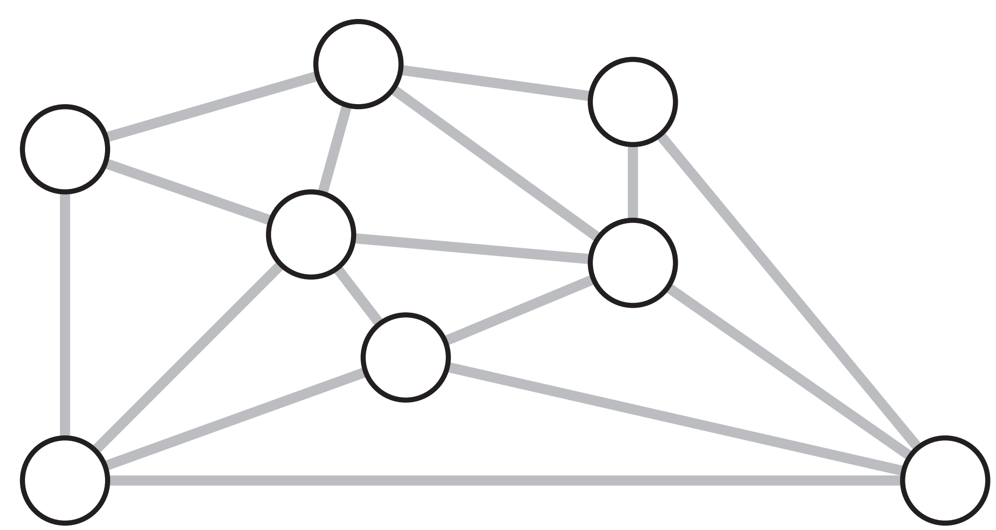
Soit une coupure (A,B) de G. Soit e l'arête traversante de poids minimum vis à vis de cette coupure. Alors e appartient forcement à ACM.

- Supposons, par l'absurde, que e n'appartienne pas à ACM.
- Ajouter e à ACM crée un cycle.
- Il existe une autre arête traversante f qui sépare ACM en deux composantes.
- ACM-{f}+{e} est un nouveau arbre couvrant.
- Son poid total est strictement plus petit que celui de ACM.
- Contradiction



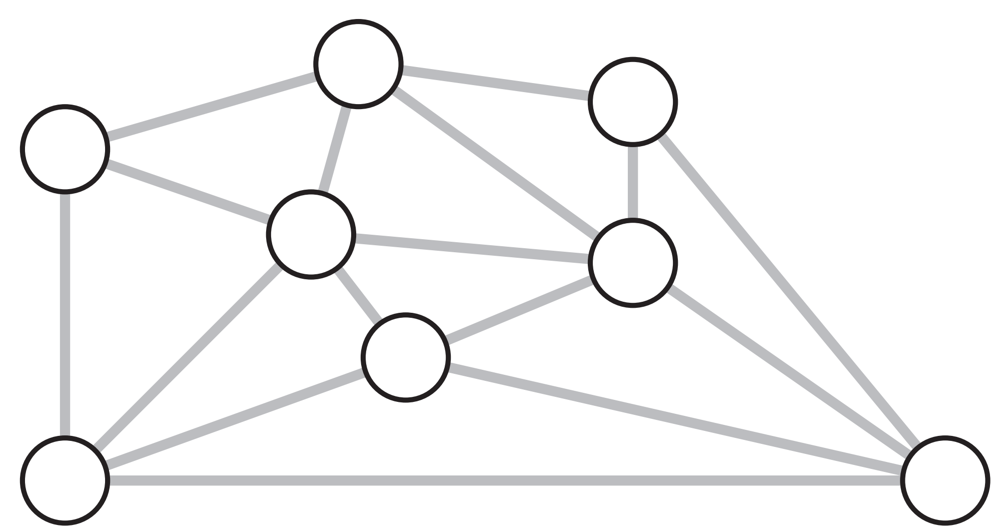
Algorithme (abstrait) glouton

- On colorie toutes les arêtes de G en gris. Puis on va progressivement colorier en noir les arêtes de ACM.
- On choisie une coupure avec aucune arête traversante noire.
- On prend l'arête traversante de poids minimum et on la colorie en noir.
- On répète jusqu'à avoir colorié ? arêtes en noir.



Algorithme (abstrait) glouton

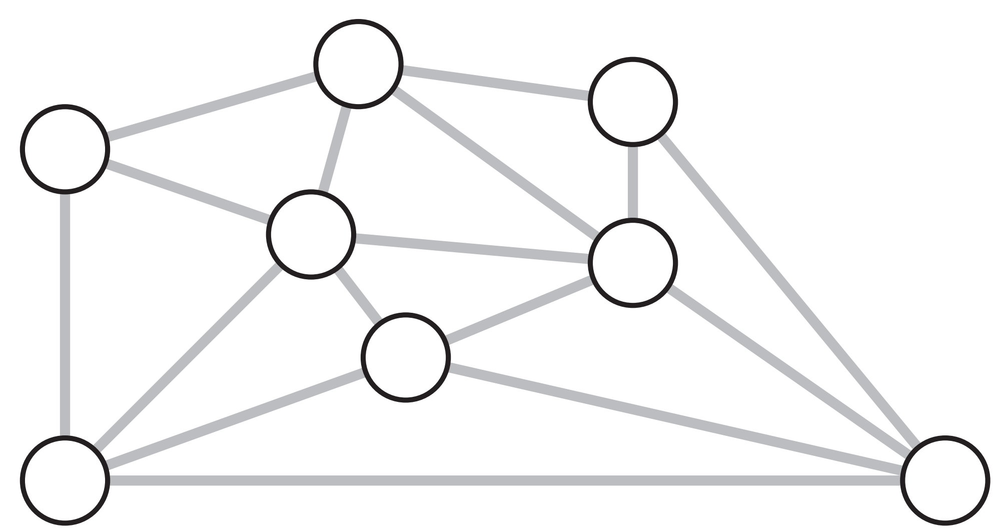
- On colorie toutes les arêtes de G en gris. Puis on va progressivement colorier en noir les arêtes de ACM.
- On choisie une coupure avec aucune arête traversante noire.
- On prend l'arête traversante de poids minimum et on la colorie en noir.
- On répète jusqu'à avoir colorié $S-1$ arêtes en noir.



Algorithme (abstrait) glouton

- On colorie toutes les arêtes de G en gris. Puis on va progressivement colorier en noir les arêtes de ACM.
- On choisie une coupure avec aucune arête traversante noire.
- On prend l’arête traversante de poids minimum et on la colorie en noir.
- On répète jusqu’à avoir colorié S-1 arêtes en noir.

un arbre (S,A) satisfait |A|=|S|-1

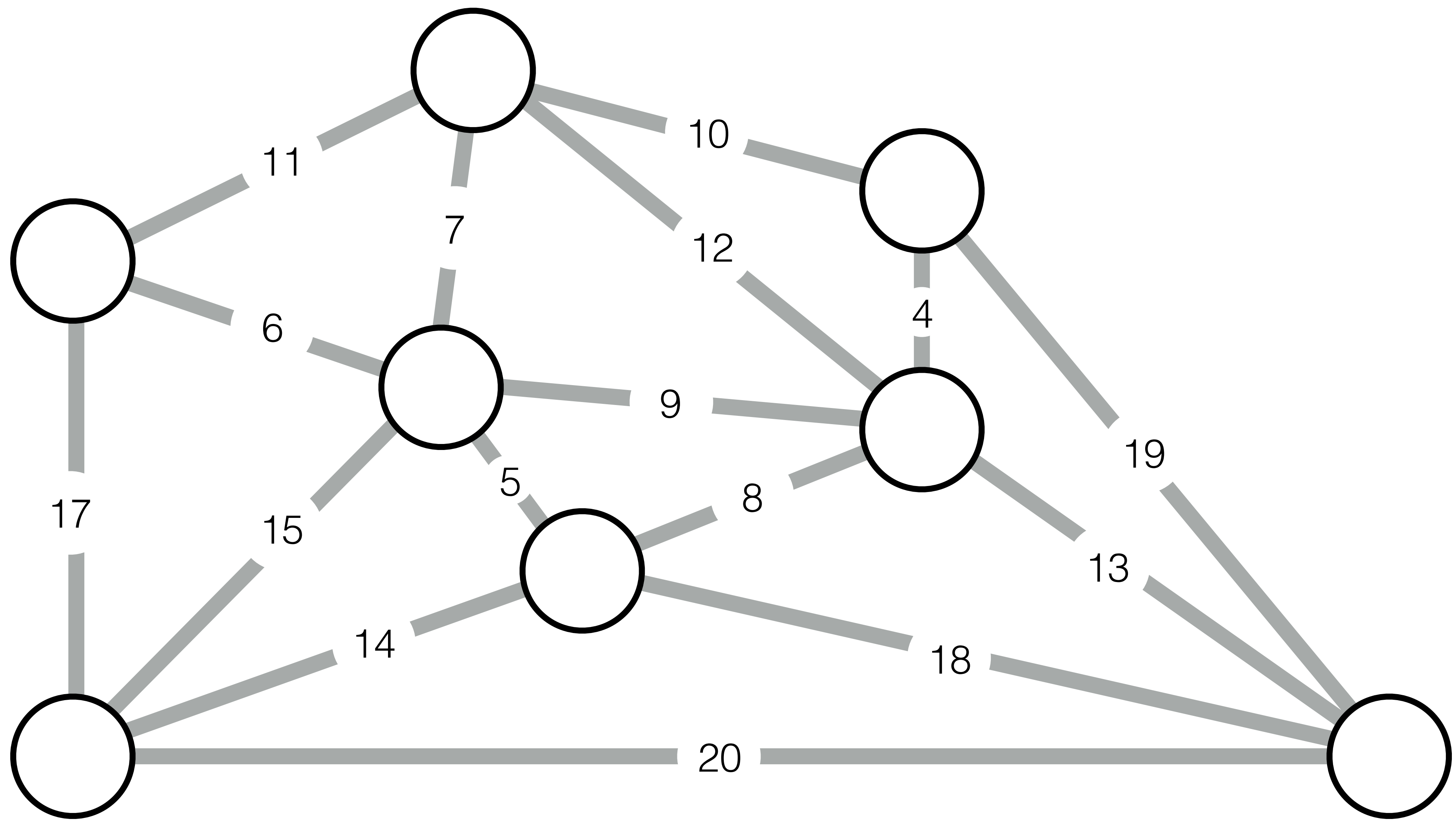


Algorithme (abstrait) glouton

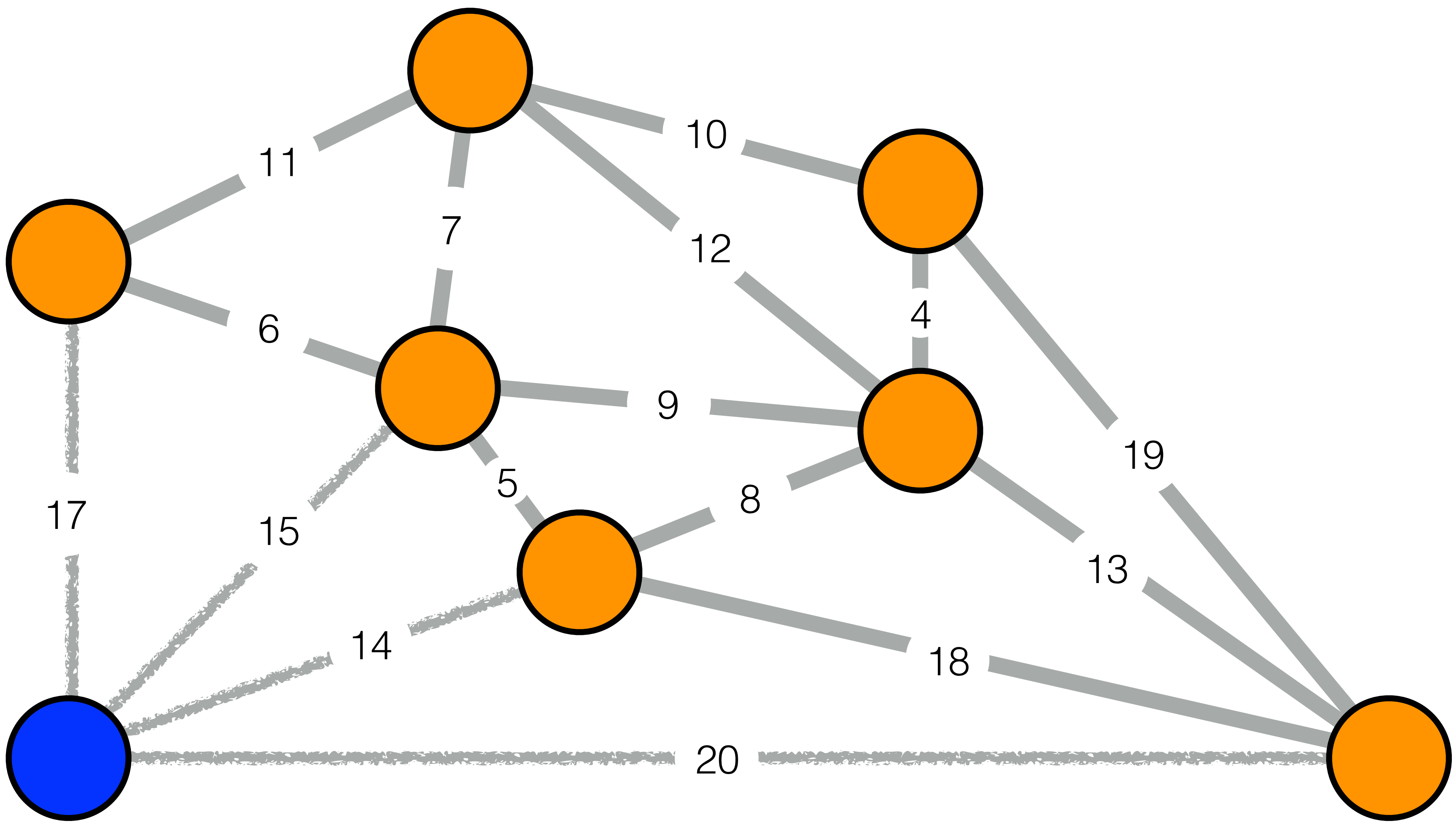
Correction

- Par la propriété de la coupure, toutes les arêtes noires sont dans ACM
- Tant qu'il y a moins de S-1 arêtes noires, on pourra toujours trouver une coupure
  - tant que la forêt noire n'est pas couvrante, ou pas connexe

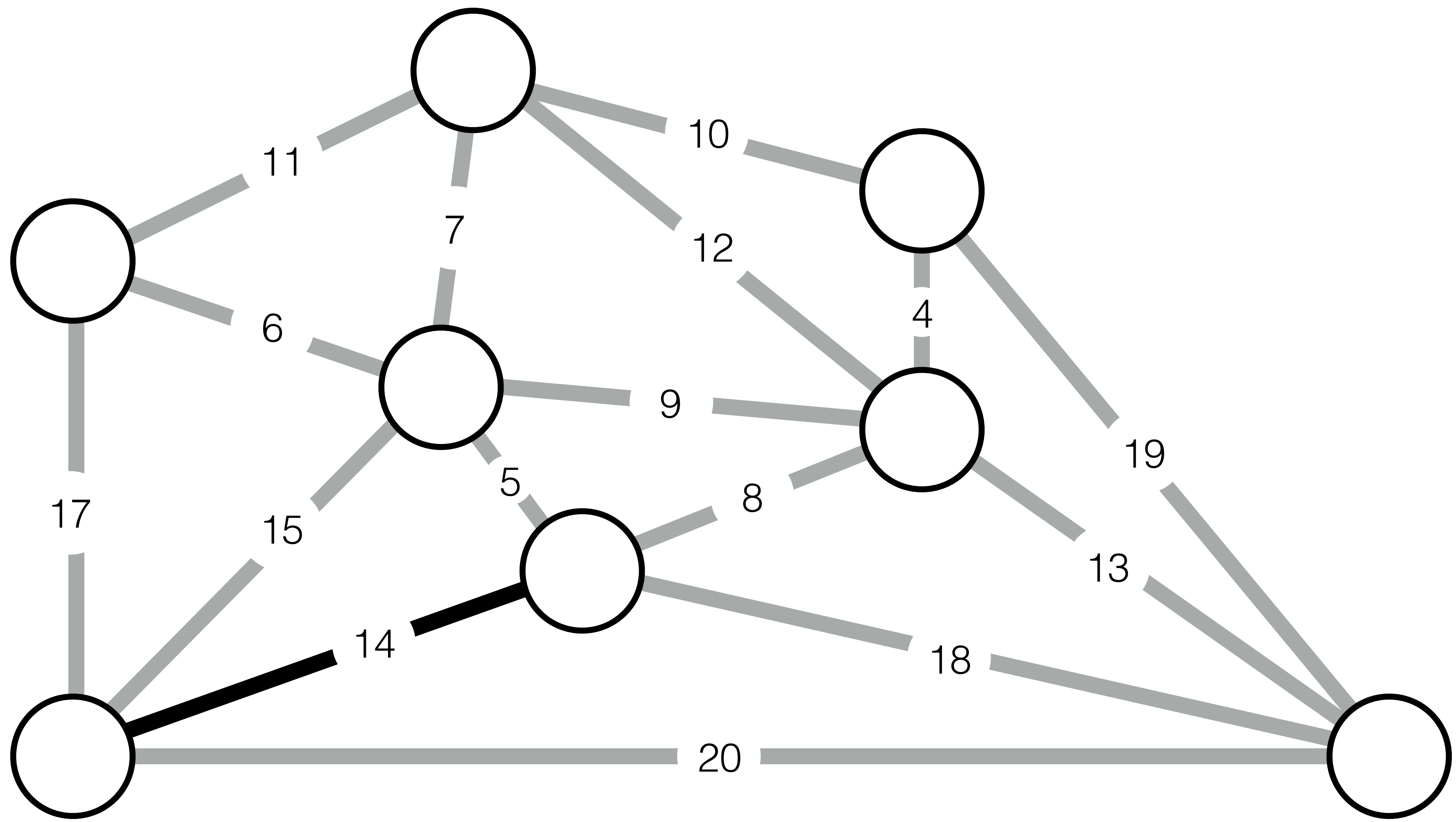
Exemple



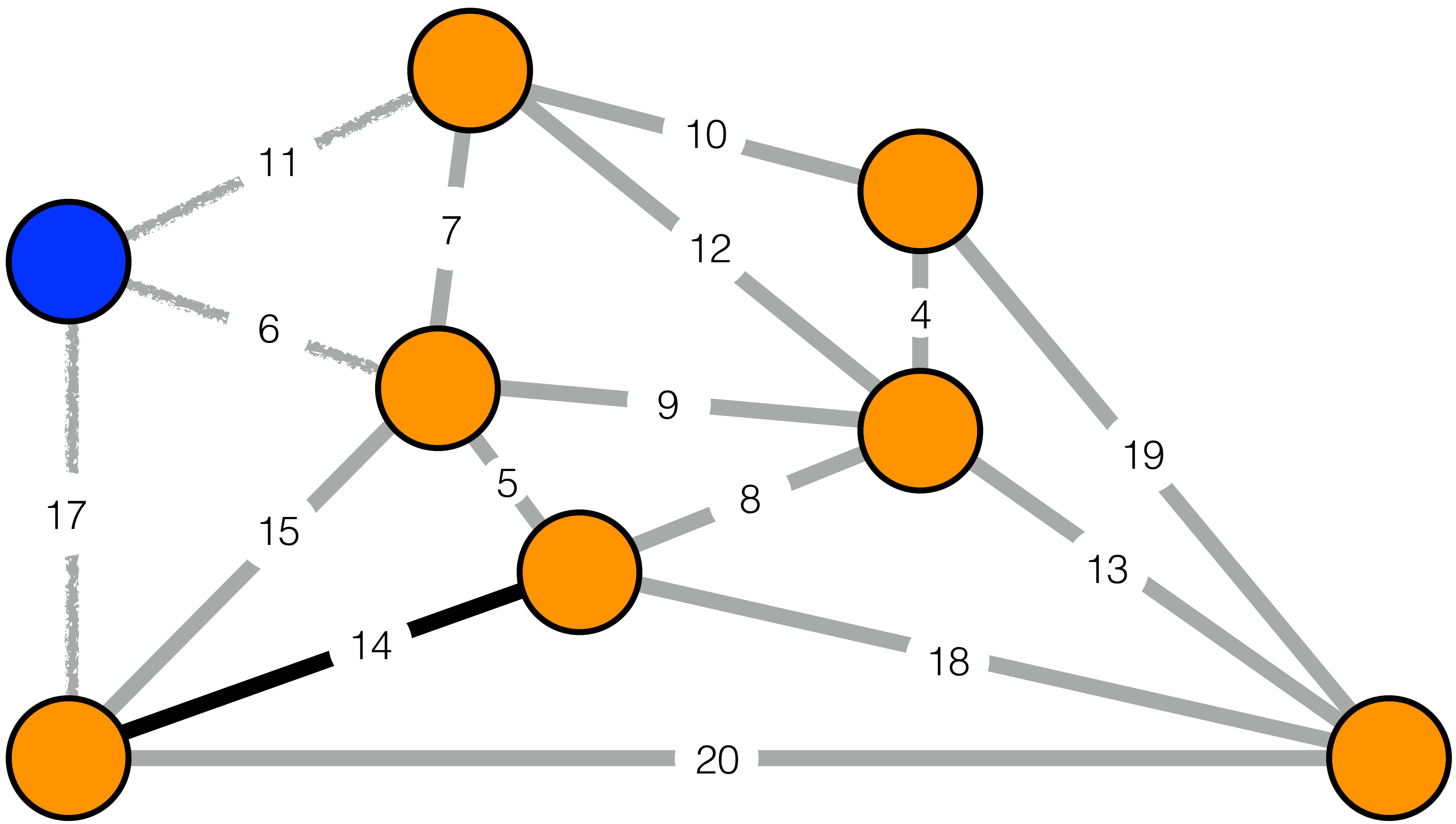
Exemple



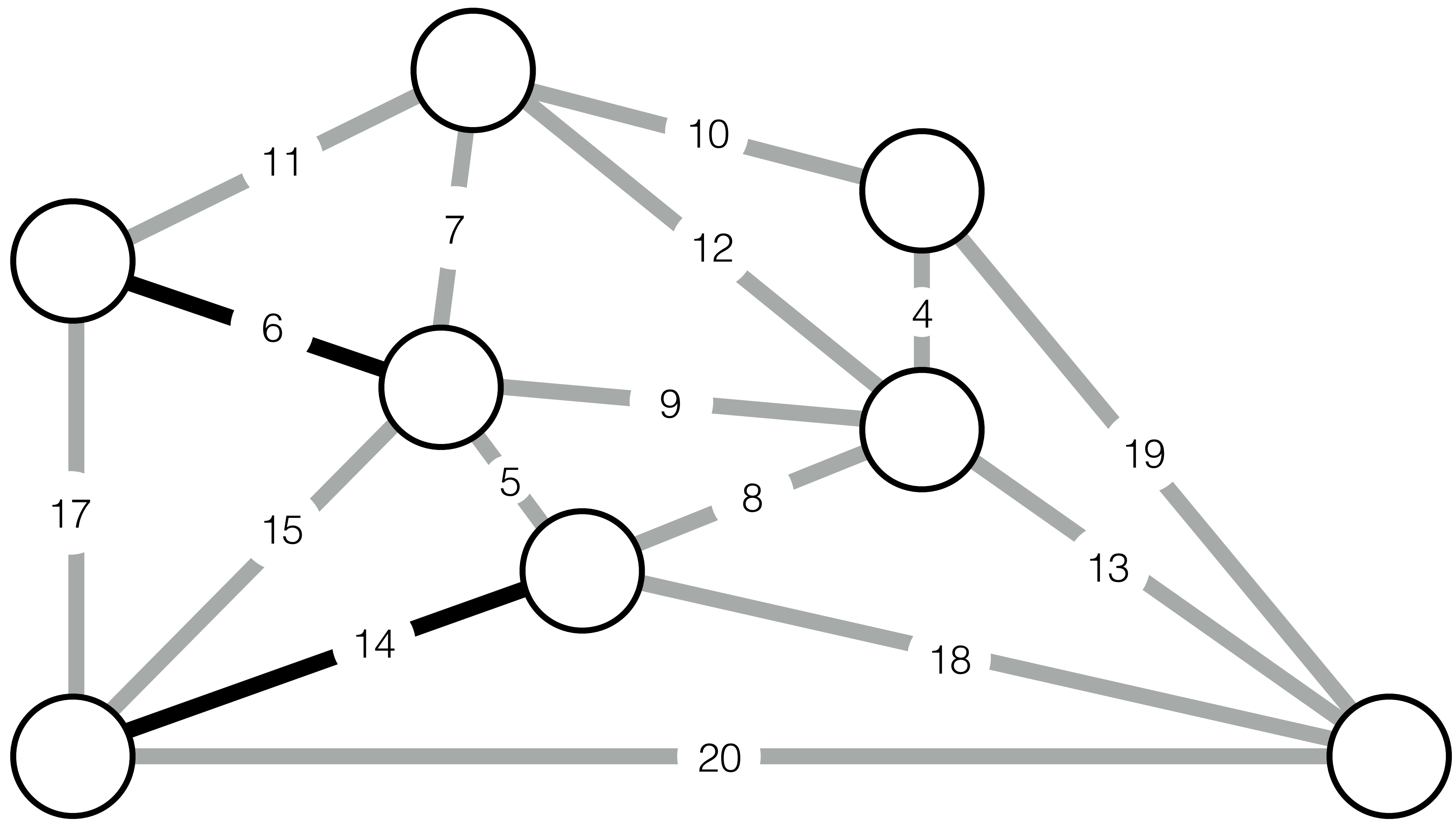
Exemple



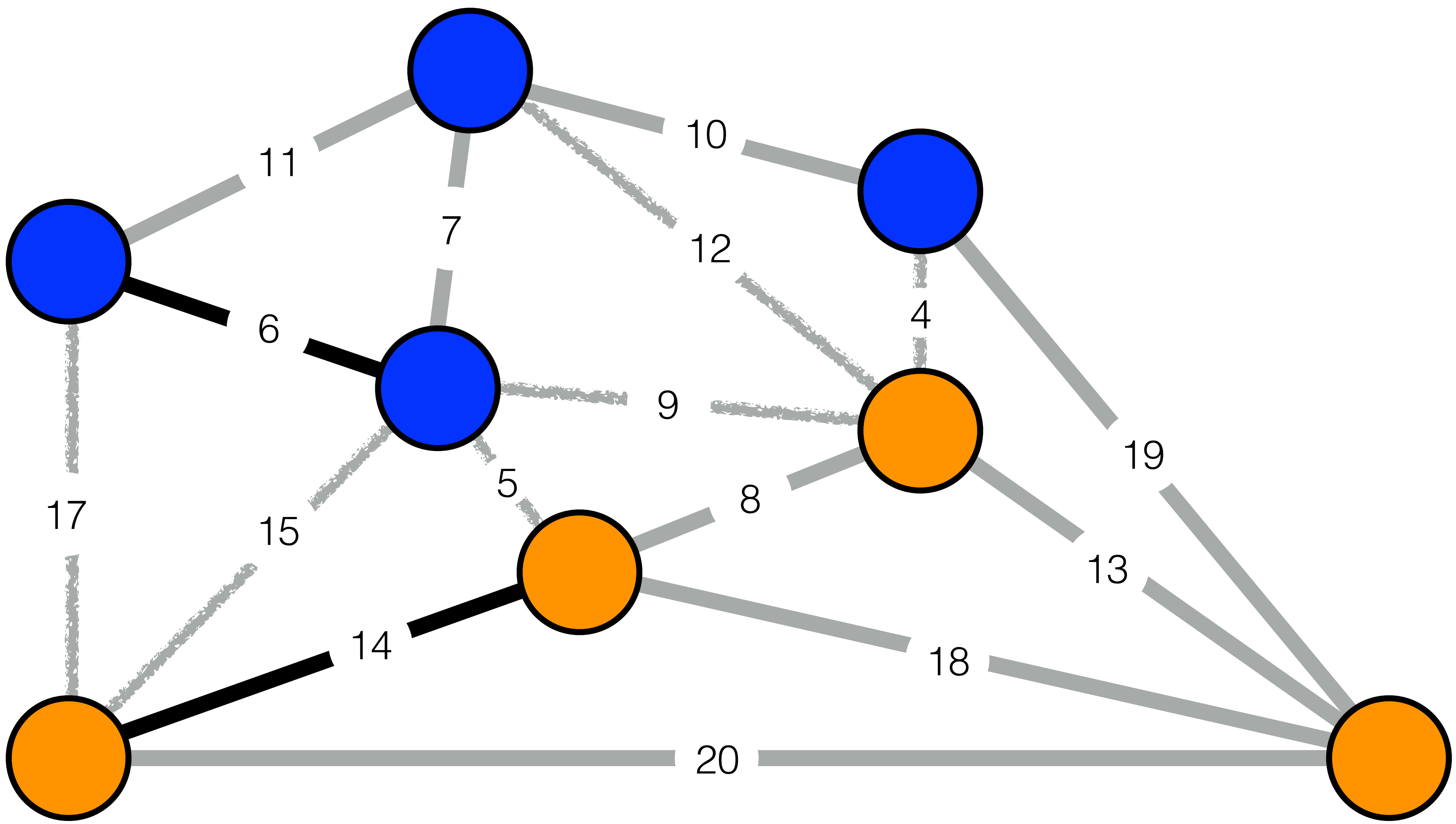
Exemple



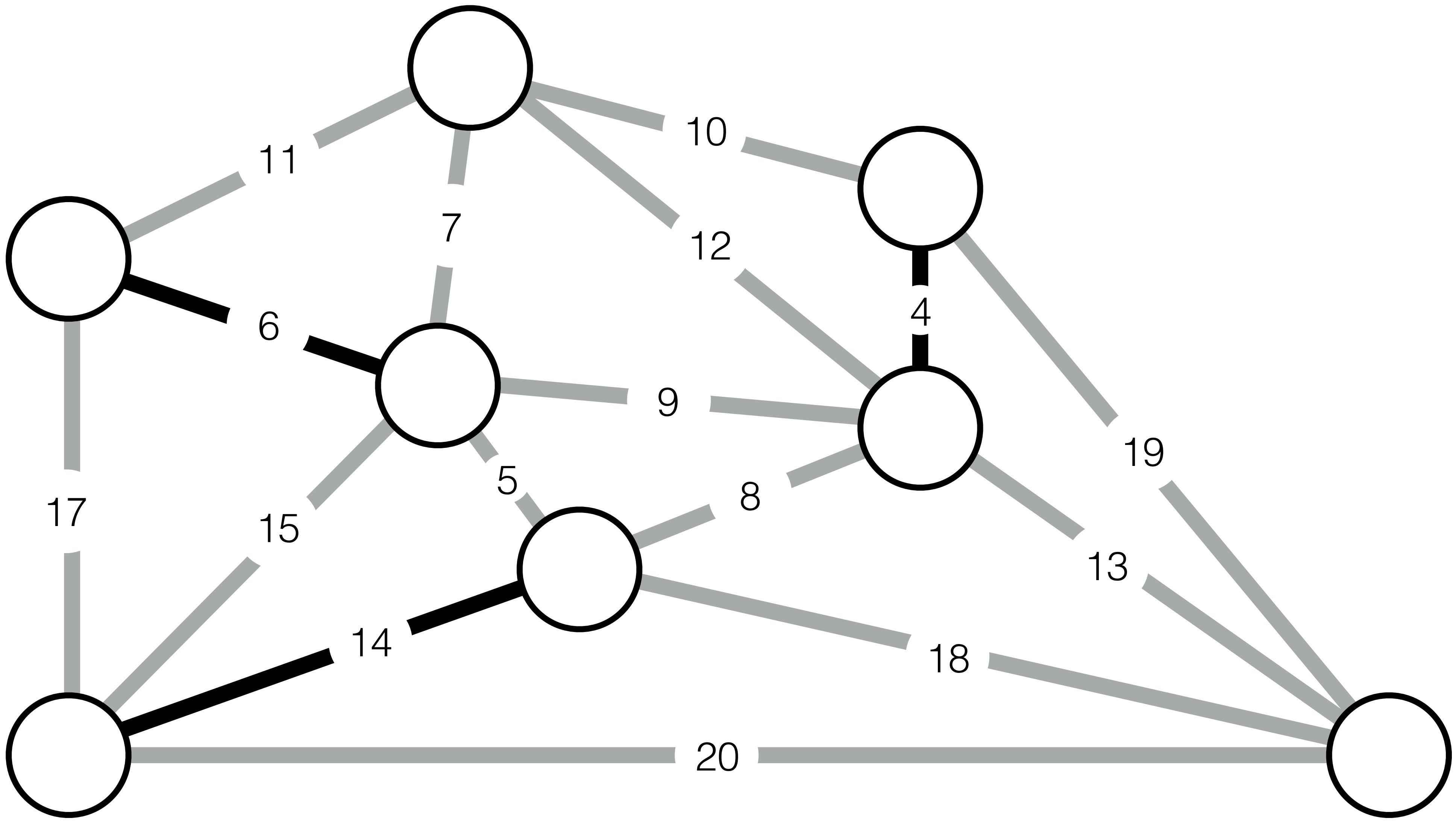
Exemple



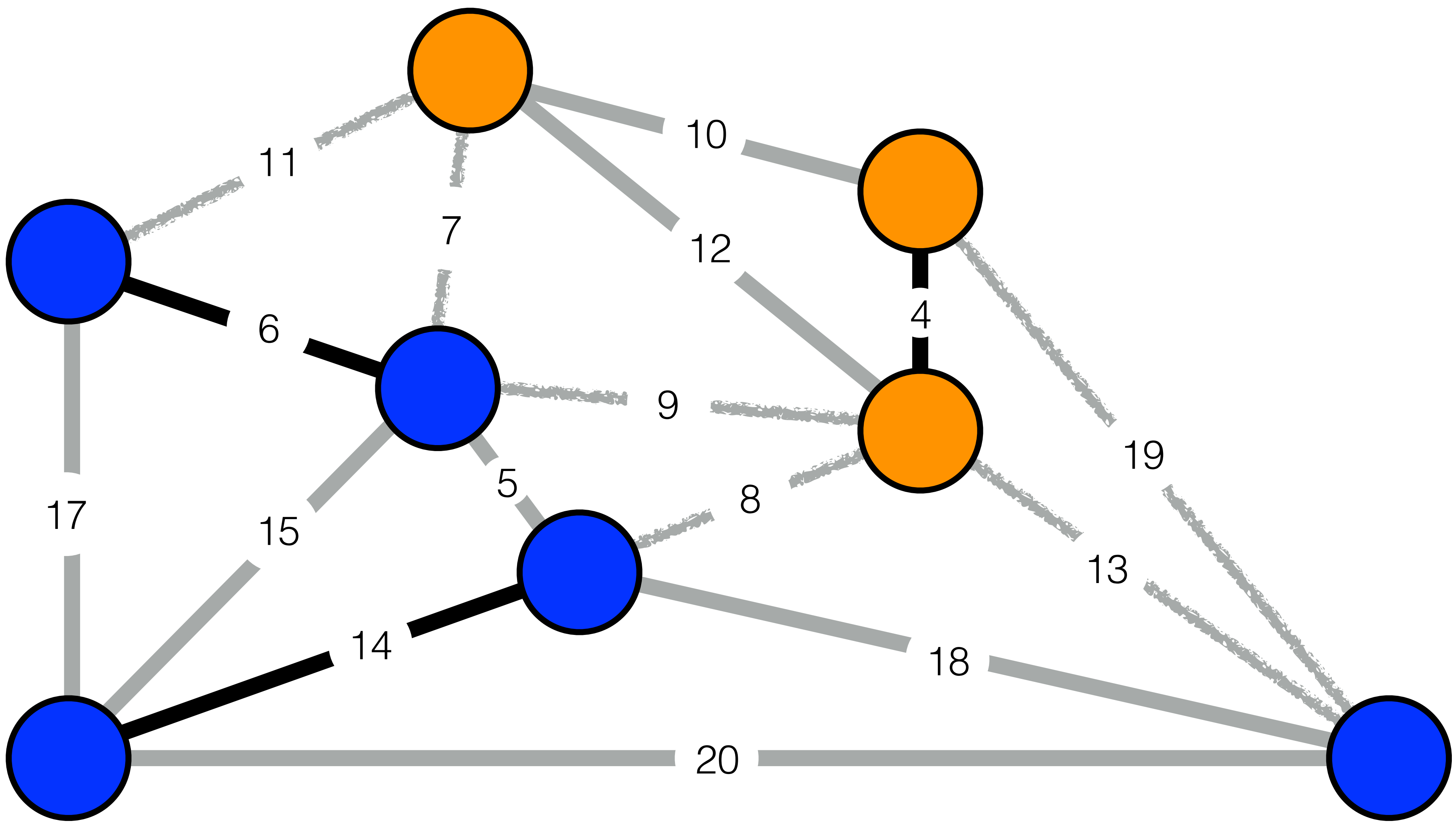
Exemple



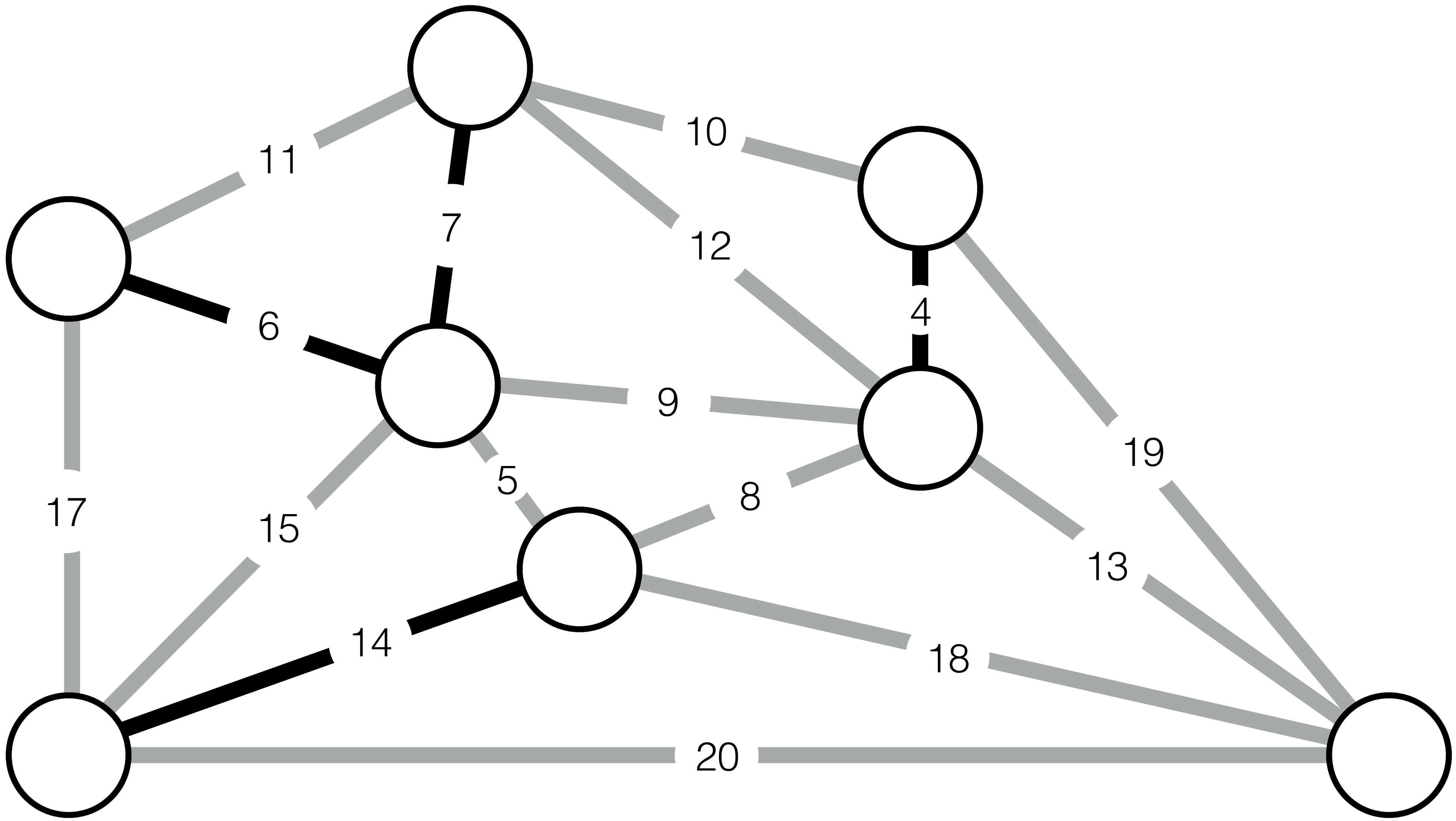
Exemple



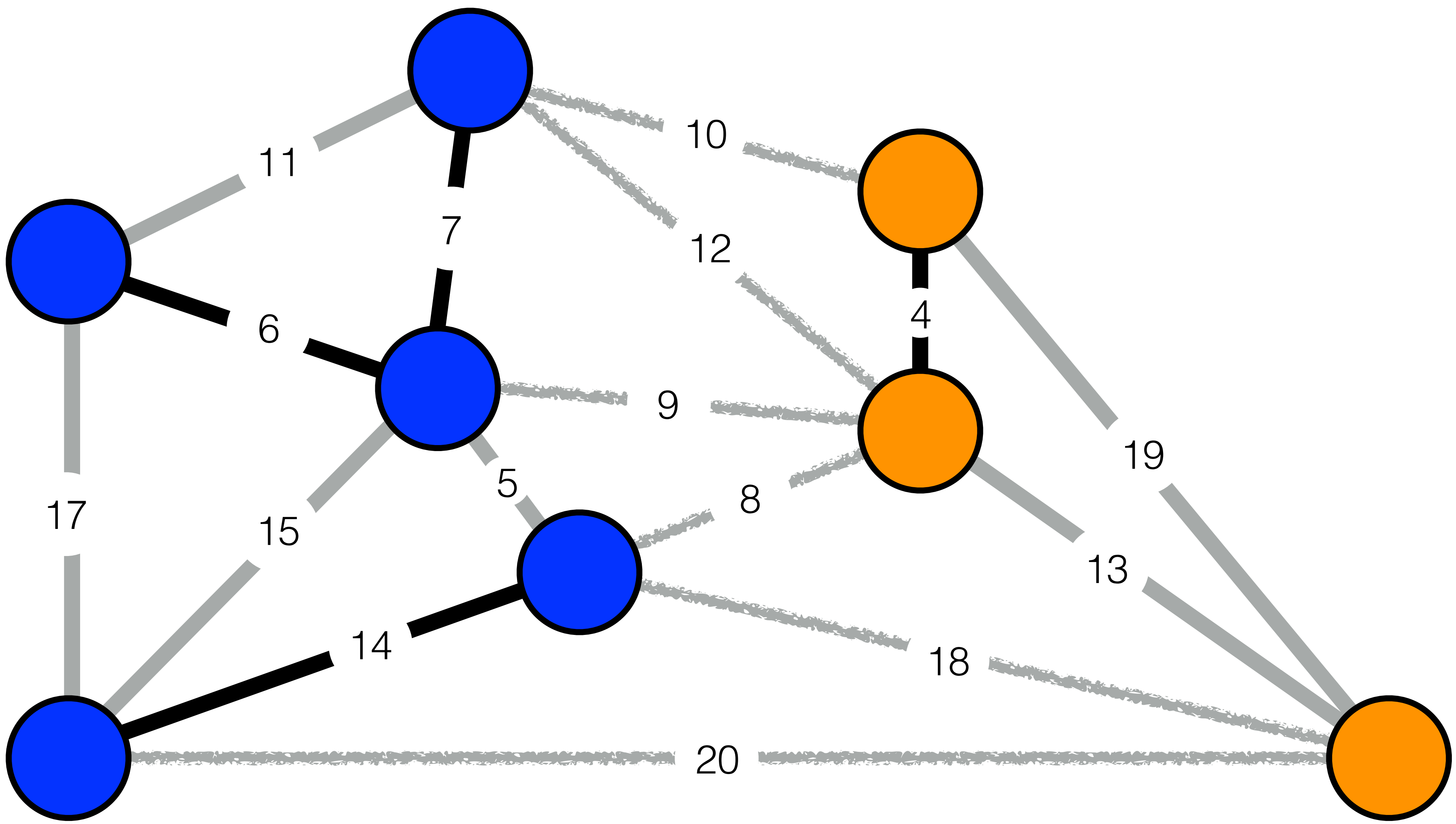
Exemple



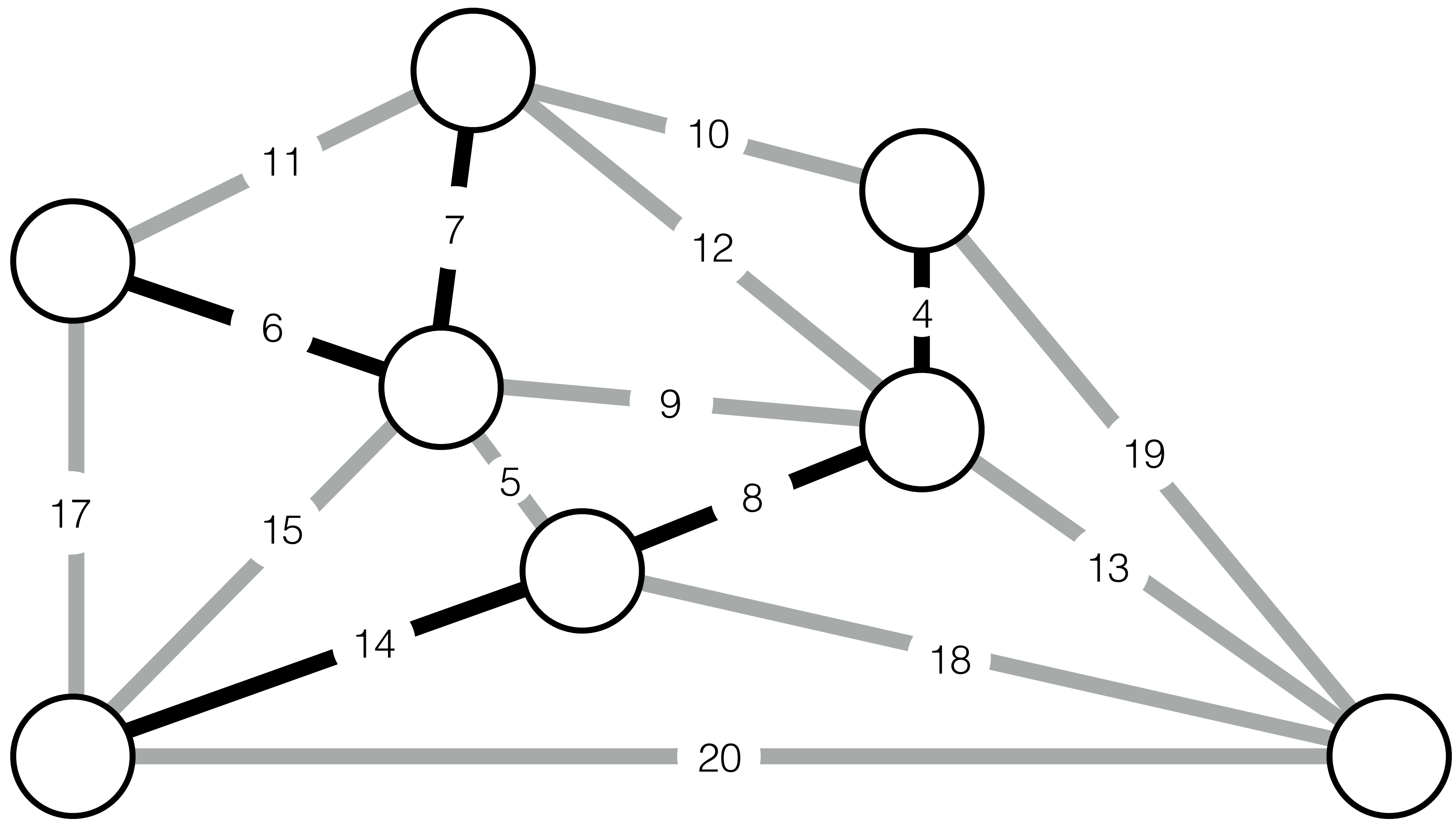
Exemple



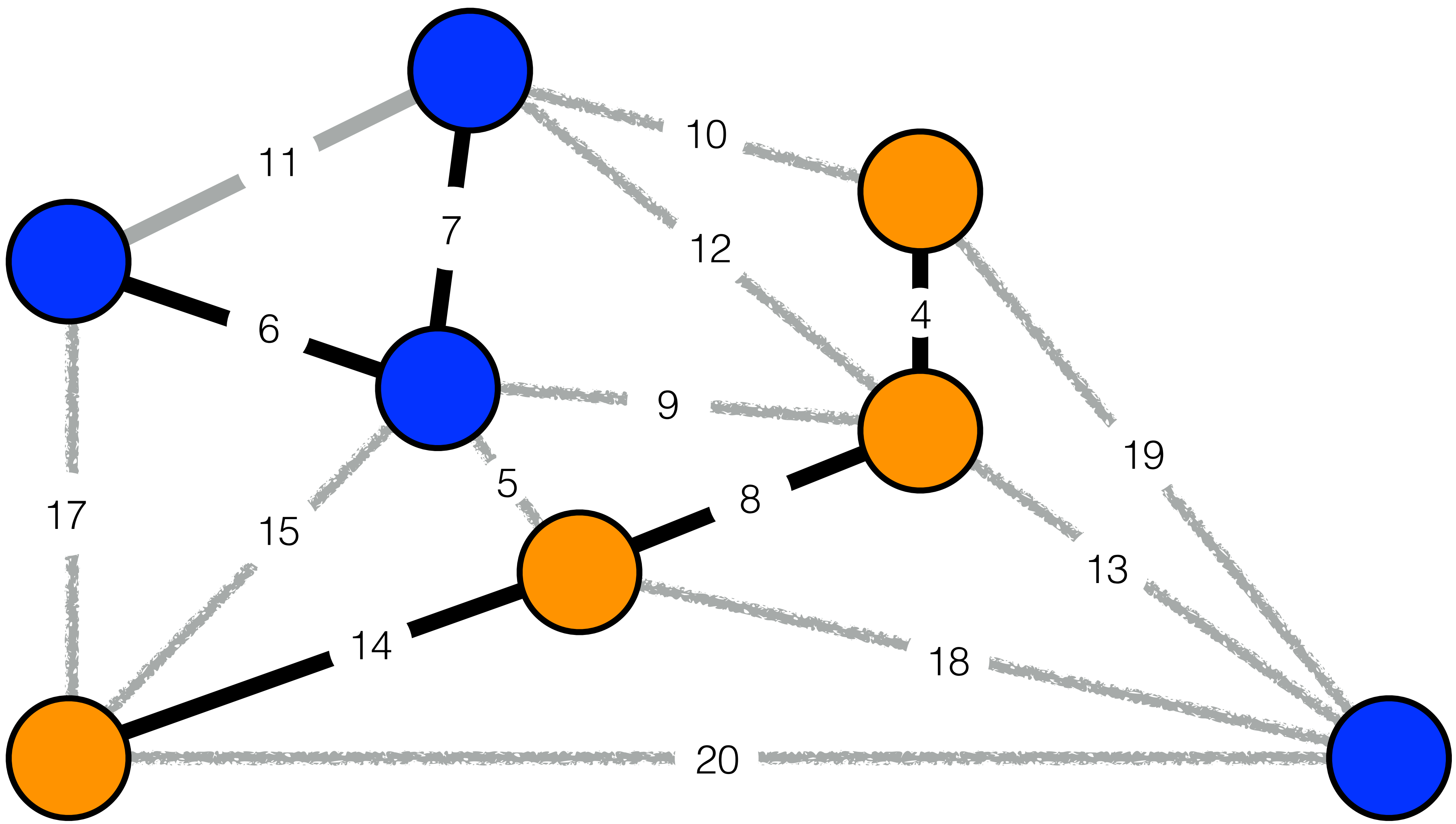
Exemple



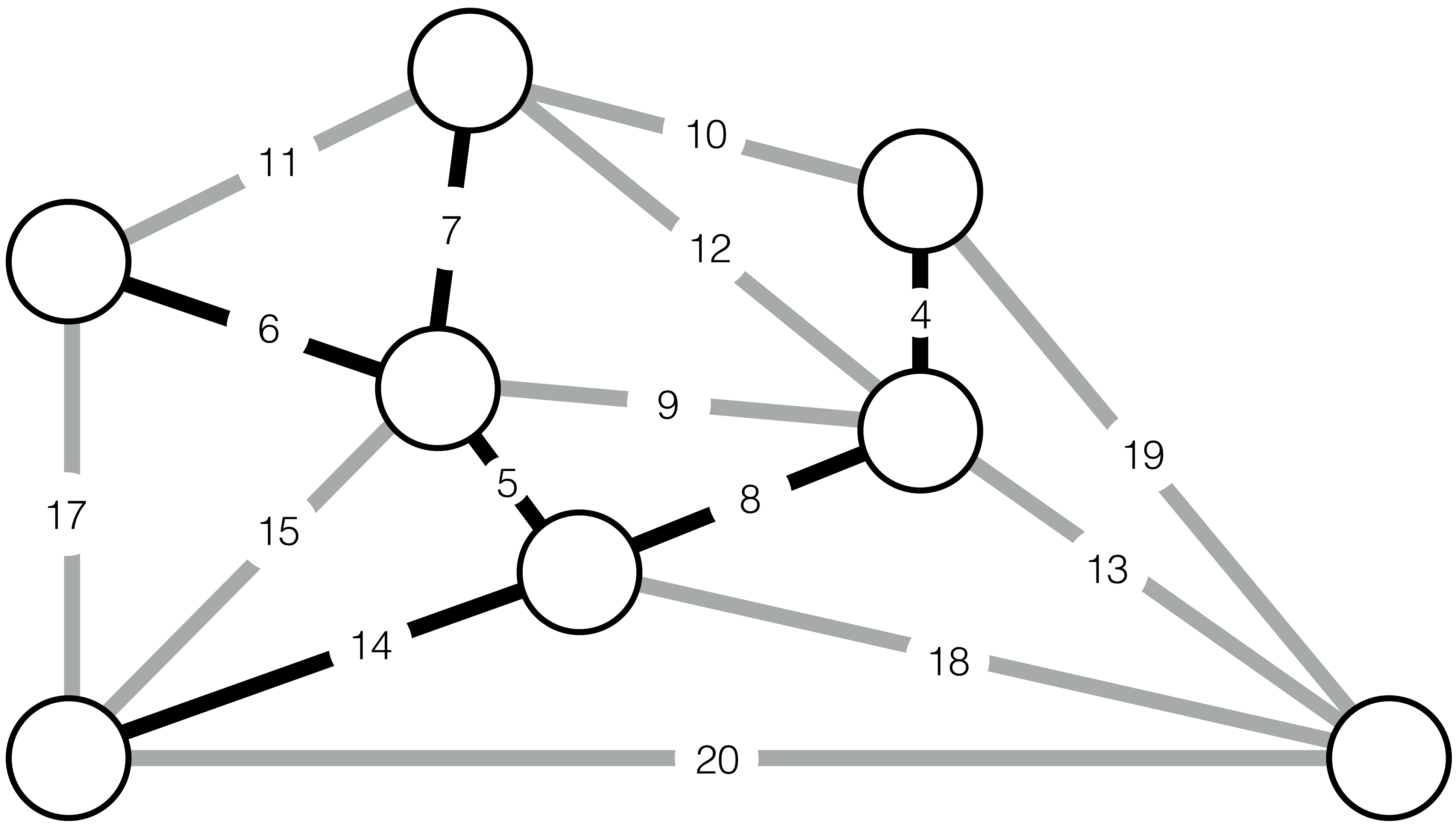
Exemple



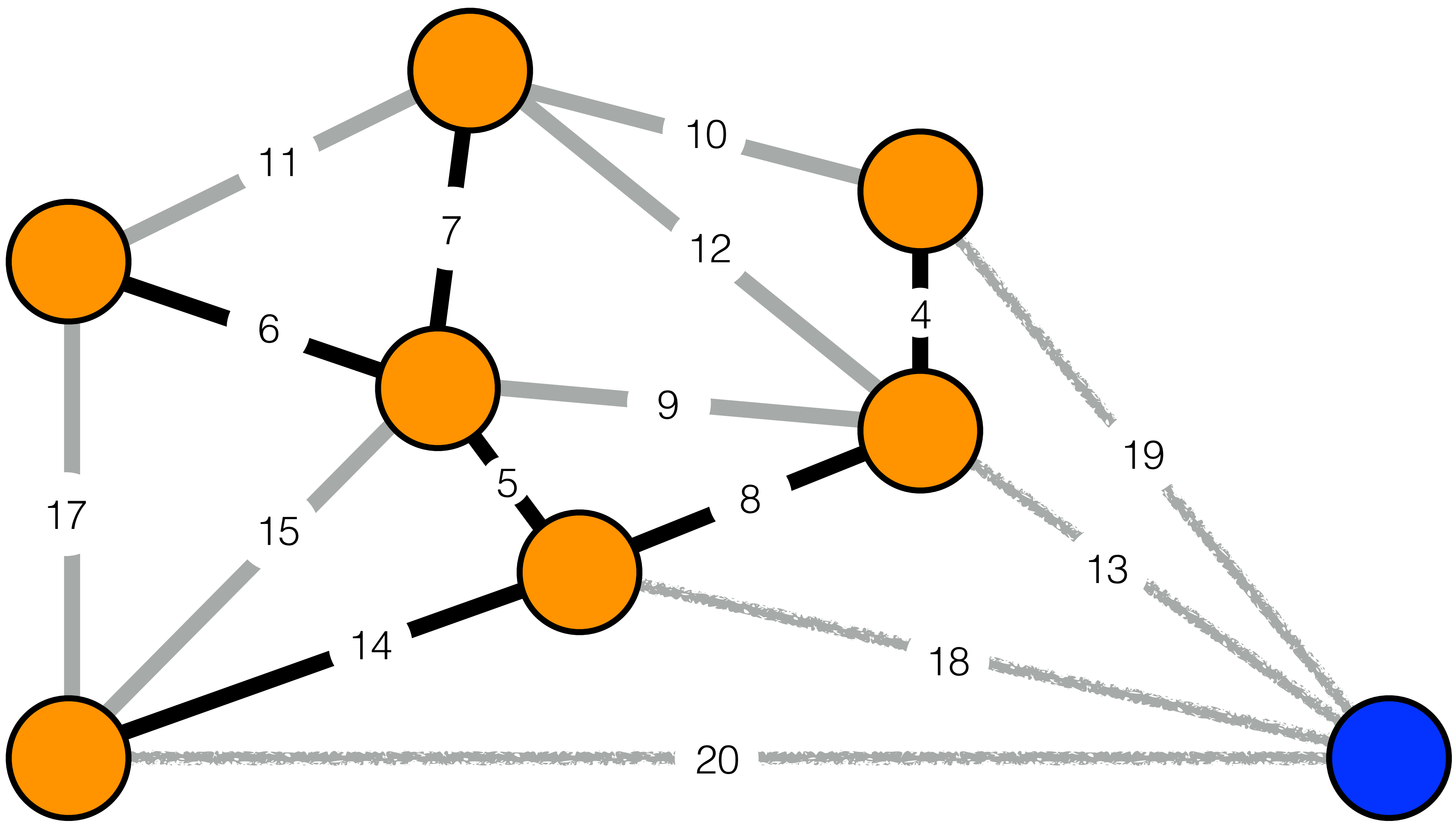
Exemple



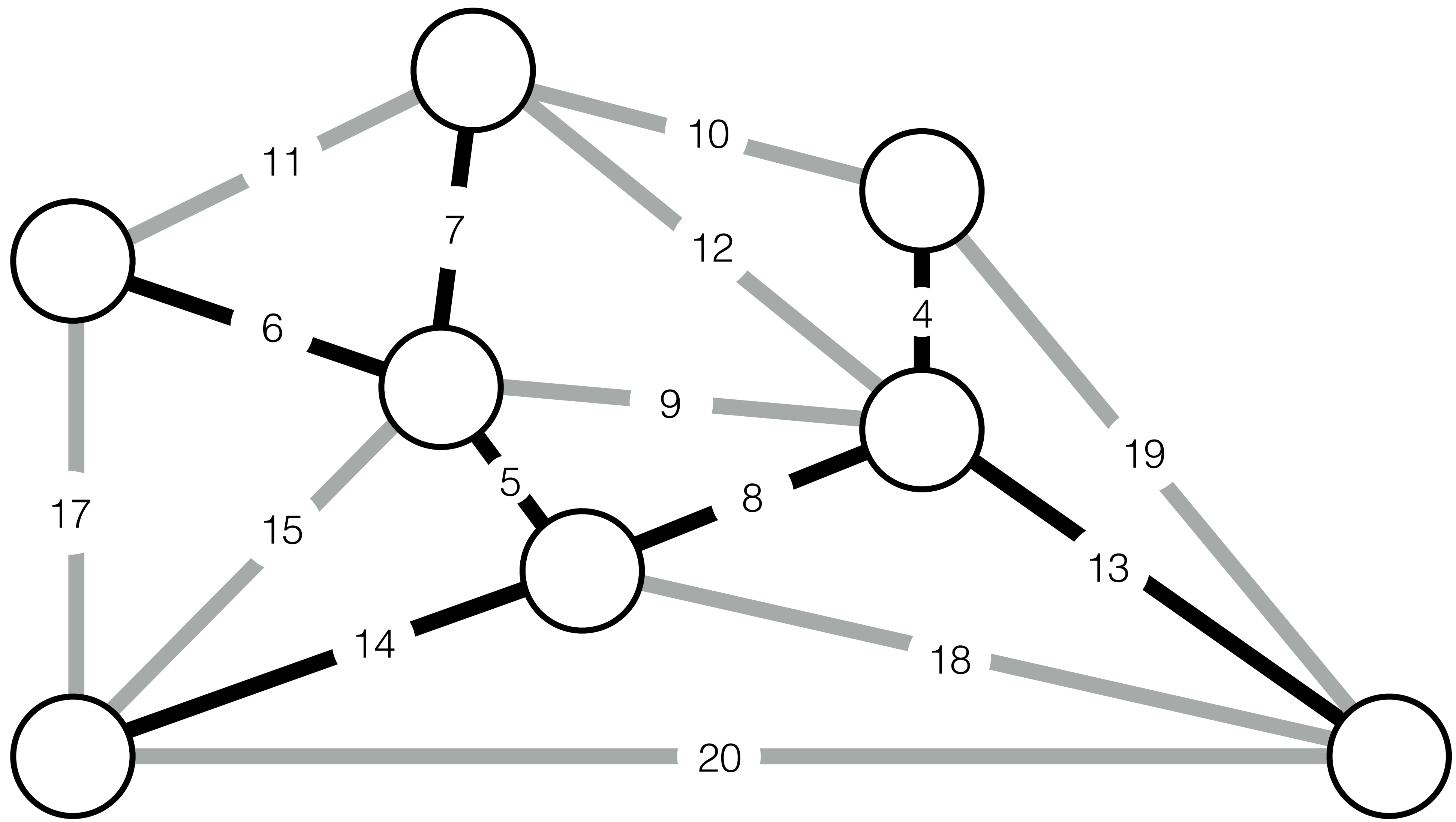
Exemple



Exemple



Exemple



Les algorithmes gloutons

- Approche algorithme générale pour résoudre des problèmes d'optimisation (trouver le plus petit …)
- On construit progressivement une solution
- A chaque étape, on fait un choix local optimal, sans se préoccuper de l'effet global
- L'approche gloutonne ne fonctionne pas toujours, mais elle est simple et efficace.

Algorithme de Prim

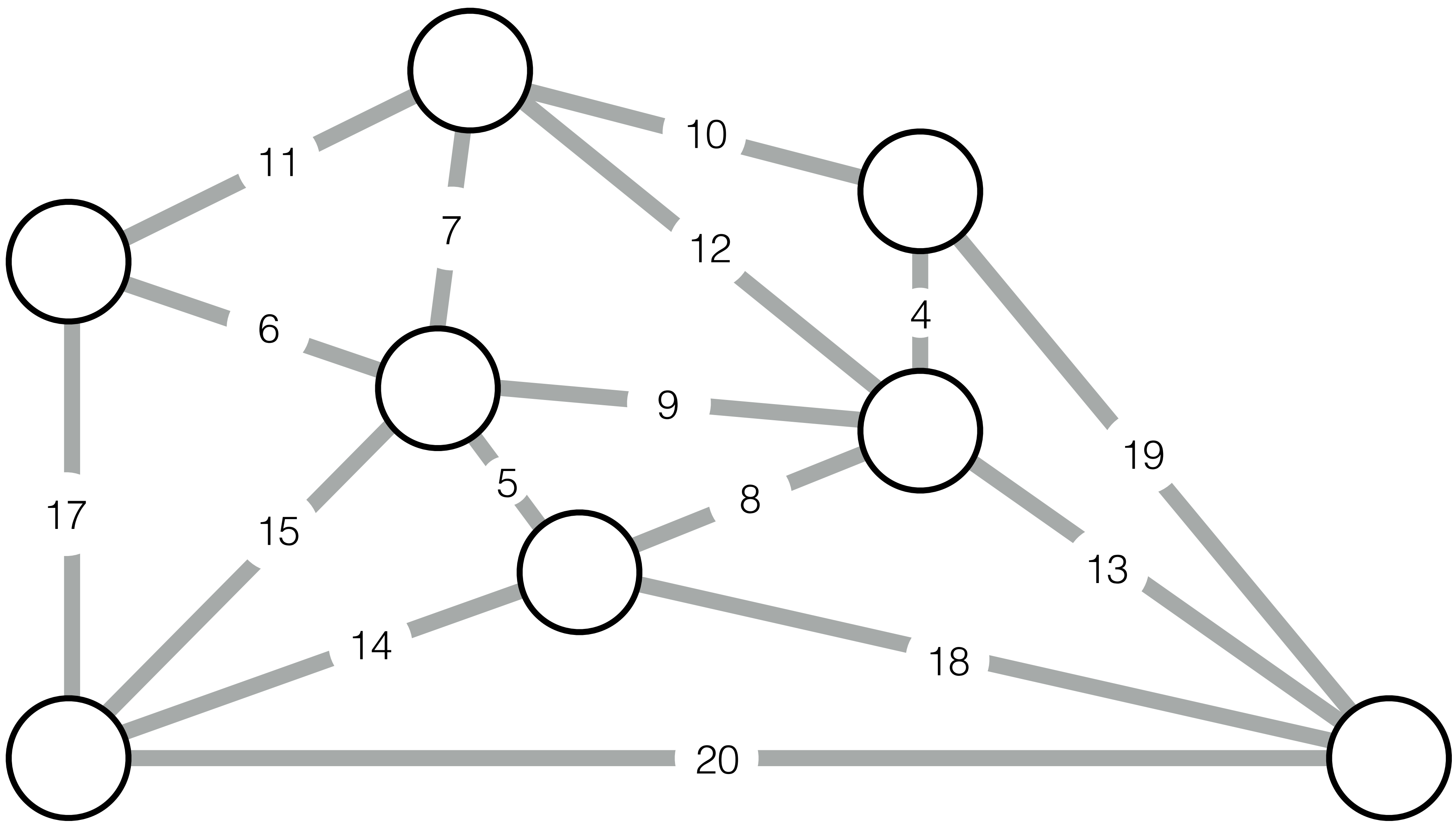
- On colorie toutes les arêtes de G en gris. Puis on va progressivement colorier en noir les arêtes de ACM. A chaque étape, le sous-graphe noir est un arbre. On colorie en noir les sommets reliés par des arêtes noirs.

- On choisie un sommet de départ.

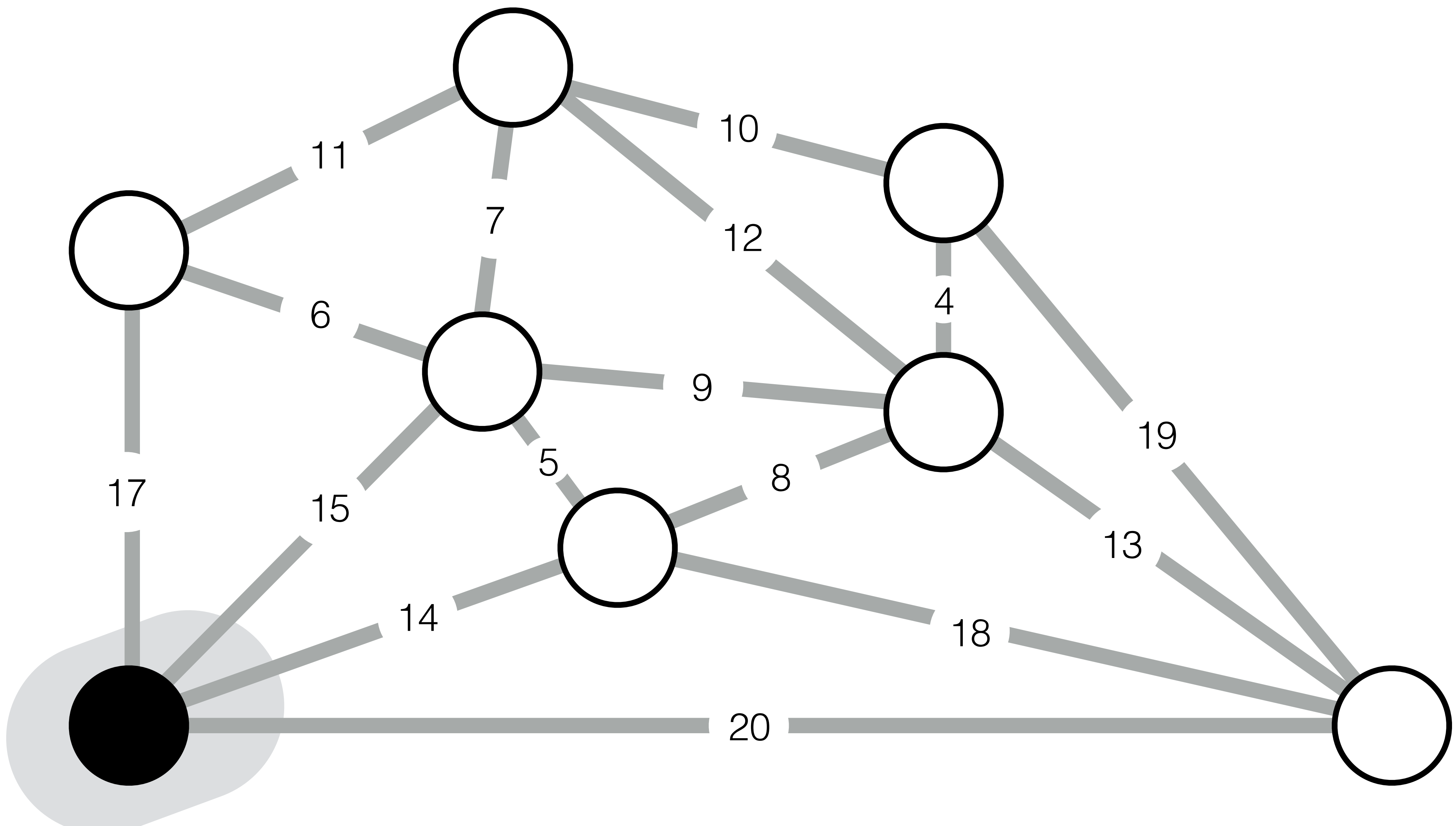
- On considère la coupure partitionnant les sommets noirs des autres. On choisi un arc traversant minimal pour cette coupure et on le colorie en noir. On colorie en noir le sommet associé qui ne l'était pas encore.

- On répète jusqu'à avoir colorié tous les sommets en noir.

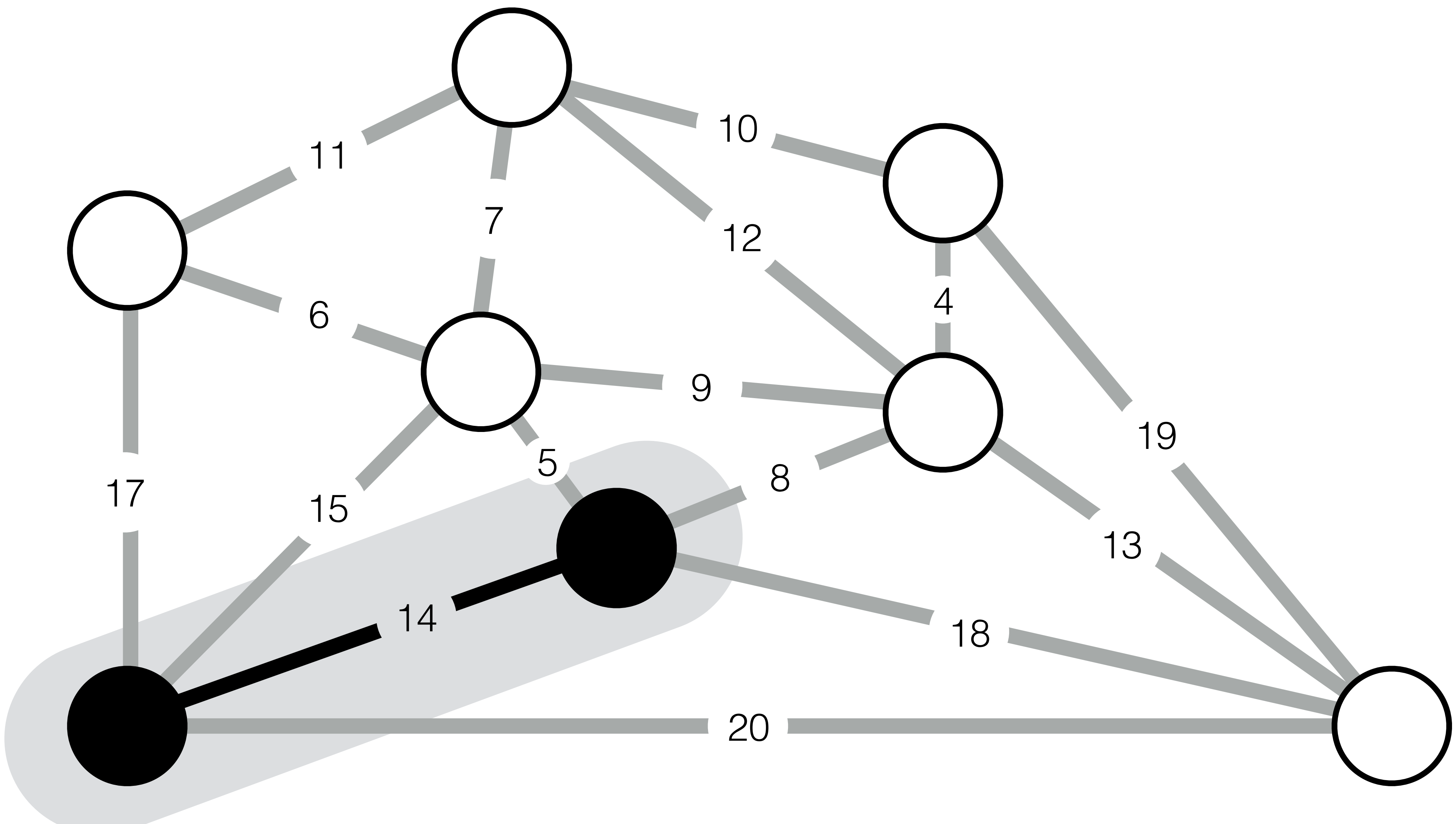
Exemple



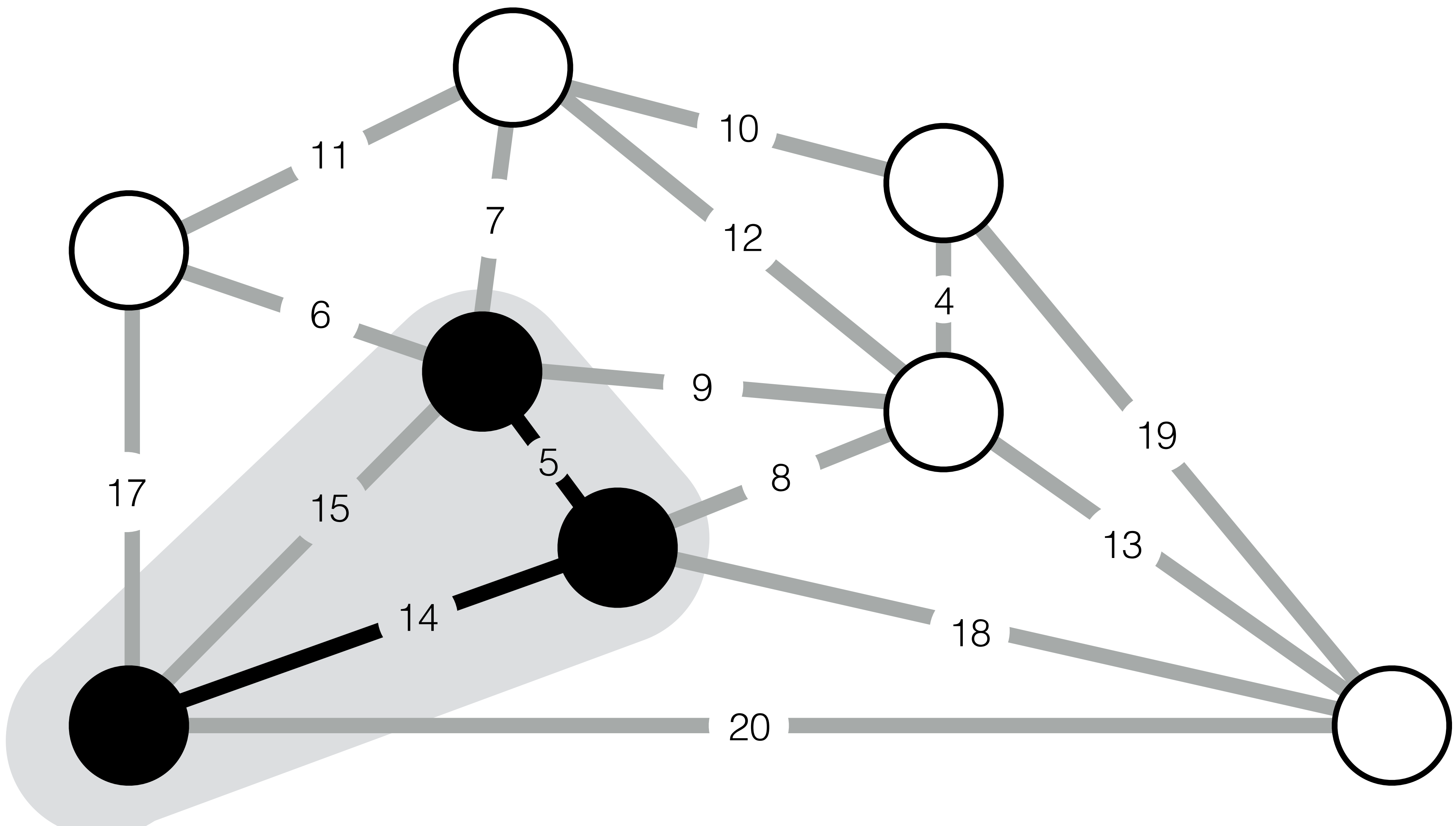
Exemple



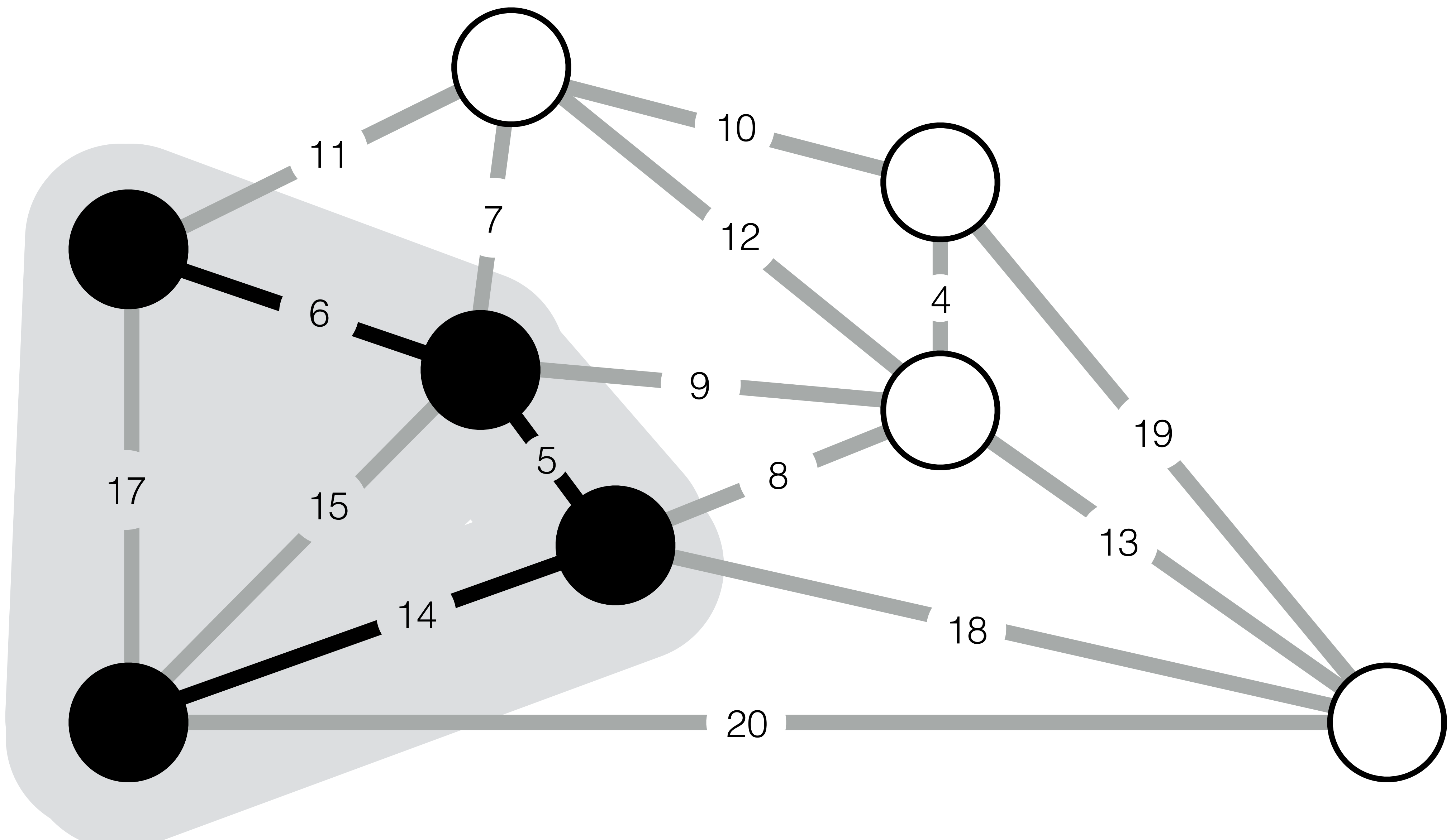
Exemple



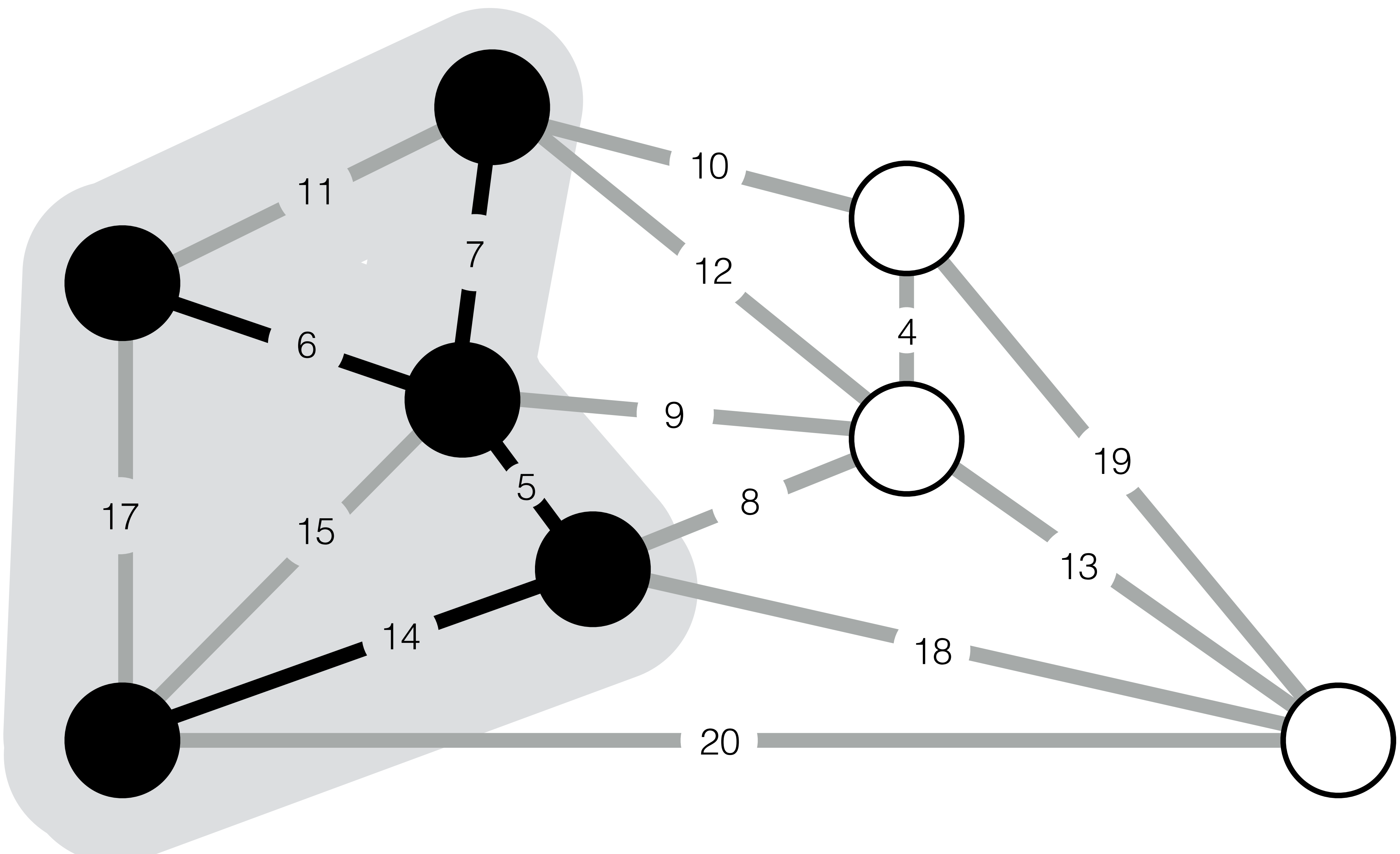
Exemple



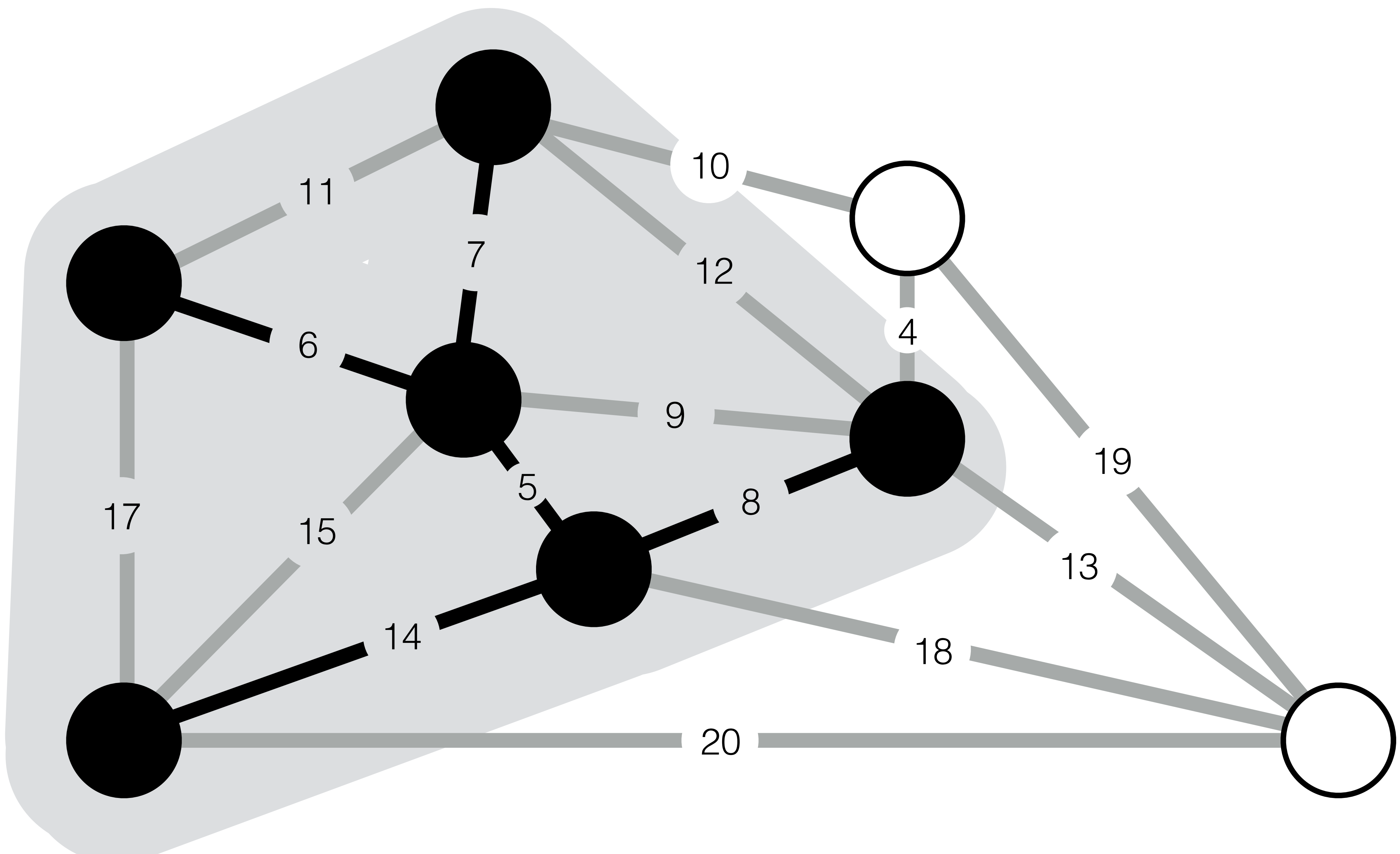
Exemple



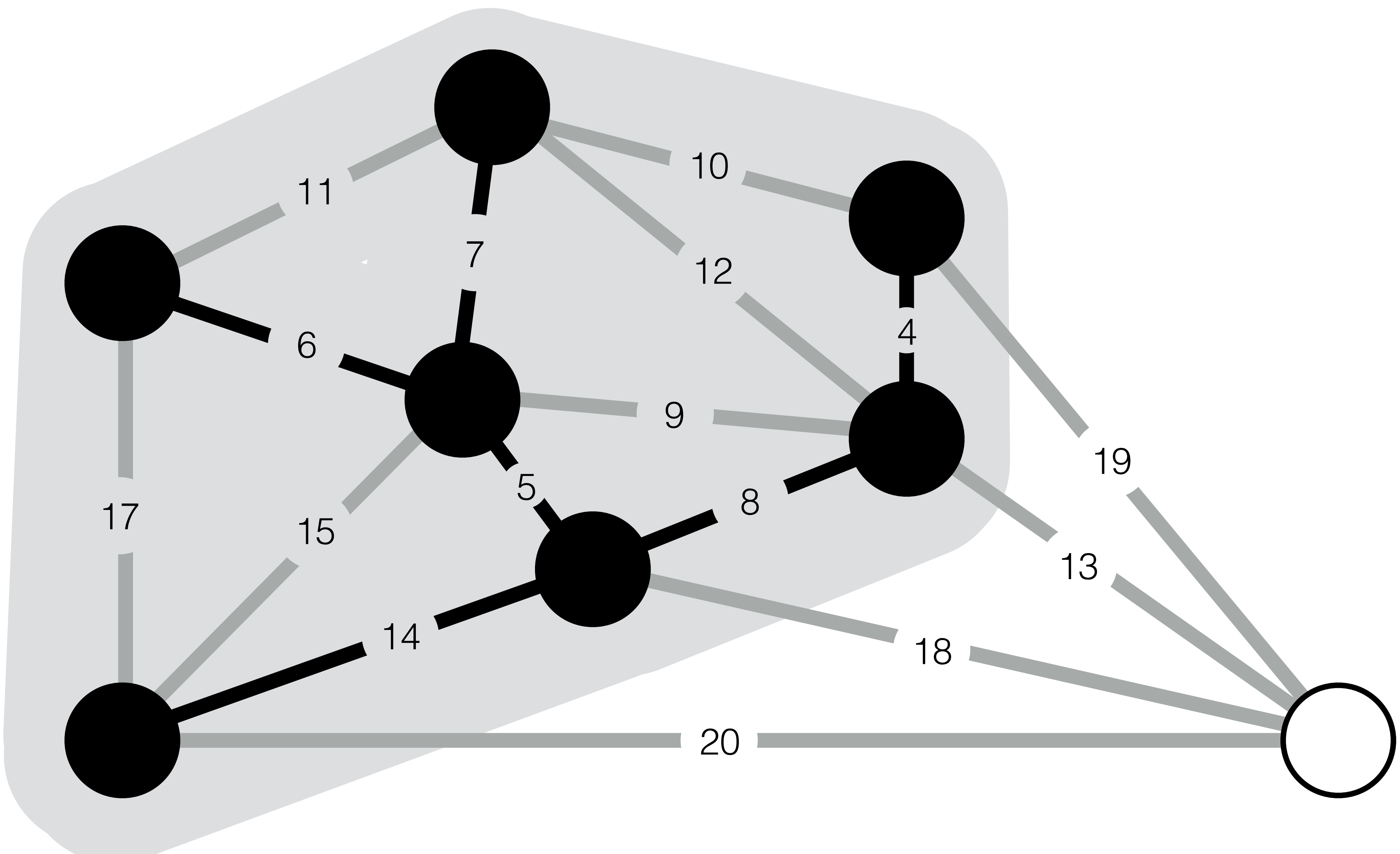
Exemple



Exemple



Exemple



Exemple