

Ford-Fulkerson Method

Jake Underland (1A193008-2)

1/30/2022

目次

1	問題設定	1
2	重要概念	1
2.1	残余グラフ	1
2.2	増加道	2
2.3	最小カット	2
3	フォードファルカーソン法	3
4	References	4

概要

フォード・ファルカーソナルゴリズムとは、最大流問題と、その双対問題である最小カット問題を解くためのアルゴリズムである。以下では、そのアルゴリズムについて解説していく。

1 問題設定

- 頂点と重み付きの辺を持つグラフ $G(V, E)$ がある。また、グラフの頂点には始点 s と終点 t がある。
- ある隣接する 2 頂点 u, v 間の辺が許容できる容量を $c(u, v)$ と表す。 $c(u, v)$ は非負の整数である。
- u から v まで流すフローを $f(u, v)$ と表す。 また、 $f(s, t)$ を省略して f と記す。
- s と t を除く全ての頂点で、出ていくフローと入っていくフローが等しい。
- 一度にどこまでの“フロー”を、このグラフは s から t まで流すことができるかを考える。

2 重要概念

2.1 残余グラフ

グラフ G とフロー f が与えられる時、その残余グラフ G_f は頂点間の辺を流れるフローをどう変えることができるか表すグラフであると言える。 G の辺 u, v は、その間で流すことが可能な容量 $c(u, v)$ から、その間に実際流れているフロー $f(u, v)$ を引いた値がその頂点間でまだ流せる量であり、残余グラフにおいては、これが正であるならば新たな辺としてグラフに書き表す。 また、2 頂点間に流れているフローを減らしたい場合も考えられるので、その時点で流れているフローを $f(u, v)$ とすると、残余グラフにおいては v から u 方向に伸びて容量が $f(u, v)$ である辺も追加する。 図 1 は、グラフ G と対応する残余グラフ G_f の例である。 残余グラフもグラフなので、問題設定で定義した概念が存在する。 ここでは、それらが元グラフではなく残余グラフのものを指すことを示すため、サブスクリプト f を使って表す（例えば残余グラフ上の (u, v) 間の経路が許容できるフローを $c_f(u, v)$ とする）。

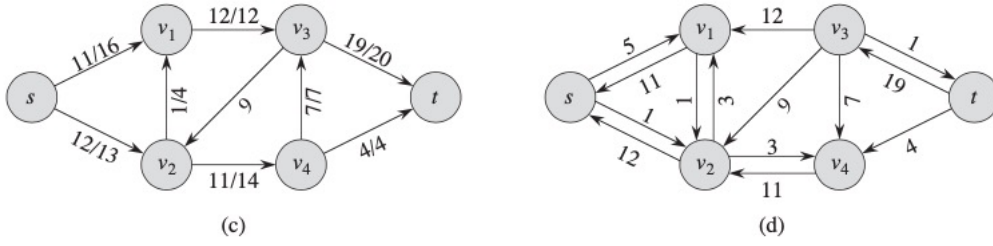


図 1: From Cormen et al. (2009)

2.2 増加道

グラフ $G(V, E)$ とフロー f が与えられる時, G_f 増加道 p とは s から t を結ぶ経路 (s - t パス) である. 残余グラフの定義より, ある経路上のフローは, その経路上の頂点 (u, v) の容量 $c_f(u, v)$ を超えない限り, 元グラフにおいてその分だけその経路間のフローを増やすことができる. 図 2 では, 残余グラフ上に増加道となる p を示している. 残余グラフ上では, 増加道 p 上の 2 頂点間で許容できる容量が最小なのは $c_f(v_2, v_3)$ なので, 増加道にフローを 4 流し, それだけ下のグラフの合計フロー f を増やすことができる.

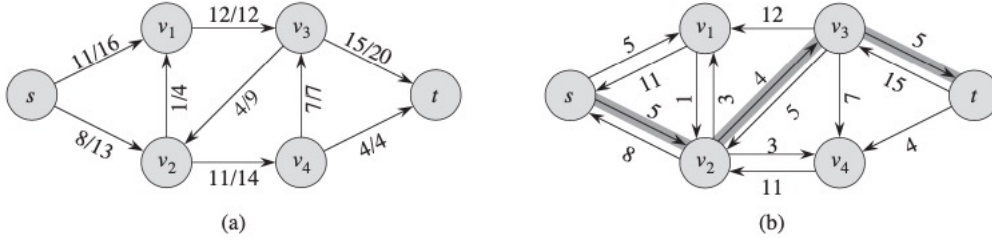


図 2: From Cormen et al. (2009)

2.3 最小カット

カット (S, T) とは $G(V, E)$ の頂点集合 V を $S, T = V - S$ への分割で, $s \in S, t \in T$ となるものを指す. カット (S, T) の容量は

$$c(S, T) = \sum_{u \in S} \sum_{v \in T} c(u, v)$$

と定義され, 上の容量が最小になるものを最小カットという. カットの容量は, S から T の容量だけを数え, 逆方向に流れる容量は数えないものとする. また, $u \in S, v \in T$ だが互いに隣接していない頂点は (u, v) が定義されないため容量の計算には含まれない.

次いで, カットの純フローを定義する.

$$f(S, T) = \sum_{u \in S} \sum_{v \in T} f(u, v) - \sum_{u \in S} \sum_{v \in T} f(v, u)$$

この時, どこでカットをとってもその純フローは等しく, そのグラフの総フローの値 $|f|$ である.

ここで大事なのはカットの純フローの上界はカットの容量であることだ. これは, 下記から明瞭である.

$$|f| = \sum_{u \in S} \sum_{v \in T} f(u, v) - \sum_{u \in S} \sum_{v \in T} f(v, u) \leq \sum_{u \in S} \sum_{v \in T} f(u, v) \leq \sum_{u \in S} \sum_{v \in T} c(u, v) = c(S, T)$$

よって, あるグラフのフローが一つの頂点で堰き止められずに終点に至るには, 最小カットの容量以下でなければいけない. そして, 最小カット (S, T) において T から S のフローを 0 にすることで, 最小カットの容量と同値の容量を流すことができます (最大流最小カット定理). ここで, 最小カットで S から T へのフローを最大限に使うということは, すなわち S から T にフローを流せるパスがなくなることと等しく, 残余グラフ上の s - t パスがなくなります. ここに, フォードファルカーソン法の終止条件があります.

3 フォードファルカーソン法

以上をまとめると、最大流問題における最大フローは、以下の手順で求められます。

1. フロー f を 0 に初期化
2. **while** 残余グラフ上に増加道 (s - t パス) p が存在する限り：
 フロー f を p にしたがって増加する
3. **return** f

以下では、図を用いて実際にフォードファルカーソン法により最大流問題を解く過程を示す。

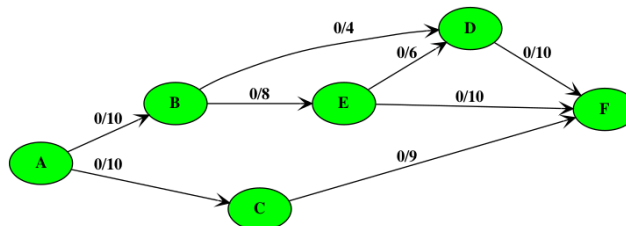
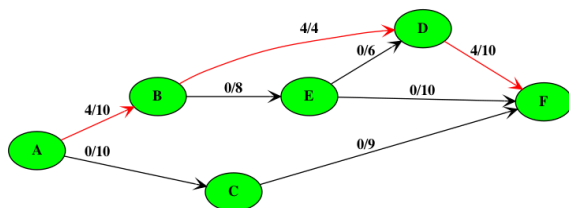
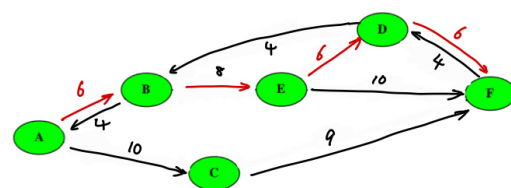


図 3: 上のグラフの最大流問題を解いていく。

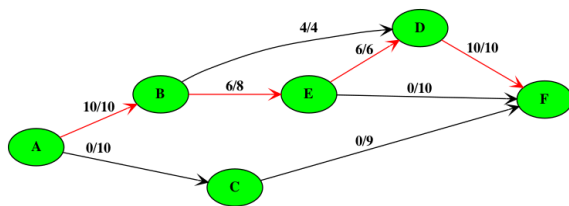


(a) 最初のフロー

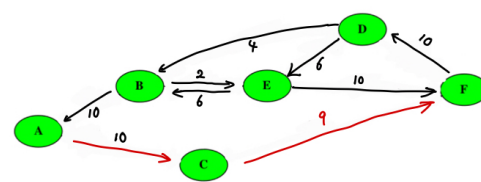


(b) 残余グラフと増加道

図 4: (a) では、最初に一つの s - t パスを定め、そのパス上に存在する辺のうち、容量が最小である $c(B, D)$ 分をフローとして流す。この時の残余グラフが (b) であり、次にとるべき増加道 p と p 上の辺の最小容量 $c(A, B) = c(E, D) = c(D, F) = 6$ を赤色で示している。



(a) 更新されたフロー



(b) 新しい残余グラフと増加道

図 5: (a) では、図 4 で提示された増加道に従って元グラフのフローを更新している。(b) はそれに対応した新たな残余グラフであり、残余グラフ上の次の増加道を示している。また、これが唯一の増加道であることに注意してほしい。

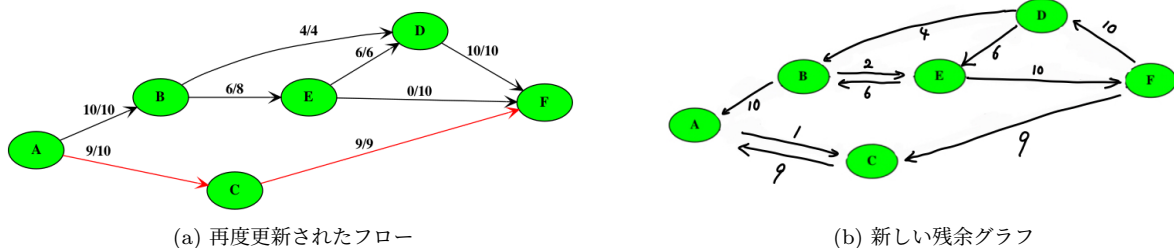


図 6: (a) では、図 5 で提示された増加道に従って元グラフのフローを更新している. (b) はそれに対応した新たな残余グラフであり、残余グラフ上にはもう s - t パスが存在しないことが観察できる. よって、フォードファルカーソン m 方の終止条件を満たし、最大フローが達成された.

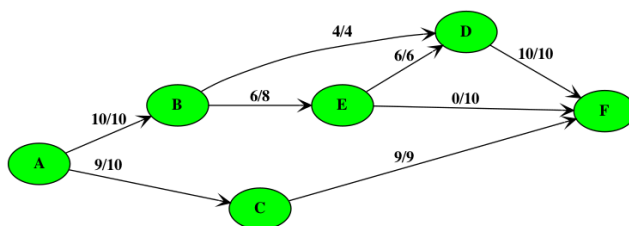


図 7: 最大流問題の最終的なフロー

こうして、最大流問題を解いた結果、最大フローは 19 であることがうかがえる.

以上が、フォードファルカーソン法である. この方法では、残余グラフに現れる s - t パスごとにフローは少なくとも 1 増えるので、最大化されたフローを $|f^*|$ とするとグラフのフローを更新する回数は最大で $|f^*|$ 回である. また、残余グラフ上で s - t パスを見つける操作は $O(E)$ であるため、フォードファルカーソン法の計算量は $O(|f^*|E)$ である.

4 References

Cormen, T.H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to algorithms (3rd ed.). MIT Press.