Résolution du problème du flot maximum

Auteurs

Cédric Audie
Marie Dalenc
Julien Lahoz
Adrien Martinelli

Encadrant

Rodolphe Giroudeau

30 janvier 2025

Table des matières

1 Définitions générales **2**
1.1 Réseau résiduel 2
1.2 Chemin et flot améliorants 2

2 Différents algorithmes et exemple **2**
2.1 L'algorithme de Ford-Fulkerson 2
2.1.1 Fonctionnement général 2
2.1.2 Algorithme 2
2.1.3 Complexité 3
2.1.4 Avantages/inconvénients 3
2.1.5 Application sur l'exemple 3
2.2 L'algorithme d'Edmonds-Karp 4
2.2.1 Fonctionnement général 4
2.2.2 Algorithme 4
2.2.3 Complexité 4
2.2.4 Avantages/inconvénients 4
2.2.5 Application sur l'exemple 4
2.3 L'algorithme de Dinic 5
2.3.1 Fonctionnement général 5
2.3.2 Algorithme 5
2.3.3 Complexité 5
2.3.4 Avantages/inconvénients 5
2.3.5 Application sur l'exemple 5

1 Définitions générales

1.1 Réseau résiduel

Soit G un graphe avec s, p et c respectivement la source, le puits et la fonction de capacité associés au graphe G. Notons $N = (G, s, p, c)$ un réseau de transport dans G avec un flot f.
Le réseau résiduel de N et d'un flot f, noté N_f, est construit de la manière suivante :
Pour chaque arc xy de G :
- Si $f(xy) < c(xy)$, on crée un arc xy dans G' avec la quantité restante disponible de la capacité : $c'(xy) = c(xy) - f(xy)$.
- Si $f(xy) > 0$ avec $x \neq s$ et $y \neq p$, on crée un arc yx dans G' avec la capacité qu'on peut ré-aiguiller : $c'(yx) = f(xy)$.

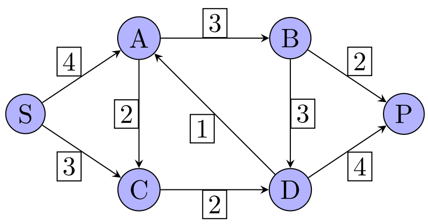
Ceci nous donne $N_f = (G', s, p, c')$.

1.2 Chemin et flot améliorants

- Un chemin améliorant pour N et f est un chemin de s à p dans la réseau résiduel N_f.
- Soit $C = x_0, x_1, ..., x_k$ un chemin améliorant dans N_f, le flot améliorant correspondant est : $f'(xy) = \begin{cases} \gamma & \text{si } xy \text{ est un arc de C} \\ 0 & \text{sinon.} \end{cases}$ avec $\gamma = \min\{c'(x_i x_{i+1}); i \in [\![0; k-1]\!]\}$

2 Différents algorithmes et exemple

Nous prendrons l'exemple suivant pour illustrer les différents algorithmes :



2.1 L'algorithme de Ford-Fulkerson

2.1.1 Fonctionnement général

L'algorithme de Ford-Fulkerson est le plus connu d'entre tous. Celui-ci consiste à trouver un chemin améliorant entre la source et le puits afin d'augmenter la valeur du flot. Pour cela nous aurons besoin du graphe résiduel de G et f qui nous permettra de savoir les capacités restantes disponibles.

2.1.2 Algorithme

Pour tout arc xy du graphe G : on initialise la valeur du flot de xy à 0.
Tant qu'il existe un chemin améliorant C dans le graphe résiduel de G et f :
- Calculer le flot améliorant f' correspondant.
- Augmenter f par f'

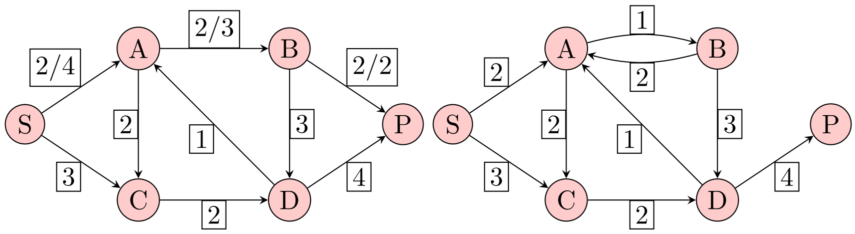
Retourner f.

2.1.3 Complexité

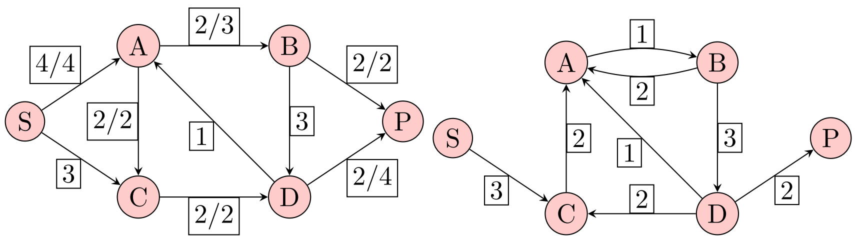
2.1.4 Avantages/inconvénients

2.1.5 Application sur l'exemple

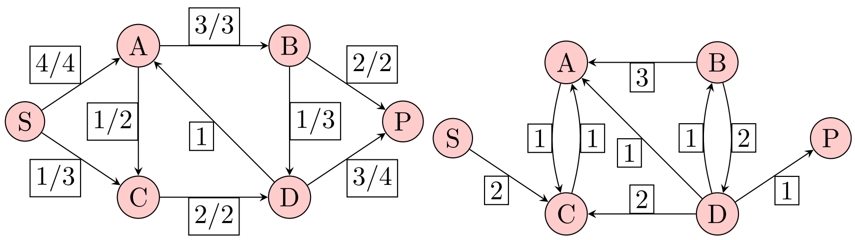
Tout d'abord, on a le chemin améliorant : $SABP$ avec une valeur de flot améliorant égal à 2. On obtient donc à droite le réseau résiduel N_f :



On voit qu'il y a le chemin améliorant $SACDP$ avec un flot améliorant égal à 2.
On obtient à droite le nouveau réseau résiduel N_f :



On voit qu'il y a le chemin améliorant $SCABDP$ avec un flot améliorant égal à 1.
On obtient à droite le nouveau réseau résiduel N_f :



Dans ce dernier réseau résiduel, si nous partons de la source S nous ne pouvons atteindre que les sommets A et C. Comme nous ne pouvons plus atteindre le puits P, cela signifie que le flot obtenu est maximal.
Finalement la coupe minimale de ce réseau est l'ensemble $\{S, A, C\}$ et son flot maximal est de valeur 5.

2.2 L'algorithme d'Edmonds-Karp

2.2.1 Fonctionnement général

L'algorithme d'Edmonds-Karp est une spécificité de celui de Ford-Fulkerson. Celui-ci consiste à trouver le plus court chemin améliorant entre la source et le puits afin d'augmenter la valeur du flot. Pour le trouver, il suffira d'utiliser un parcours en largeur.

2.2.2 Algorithme

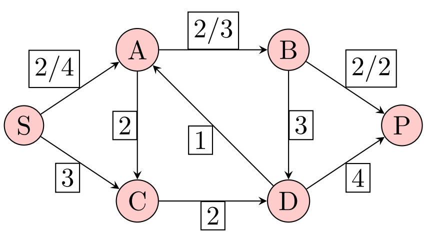
Pour tout arc xy du graphe G : on initialise la valeur du flot de xy à 0.
Tant qu'il existe un plus court chemin améliorant C dans le graphe résiduel de G et f :
— Calculer le flot améliorant f' correspondant.
— Augmenter f par f'
Retourner f.

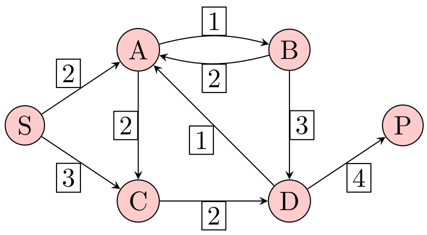
2.2.3 Complexité

2.2.4 Avantages/inconvénients

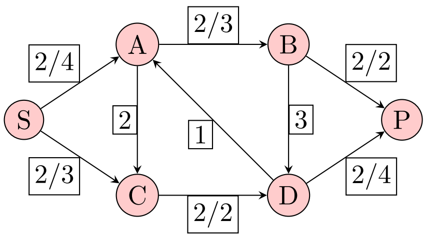
2.2.5 Application sur l'exemple

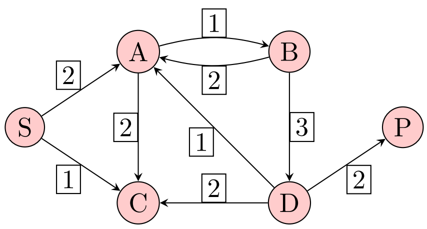
Tout d'abord, on a le chemin améliorant : $SABP$ de taille 4 avec une valeur de flot améliorant égal à 2.
On obtient donc à droite le réseau résiduel N_f :



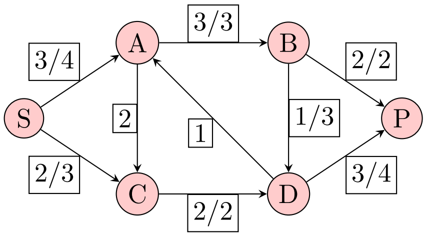


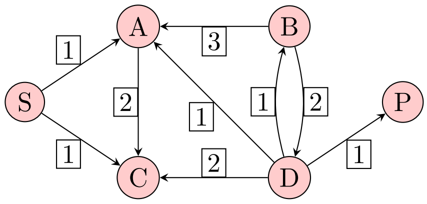
On voit qu'il y a le chemin améliorant $SACDP$ de taille 5 avec un flot améliorant égal à 2. Regardons si on trouve un chemin améliorant plus petit, le chemin $SCDP$ est un autre chemin améliorant de taille 4 avec un flot améliorant égal à 2. C'est ce chemin que nous choisissons.
On obtient à droite le nouveau réseau résiduel N_f :





Il n'y a plus de chemin améliorant de taille 4 allant de S à P. On va choisir un chemin améliorant de taille 5 : $SABDP$ avec un flot améliorant égal à 1.
Nous obtenons à droite le nouveau réseau résiduel N_f :





Dans ce dernier réseau résiduel, si nous partons de la source S nous ne pouvons atteindre que les sommets A et C. Comme nous ne pouvons plus atteindre le puits P, cela signifie que le flot obtenu est maximal.
Finalement la coupe minimale de ce réseau est l'ensemble $\{S, A, C\}$ et son flot maximal est de valeur 5.

2.3 L'algorithme de Dinic

2.3.1 Fonctionnement général

L'algorithme de Dinic est semblable à celui d'Edmonds-Karp. Comme lui, il utilise des plus courts chemins améliorants entre la source et le puits afin d'augmenter la valeur du flot.
Pour les trouver, il faudra renommer les sommets en fonction de leur distance par rapport à la source et garder les arcs qui relient un sommet à un sommet de distance immédiatement supérieure. Ceci nous donnera le réseau de niveau N_L obtenu à partir du réseau résiduel N_f.
Nous aurons également besoin du flot bloquant. Il est défini comme suit : un flot est bloquant si $\forall C$ chemin entre la source et le puits, $\exists xy$ un arc dans C où $f(xy) = c(xy)$.

2.3.2 Algorithme

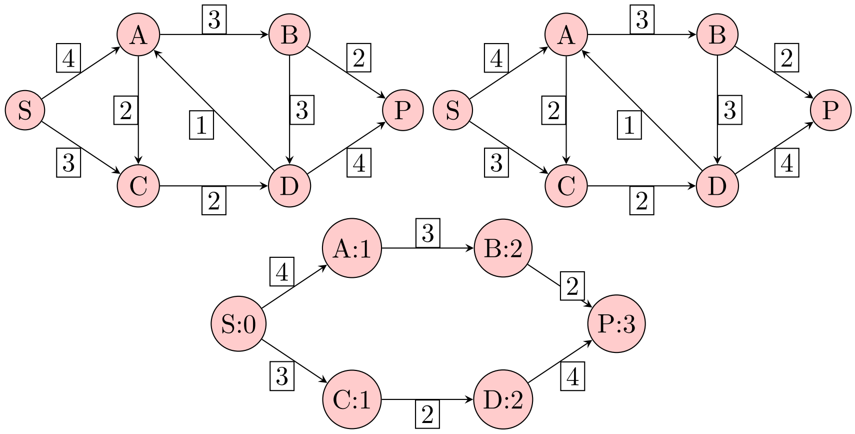
Pour tout arc xy du graphe G : on initialise la valeur du flot de xy à 0.
Tant qu'il existe un plus court chemin améliorant C dans le graphe résiduel de G et f :
— On détermine le réseau de niveau N_L de N_f.
— Calculer le flot bloquant f' correspondant.
— Augmenter f par f'
Retourner f.

2.3.3 Complexité

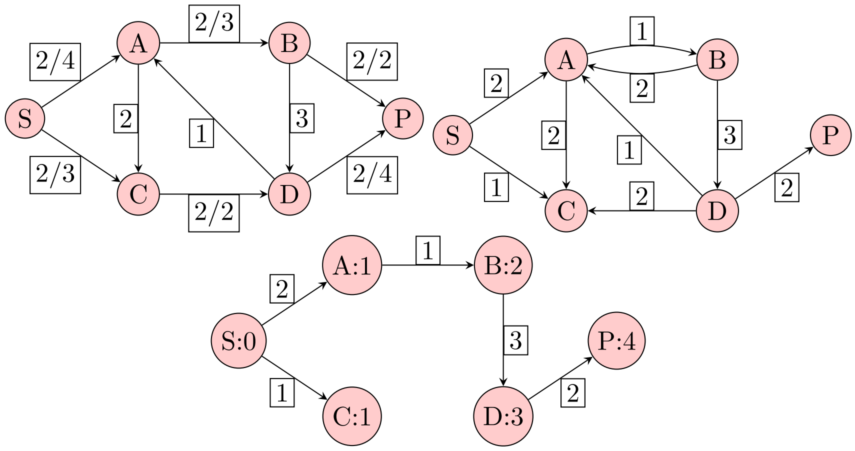
2.3.4 Avantages/inconvénients

2.3.5 Application sur l'exemple

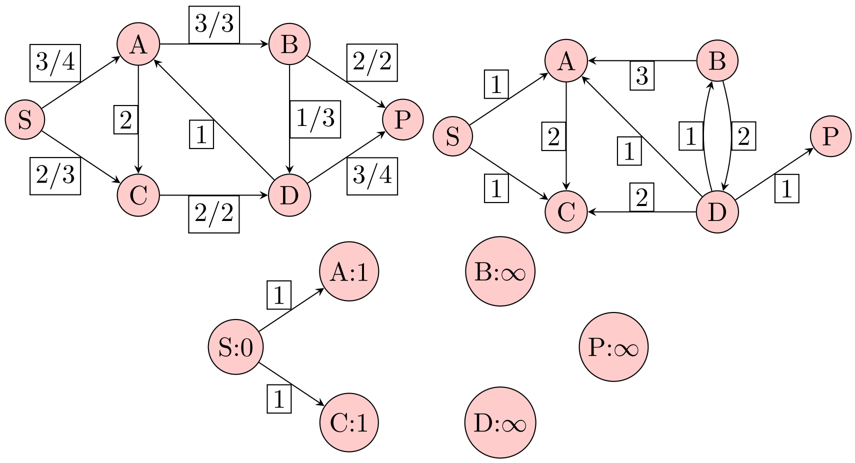
Au début le flot est nul, le réseau résiduel N_f à droite est donc le même que le réseau de transport N.
Nous avons aussi en dessous, le réseau de niveau N_L :



On a donc un chemin améliorant $SABP$ de valeur 2 ainsi qu'un chemin améliorant $SCDP$ de valeur 2.



Nous avons un chemin améliorant $SABDP$ de valeur 1.



Le puits P n'est plus atteignable, l'algorithme se termine.
Finalement la coupe minimale de ce réseau est l'ensemble $\{S, A, C\}$ et son flot maximal est de

valeur 5.