Recherche Opérationnelle

Flots et Couplages

Nadia Brauner & Alice Joffard

19 mars 2021

Université Grenoble Alpes

G-SCOP
SCIENCES POUR LA CONCEPTION, L'OPTIMISATION ET LA PRODUCTION

Enregistrement du cours

https://youtu.be/OTbC7nyJMQ4

Flot Max

Flot Max

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij

Flot Max

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Un flot, i.e. une valeur entière f_{ij} sur les arcs tel que:

Flot Max

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Un flot, i.e. une valeur entière f_{ij} sur les arcs tel que:
  - Admissibilité: Pour chaque arc, le flot n'excède pas la capacité

Flot Max

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Un flot, i.e. une valeur entière f_{ij} sur les arcs tel que:
  - Admissibilité: Pour chaque arc, le flot n'excède pas la capacité
  - Conservation: Pour chaque sommet qui n'est pas la source ni le puit, le flot entrant est égal au flot sortant

Flot Max

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Un flot, i.e. une valeur entière f_{ij} sur les arcs tel que:
  - Admissibilité: Pour chaque arc, le flot n'excède pas la capacité
  - Conservation: Pour chaque sommet qui n'est pas la source ni le puit, le flot entrant est égal au flot sortant
- Objectif: Maximiser la valeur du flot, i.e. le flot qui sort de s

Le dual: Coupe Min

Le dual: Coupe Min

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij

Le dual: Coupe Min

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Une coupe, i.e. une partition des sommets en deux parties X et Y, $s \in X, p \in Y$

Le dual: Coupe Min

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Une coupe, i.e. une partition des sommets en deux parties X et Y, $s \in X, p \in Y$
- Objectif: Minimiser la valeur de la coupe, i.e. la somme des capacités des arcs qui sortent de X

Le dual: Coupe Min

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Une coupe, i.e. une partition des sommets en deux parties X et Y, $s \in X, p \in Y$
- Objectif: Minimiser la valeur de la coupe, i.e. la somme des capacités des arcs qui sortent de X
- Théorème de dualité forte: La valeur du flot max est égale à la valeur de la coupe min

Le dual: Coupe Min

- Entrée: Un graphe orienté $G = (V, A)$ sans cycle, un sommet source s, un sommet puit p, une capacité c_{ij} sur chaque arc ij
- Sortie: Une coupe, i.e. une partition des sommets en deux parties X et Y, $s \in X, p \in Y$
- Objectif: Minimiser la valeur de la coupe, i.e. la somme des capacités des arcs qui sortent de X
- Théorème de dualité forte: La valeur du flot max est égale à la valeur de la coupe min
- A rendre pour le 2/04: Prouvez que le dual de Flot Max est Coupe Min. Vous pourrez considérer qu'il y a un arc de t jusqu'à s avec une capacité infinie, qu'il y a conservation du flot en s et t également, et que l'on cherche ainsi à maximiser le flot f_{ts}.

Complexité

Complexité

- Pourriez-vous de manière générale résoudre flot max ou coupe min graphiquement ?

Complexité

- Pourriez-vous de manière générale résoudre flot max ou coupe min graphiquement ?

$\rightarrow$ Non, il peut y avoir un nombre arbitraire de variables !

Complexité

- Pourriez-vous de manière générale résoudre flot max ou coupe min graphiquement ?

$\rightarrow$ Non, il peut y avoir un nombre arbitraire de variables !

- Et avec la méthode du simplexe ?

Complexité

- Pourriez-vous de manière générale résoudre flot max ou coupe min graphiquement ?

→ Non, il peut y avoir un nombre arbitraire de variables !

- Et avec la méthode du simplexe ?

→ Non, les variables sont entières !

Complexité

- Pourriez-vous de manière générale résoudre flot max ou coupe min graphiquement ?

→ Non, il peut y avoir un nombre arbitraire de variables !

- Et avec la méthode du simplexe ?

→ Non, les variables sont entières !

- Comme ce sont des PLNE, ces problèmes pourraient être NP-difficiles

Complexité

- Pourriez-vous de manière générale résoudre flot max ou coupe min graphiquement ?

→ Non, il peut y avoir un nombre arbitraire de variables !

- Et avec la méthode du simplexe ?

→ Non, les variables sont entières !

- Comme ce sont des PLNE, ces problèmes pourraient être NP-difficiles
- Mais pour ces PLNE là, il existe un algorithme polynomial !

Vocabulaire

Vocabulaire

- Arc saturé: $f_{ij} = c_{ij}$

Vocabulaire

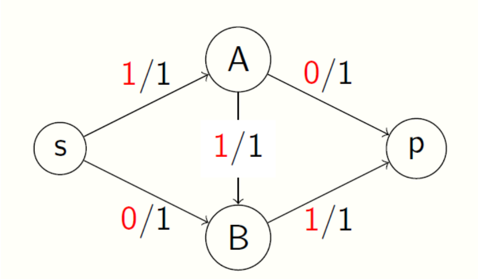
- Arc saturé: $f_{ij} = c_{ij}$
- Flot complet: Tous les chemins de s à p ont un arc saturé

Vocabulaire

- Arc saturé: $f_{ij} = c_{ij}$
- Flot complet: Tous les chemins de s à p ont un arc saturé
- Flot maximum: Flot dont la valeur est maximum

Vocabulaire

- Arc saturé: $f_{ij} = c_{ij}$
- Flot complet: Tous les chemins de s à p ont un arc saturé
- Flot maximum: Flot dont la valeur est maximum
- Un flot maximum est complet, mais un flot complet n'est pas forcément maximum:



Algorithme de Ford-Fulkerson: Etape 1

Algorithme de Ford-Fulkerson: Etape 1

- But: obtenir un flot complet

Algorithme de Ford-Fulkerson: Etape 1

- But: obtenir un flot complet
- Commencer avec $f_{ij} = 0 \forall ij \in A$

Algorithme de Ford-Fulkerson: Etape 1

- But: obtenir un flot complet
- Commencer avec $f_{ij} = 0 \forall ij \in A$
- Tant que le flot n'est pas complet:

Algorithme de Ford-Fulkerson: Etape 1

- But: obtenir un flot complet
- Commencer avec $f_{ij} = 0 \forall ij \in A$
- Tant que le flot n'est pas complet:
  - Prendre un chemin non saturé, calculer le min de $c_{ij} - f_{ij}$ sur ce chemin

Algorithme de Ford-Fulkerson: Etape 1

- But: obtenir un flot complet
- Commencer avec $f_{ij} = 0 \forall ij \in A$
- Tant que le flot n'est pas complet:
  - Prendre un chemin non saturé, calculer le min de $c_{ij} - f_{ij}$ sur ce chemin
  - Augmenter le flot de cette valeur sur tout le chemin

Algorithme de Ford-Fulkerson: Etape 2

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet
- Si p est atteint:

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet
- Si p est atteint:
  - Trouver un chemin de sommets marqués, calculer sa capacité disponible: minimum entre les $c_{ij} - f_{ij}$ pour les arcs marqués de i vers j et des f_{ij} pour les arcs marqués de j vers i

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet
- Si p est atteint:
  - Trouver un chemin de sommets marqués, calculer sa capacité disponible: minimum entre les $c_{ij} - f_{ij}$ pour les arcs marqués de i vers j et des f_{ij} pour les arcs marqués de j vers i
  - Ajouter ce flot au chemin en retranchant la valeur pour les arcs marqués en arrière

Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet
- Si p est atteint:
  - Trouver un chemin de sommets marqués, calculer sa capacité disponible: minimum entre les $c_{ij} - f_{ij}$ pour les arcs marqués de i vers j et des f_{ij} pour les arcs marqués de j vers i
  - Ajouter ce flot au chemin en retranchant la valeur pour les arcs marqués en arrière
  - Revenir à l'étape du marquage

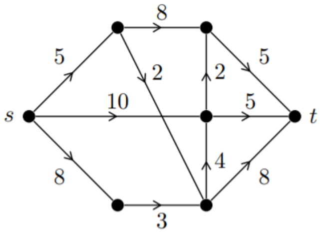
Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet
- Si p est atteint:
  - Trouver un chemin de sommets marqués, calculer sa capacité disponible: minimum entre les $c_{ij} - f_{ij}$ pour les arcs marqués de i vers j et des f_{ij} pour les arcs marqués de j vers i
  - Ajouter ce flot au chemin en retranchant la valeur pour les arcs marqués en arrière
  - Revenir à l'étape du marquage
- Si p n'est pas atteint, le flot est maximum

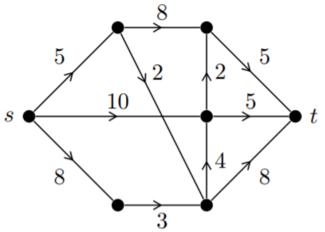
Algorithme de Ford-Fulkerson: Etape 2

- But: améliorer le flot complet jusqu'à ce qu'il soit max
- Marquer les sommets:
  - Marquer s
  - Si ij est un arc non saturé et i est marqué, marquer j
  - Si ij a un flot non nul et j est marqué, marquer i
  - Continuer jusqu'à marquer p ou ne plus pouvoir marquer de sommet
- Si p est atteint:
  - Trouver un chemin de sommets marqués, calculer sa capacité disponible: minimum entre les $c_{ij} - f_{ij}$ pour les arcs marqués de i vers j et des f_{ij} pour les arcs marqués de j vers i
  - Ajouter ce flot au chemin en retranchant la valeur pour les arcs marqués en arrière
  - Revenir à l'étape du marquage
- Si p n'est pas atteint, le flot est maximum
- Les sommets marqués à la fin donnent une coupe minimum

Exemple

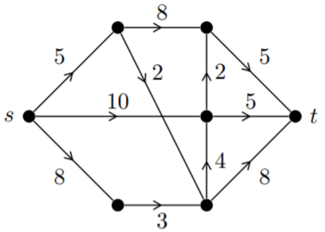


Exemple



- Remarque: Le nombre d'itérations à effectuer dépend de l'ordre du marquage des sommets

Exemple



- Remarque: Le nombre d'itérations à effectuer dépend de l'ordre du marquage des sommets
- Certaines façons de choisir permettent un algorithme polynomial

Couplage Max

Couplage Max

- Entrée: Un graphe biparti de parties A et B

Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes

Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage

Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?

Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?
    - On ajoute s et un arc à capacité 1 de s vers tout A

Couplage Max

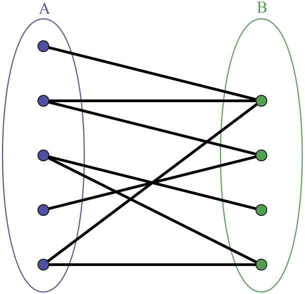
- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?
  - On ajoute s et un arc à capacité 1 de s vers tout A
  - On ajoute t et un arc à capacité 1 de tout B vers t

Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?
  - On ajoute s et un arc à capacité 1 de s vers tout A
  - On ajoute t et un arc à capacité 1 de tout B vers t
  - On oriente toutes les arêtes A vers B, à capacité à 1

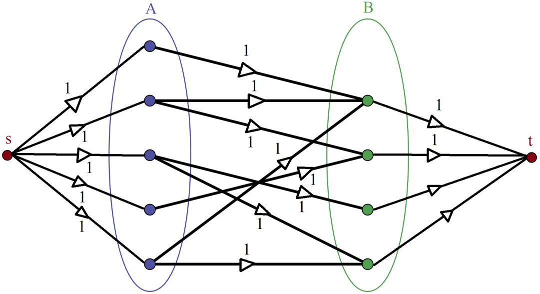
Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?
  - On ajoute s et un arc à capacité 1 de s vers tout A
  - On ajoute t et un arc à capacité 1 de tout B vers t
  - On oriente toutes les arêtes A vers B, à capacité à 1



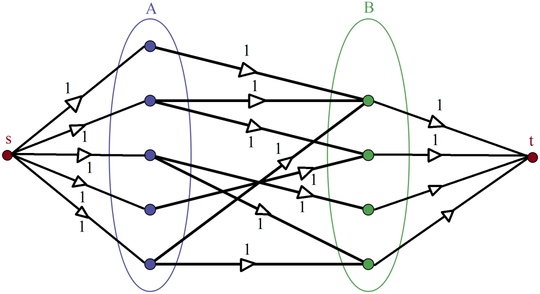
Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?
  - On ajoute s et un arc à capacité 1 de s vers tout A
  - On ajoute t et un arc à capacité 1 de tout B vers t
  - On oriente toutes les arêtes A vers B, à capacité à 1



Couplage Max

- Entrée: Un graphe biparti de parties A et B
- Sortie: Un couplage, i.e. un sous-ensemble d'arêtes disjointes
- Objectif: Maximiser le nombre d'arêtes dans le couplage
- A vous de jouer: Comment transformer un problème de couplage max en un problème de flot max ?
  - On ajoute s et un arc à capacité 1 de s vers tout A
  - On ajoute t et un arc à capacité 1 de tout B vers t
  - On oriente toutes les arêtes A vers B, à capacité à 1



- Un couplage parfait correspond à un flot de valeur $|A|$

Théorème de Hall

Théorème de Hall

- Quelles conditions doit vérifier G pour avoir un couplage parfait ?

Théorème de Hall

- Quelles conditions doit vérifier G pour avoir un couplage parfait ?
  - $|A| = |B|$

Théorème de Hall

- Quelles conditions doit vérifier G pour avoir un couplage parfait ?
  - $|A| = |B|$
  - Si $X \subseteq A$ a strictement moins de $|\mathrm{X}|$ voisins, pas de couplage parfait

Théorème de Hall

- Quelles conditions doit vérifier G pour avoir un couplage parfait ?
  - $|A| = |B|$
  - Si $X \subseteq A$ a strictement moins de $|\mathrm{X}|$ voisins, pas de couplage parfait
- Ce sont en fait des conditions nécessaires et suffisantes:

Théorème de Hall

- Quelles conditions doit vérifier G pour avoir un couplage parfait ?
  - $|A| = |B|$
  - Si $X \subseteq A$ a strictement moins de |X| voisins, pas de couplage parfait
- Ce sont en fait des conditions nécessaires et suffisantes:
- Théorème de Hall (Lemme des mariages): G admet un couplage parfait si et seulement si $|A| = |B|$ et pour tout $X \subseteq A$, $|X| \leq |N(X)|$

Preuve du théorème de Hall

Preuve du théorème de Hall

- Condition nécessaire: trivial

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:
  - Supposons qu'il n'existe pas de couplage parfait. Alors la valeur du flot max est inférieure à $|A|$

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:
  - Supposons qu'il n'existe pas de couplage parfait. Alors la valeur du flot max est inférieure à $|A|$
  - On applique Ford-Fulkerson, on regarde le marquage final

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:
  - Supposons qu'il n'existe pas de couplage parfait. Alors la valeur du flot max est inférieure à $|A|$
  - On applique Ford-Fulkerson, on regarde le marquage final
  - s et au moins un sommet de A sont marqués

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:
  - Supposons qu'il n'existe pas de couplage parfait. Alors la valeur du flot max est inférieure à |A|
  - On applique Ford-Fulkerson, on regarde le marquage final
  - s et au moins un sommet de A sont marqués
  - Soient X_A les sommets de A marqués et X_B les sommets de B marqués. On peut montrer que $N(X_A) = X_B$

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:
  - Supposons qu'il n'existe pas de couplage parfait. Alors la valeur du flot max est inférieure à |A|
  - On applique Ford-Fulkerson, on regarde le marquage final
  - s et au moins un sommet de A sont marqués
  - Soient X_A les sommets de A marqués et X_B les sommets de B marqués. On peut montrer que $N(X_A) = X_B$
  - La valeur de la coupe min est $|A| - |X_A| + |X_B|$

Preuve du théorème de Hall

- Condition nécessaire: trivial
- Condition suffisante:
  - Supposons qu'il n'existe pas de couplage parfait. Alors la valeur du flot max est inférieure à |A|
  - On applique Ford-Fulkerson, on regarde le marquage final
  - s et au moins un sommet de A sont marqués
  - Soient X_A les sommets de A marqués et X_B les sommets de B marqués. On peut montrer que $N(X_A) = X_B$
  - La valeur de la coupe min est $|A| - |X_A| + |X_B|$
  - $|X_B| = |N(X_A)| < |X_A|$, la condition de Hall n'est pas respectée

Bilan des méthodes

Bilan des méthodes

- Trouver un flot max dans un graphe orienté avec des capacités sur les arcs: algorithme de Ford-Fulkerson

Bilan des méthodes

- Trouver un flot max dans un graphe orienté avec des capacités sur les arcs: algorithme de Ford-Fulkerson
- Trouver une coupe min dans un graphe orienté avec des capacités sur les arcs:
  - Si vous connaissez la valeur du flot max, trouver à taton une coupe de la même valeur, vous savez qu'elle est minimum par le théorème de dualité forte

Bilan des méthodes

- Trouver un flot max dans un graphe orienté avec des capacités sur les arcs: algorithme de Ford-Fulkerson
- Trouver une coupe min dans un graphe orienté avec des capacités sur les arcs:
  - Si vous connaissez la valeur du flot max, trouver à taton une coupe de la même valeur, vous savez qu'elle est minimum par le théorème de dualité forte
  - Sinon, appliquez Ford-Fulkerson, une coupe est donnée par l'ensemble des sommets marqués à la fin

Bilan des méthodes

- Trouver un flot max dans un graphe orienté avec des capacités sur les arcs: algorithme de Ford-Fulkerson
- Trouver une coupe min dans un graphe orienté avec des capacités sur les arcs:
  - Si vous connaissez la valeur du flot max, trouver à taton une coupe de la même valeur, vous savez qu'elle est minimum par le théorème de dualité forte
  - Sinon, appliquez Ford-Fulkerson, une coupe est donnée par l'ensemble des sommets marqués à la fin
- Trouver un couplage max dans un graphe biparti: trouvez en un à taton, montrer qu'il est max en trouvant un transversal de même taille et avec le théorème de dualité

Bilan des méthodes

- Trouver un flot max dans un graphe orienté avec des capacités sur les arcs: algorithme de Ford-Fulkerson
- Trouver une coupe min dans un graphe orienté avec des capacités sur les arcs:
  - Si vous connaissez la valeur du flot max, trouver à taton une coupe de la même valeur, vous savez qu'elle est minimum par le théorème de dualité forte
  - Sinon, appliquez Ford-Fulkerson, une coupe est donnée par l'ensemble des sommets marqués à la fin
- Trouver un couplage max dans un graphe biparti: trouvez en un à taton, montrer qu'il est max en trouvant un transversal de même taille et avec le théorème de dualité
- Savoir si il existe un couplage parfait dans un graphe biparti: essayez d'en trouver en un à taton, si vous en trouvez, donnez le, sinon essayez de prouver qu'il n'y en a pas avec le théorème de Hall. Vous pouvez aussi utiliser Ford-Fulkerson sur le graphe orienté obtenu en rajoutant s et t

A vous de jouer !

- Exercices 105, 106, 111, 4 et 5 de la fiche de TD: https://moodle.caseine.org/pluginfile.php/112933/mod_resource/content/1/TD.pdf

A vous de jouer !

- Exercices 105, 106, 111, 4 et 5 de la fiche de TD: https://moodle.caseine.org/pluginfile.php/112933/mod_resource/content/1/TD.pdf
- Correction de l'exercice 105: https://youtu.be/dsvzeNEK5AM

A vous de jouer !

- Exercices 105, 106, 111, 4 et 5 de la fiche de TD: https://moodle.caseine.org/pluginfile.php/112933/mod_resource/content/1/TD.pdf
- Correction de l'exercice 105: https://youtu.be/dsvzeNEK5AM
- Correction de l'exercice 106: https://youtu.be/oy5A6dsBXI4

A vous de jouer !

- Exercices 105, 106, 111, 4 et 5 de la fiche de TD: https://moodle.caseine.org/pluginfile.php/112933/mod_resource/content/1/TD.pdf
- Correction de l'exercice 105: https://youtu.be/dsvzeNEK5AM
- Correction de l'exercice 106: https://youtu.be/oy5A6dsBXI4
- Correction de l'exercice 111: https://youtu.be/zxIRUcYqp4M

A vous de jouer !

- Exercices 105, 106, 111, 4 et 5 de la fiche de TD: https://moodle.caseine.org/pluginfile.php/112933/mod_resource/content/1/TD.pdf
- Correction de l'exercice 105: https://youtu.be/dsvzeNEK5AM
- Correction de l'exercice 106: https://youtu.be/oy5A6dsBXI4
- Correction de l'exercice 111: https://youtu.be/zxIRUcYqp4M
- Correction de l'exercice 4: https://youtu.be/gU7CcPtwZ_M

A vous de jouer !

- Exercices 105, 106, 111, 4 et 5 de la fiche de TD: https://moodle.caseine.org/pluginfile.php/112933/mod_resource/content/1/TD.pdf
- Correction de l'exercice 105: https://youtu.be/dsvzeNEK5AM
- Correction de l'exercice 106: https://youtu.be/oy5A6dsBXI4
- Correction de l'exercice 111: https://youtu.be/zxIRUcYqp4M
- Correction de l'exercice 4: https://youtu.be/gU7CcPtwZ_M
- Correction de l'exercice 5: https://youtu.be/ensUT-uaQfk