

1 Algorithmes de préflots

1.1 Introduction

Les algorithmes de flot maximum parmi les plus rapides connues à ce jour sont des algorithmes de préflots. Ces algorithmes travaillent de façon plus localisée que la méthode de Ford-Fulkerson. Ce tableau présente les deux principes :

Ford-Fulkerson	Les algorithmes de préflots
Recherche un chemin améliorant en examinant le réseau résiduel tout entier.	Ils se concentrent sur un sommet à la fois en ne s'intéressant qu'à ses voisins dans le réseau résiduel.

Ces algorithmes de préflot maintiennent un préflot défini par une fonction :

$$f : S \times S \longrightarrow R$$

qui vérifie la propriété de symétrie et la conservation de flot tel que :

$$f(S, u) \geq 0 \quad \forall u \in S - \{s\},$$

on appelle cette quantité l'excédent de flot sur u , donné par :

$$e(u) = f(S, u).$$

Propriété :

On dit qu'un sommet $u \in S - \{s, t\}$, déborde si $e(u) \geq 0$.

1.2 Principe

Propriétés :

L'algorithme de préflots générique possède deux propriétés intéressantes :

1. Pour évacuer le flot excédentaire, chaque sommet est doté d'un tuyau de vidange, menant à un réservoir pouvant accumuler le liquide.
2. Chaque sommet, son réservoir et tous ses raccordements se trouvent sur une plate-forme dont la hauteur croît à mesure que l'algorithme progresse.

Alors les hauteurs des sommets déterminent la façon dont le flot est poussé : on ne pousse que vers le bas, ie, d'un sommet supérieur vers un sommet inférieur. La hauteur de la source est fixée à $|S|$ et celle du puits est fixée à 0. Tous les autres sommets commencent à la hauteur 0 et montent avec le temps.

L'algorithme commence par faire descendre le plus de flot possible depuis la source vers le puits.

La quantité envoyée est juste assez importante pour remplir au maximum chaque tuyau partant de la source.

Quand le flot arrive sur un sommet intermédiaire, il est récupéré dans le réservoir du sommet, après, il sera ensuite poussé vers le bas.

Et pour débarasser un sommet u débordant de son flot excédentaire, il faut augmenter sa hauteur, opération appelée "ré-étiquetage" du sommet u , sa hauteur est augmentée d'une unité de plus que la hauteur du plus bas de ses voisins vers lequel il possède un tuyau non saturé.

L'algorithme renvoie ensuite à la source l'excédent contenu dans les réservoirs des sommets débordants, en continuant à ré-étiqueter les sommets pour qu'ils dépassent la hauteur $|S|$ fixée de la source.

Une fois que les réservoirs ont été vidés, le préflot est un flot maximum.

Cette discussion montre qu'il existe deux opérations principales effectuées par un algorithme de préflot : le poussage du flot excédentaire d'un sommet vers ses voisins et le ré-étiquetage d'un sommet.

Définition :

Soit $G = (S, A)$ un réseau de transport de source s et de puits t et soit f un préflot de G .

Une fonction $h : S \rightarrow N$ est une fonction de hauteur si

$$h(s) = |S|, h(t) = 0 \text{ et } h(u) \leq h(v) + 1 \quad \forall (u, v) \in A_f.$$

Définition :

L'expression $d_f(u, v)$ est une variable temporaire qui stocke la quantité de flot pouvant être poussée de u vers v .

L'opération de poussage peut s'appliquer si u est un sommet débordant et que la capacité résiduelle de (u, v) est positive, ainsi qu'on interdise son exécution tant qu'on a pas $h[u] = h[v] + 1$, donc, le flot excédentaire n'est poussé vers le bas que si la différence de hauteur est égale à 1.

Le pseudo code suivant présente cette opération :

Pousser(u, v) :

si : $e[u]$ positif, $c_f[u, v] > 0$, $h[u] = h[v] + 1$.

faire : pousser $d_f(u, v) = \min(e[u], c_f[u, v])$.

$d_f(u, v) \leftarrow \min(e[u], c_f[u, v])$. // calcule de la valeur de $d_f(u, v)$

$f[u, v] \leftarrow f[u, v] + d_f(u, v)$ // actualiser f

$f[v, u] \leftarrow -f[u, v]$

$e[u] \leftarrow e[u] - d_f(u, v)$ // actualiser e

$e[v] \leftarrow e[v] + d_f(u, v)$

L'opération de ré-étiquetage s'applique si u est un sommet débordant et si $h[u] \leq h[v] \quad \forall (u, v) \in A_f$.

cette opération permet d'accroître la hauteur de u , elle par suite à u la plus grande hauteur permise par les contraintes de la fonction de hauteur.

Le pseudo code suivant présente cette opération :

Ré-étiqueter(u, v) :

si : $e[u]$ positif et $h[u] \leq h[v] \quad \forall (u, v) \in A_f$.

faire : accroît la hauteur de u :

$h[u] \leftarrow 1 + \min\{h[v] \quad \forall (u, v) \in A_f\}$.

1.3 Algorithme générique

L'algorithme générique fait appel à un sous programme qui permet de créer un préflot initial dans le réseau. la procédure suivante permet de créer un préflot initial :

$$f\{u, v\} = \begin{cases} c(u, v) & \text{si } u = s, \\ -c(u, v) & \text{si } v = s, \\ 0 & \text{sinon.} \end{cases}$$

L'algorithme générique démarre avec la fonction de hauteur définit par:

$$h[u] = \begin{cases} |S| & \text{si } u = s, \\ 0 & \text{sinon.} \end{cases}$$

Alors l'initialement, suivie d'une séquence de poussage et ré-étiquetage exécutés dans un ordre quelconque, donne l'algorithme générique du préflot:

Préflot-Générique(G)

1. Initialiser-Freflot(G,s)

2. **tant que il** est possible d'appliquer un poussage ou un ré-étiquetage

Faire choisir un poussage ou ré-étiquetage applicable et l'exécuter.