1　未定義の用語

1.1　有向グラフ (Digraph)

定義 1.1.1 (有向グラフ (Digraph)). V:空でない有限な集合であり, $E \subseteq V^2 \backslash \{(x,x) | x \in V\}$ であるとき, $D = (V,E)$ を有向グラフ (Digraph) という.　また, 有向グラフ $D = (V,E)$ に対して, V を頂点集合, E を有向辺集合と呼び, 各要素を頂点 (vertices), 有向辺 (arcs,directed edges) という.

　だいたいは $E \neq \emptyset$ の場合を考える. また, 図で表す際には, $(u,v) \in E$ を u から v への矢印で表すことが多い. 定義より, 辺 (u,v) と (v,u) は区別され, これらは向きを持っていると考えることが出来る. また, (u,v) と (v,u) のように向きが逆な辺であれば、2 頂点間に辺が 2 本接続することが可能である.

定義 1.1.2 (隣接 (adjacent)). グラフ D の有向辺 (u,v) に対して, u は v へ隣接している (u is adjacent to v) といい, 逆に v は u から隣接している (v is adjacent from u) という.

定義 1.1.3 (近傍 (out-neighborhood,in-neighborhood)). グラフ D の頂点 v に対して,

$$N^+(v) = \{x \in V | (v,x) \in E\}$$

$$N^-(v) = \{x \in V | (x,v) \in E\}$$

をそれぞれグラフ D における v の外近傍 (out-neighborhood), 内近傍 (in-neighborhood) という.

定義 1.1.4 (次数 (outdegree,indegree)). グラフ D の頂点 v に対して,

$$od(v) = |\{(v,x) \in E | ^\exists x \in V\}|$$

$$id(v) = |\{(x,v) \in E | ^\exists x \in V\}|$$

をそれぞれグラフ D における v の外次数 (outdegree), 内次数 (indegree) という. また, v の次数 $d(v)$ を $d(v) = od(v) + id(v)$ と定める.

　つまり, 次数とは v に接続している辺の本数である. また, 明らかに $od(v) = |N^+|, id(v) = |N^-|$ である.

1.2 ネットワーク (Network)

定義 1.2.1 (ネットワーク (Network))**.** 有向グラフ $D = (V, E)$ が, source と sink という 2 つの異なる頂点 u, v をもち, また $c : E \to \mathbb{R}^+ (\mathbb{R}^+ = \{|x| | x \in \mathbb{R}\})$ が存在するとき, $N = (D, v, u, c)$ をネットワーク (Network) という.

D を N の underlying digraph , c を N の容量関数 (capacity function) , $e = (x, y) \in E$ に対する $c(e) = c(x, y)$ の値を e の capacity , v, u 以外の $N(D)$ の頂点を N の intermediate vertex という.

表記 1.2.2. 有向グラフ $D, g : E(D) \to \mathbb{R}, X, Y \subset V(D)$ に対して,

$$[X, Y] = \{(x, y) | x \in X, y \in Y\}$$

$$g(X, Y) = \sum_{(x,y)\in[X,Y]} g(x, y) \ ([X, Y] = \emptyset \Rightarrow g(X, Y) = 0)$$

また, $x \in V(D)$ のとき,

$$g^+(x) = \sum_{y \in N^+(x)} g(x, y) \ , \ g^-(x) = \sum_{y \in N^-(x)} g(y, x)$$

とし, より一般に $X \subseteq V(D)$ のとき,

$$g^+(X) = \sum_{x \in X} g^+(x) \ , \ g^-(X) = \sum_{x \in X} g^-(x)$$

1.3 フローネットワーク (Flow Network)

定義 1.3.1 (Flow)**.** ネットワーク $N = (D, u, v, c)$ に対して, $f : E(D) \to \mathbb{R}$ が以下を満たしているとき, f は N のフロー (flow) であるという.

(i) ${}^{\forall} a \in E(D), 0 \leq f(a) \leq c(a)$

(ii) $\forall x \in V(D) \backslash \{u, v\}, f^+(x) = f^-(x)$

$a = (x, y) \in E(D)$ のとき, $f(a) = f(x, y)$ を a に沿ったフローという. また, (2) をフローの保存則 (conservation equation) という.

他にも, $f : E(D) \to 0$ の場合, f はフローになる. これをゼロフローという. $X \subset V(D)$ に対して, $f^+(X) - f^-(X)$ を X から出ていくネットフロー, $f^-(X) - f^+(X)$ を X に入っていくネットフローという. とくに $x \in V(D)$ に対して, $f^+(x) - f^-(x)$

を x から出ていくネットフロー, $f^-(x) - f^+(x)$ を x に入っていくネットフローという. x が intermediate vertex であるとき, これらは (2) より 0 となる.

$a \in E(D)$ に対して $f(a) = c(a)$ であるとき, a は f について飽和している (saturated) といい, そうでないときには不飽和 (unsaturated) であるという.

定理 1.3.2. グラフ $N = (D, u, v, c)$, f を N 上のフローとするとき,

$$f^+(u) - f^-(u) = f^-(v) - f^+(v)$$

が成り立つ.

Proof.

$$\sum_{x \in V(D)} f^+(x) = \sum_{x \in V(D)} f^-(x) \ i.e.\ f^+(V(D)) = f^-(V(D))$$

であるから, 定義 1.3.1 の (2) より

$$f^+(u) - f^-(u) = f^-(v) - f^+(v)$$

が導ける. □

1.4 最大フロー (Maximum Flow)

定義 1.4.1 (value)**.** $N = (D, u, v, c)$ において, source u から出ていくネットフローをフロー f の value といい, $\mathrm{val}(f)$ で表す. すなわち $\mathrm{val}(f) = f^+(u) - f^-(u)$ である.

定義 1.4.2 (最大フロー (Maximum Flow))**.** $N = (D, u, v, c)$ に対して, value が最大となるフロー f のことを N の最大フロー (Maximum flow) という. すなわち $^\forall f'$:N 上のフロー, $\mathrm{val}(f) \geq \mathrm{val}(f')$ である.

一意には定まらないが存在する. これは定義 1.3.1 の (1) より従う.

定義 1.4.3 (カット (Cut))**.** $N = (D, u, v, c)$. $X \subset V(D)$ に対して $\overline{X} = V(D) - X$ と定める. $u \in X \wedge v \in \overline{X}$ であるとき, $A = [X, \overline{X}] \subset E(D)$ を N のカット (cut) という.

u から v への path は必ず K を通らなければならない,

補題 1.4.4. $N = (D, u, v, c)$: ネットワーク, f: フロー, $X \subset V(D)$,

$$f^+(X) - f^-(X) = f(X, \overline{X}) - f(\overline{X}, X)$$

Proof.

$$
\begin{aligned}
f^+(X) - f^-(X) &= \sum_{x\in X} f^+(x) - \sum_{x\in X} f^-(x) \\
&= \sum_{x\in X}\left(\sum_{y\in N^+(x)} f(x,y)\right) - \sum_{x\in X}\left(\sum_{y\in N^-(x)} f(y,x)\right) \\
&= \sum_{x\in X}\left(\sum_{y\in N^+(x)\cap X} f(x,y)\right) + \sum_{x\in X}\left(\sum_{y\in N^+(x)\cap \overline{X}} f(x,y)\right) \\
&\quad - \sum_{x\in X}\left(\sum_{y\in N^-(x)\cap X} f(y,x)\right) + \sum_{x\in X}\left(\sum_{y\in N^-(x)\cap \overline{X}} f(y,x)\right) \\
&= \sum_{a\in[X,X]} f(a) + \sum_{a\in[X,\overline{X}]} f(a) - \sum_{a\in[X,X]} f(a) - \sum_{a\in[\overline{X},X]} f(a) \\
&\quad \Big((\because)[A,B] = \{(x,y)\in E(D) | x\in A, y\in B\} \\
&\qquad = \{(x,y)|x\in A, y\in B\cap N^+(x)\}(\{(x,y)|x\in A\cap N^-(x), y\in B\})\Big) \\
&= \sum_{a\in[X,\overline{X}]} f(a) - \sum_{a\in[\overline{X},X]} f(a) \\
&= f(X,\overline{X}) - f(\overline{X},X)
\end{aligned}
$$

□

定義 1.4.5 (容量 (Capacity))**.** $N = (D,u,v,c)$. $K = [X,\overline{X}]$ を N のカットとする. このとき, カットに含まれる arc の容量の合計値をカット K の容量 (capacity) といい,$\mathrm{cap}(K)$ と表す. すなわち

$$\mathrm{cap}(K) = c(X,\overline{X}) = \sum_{(x,y)\in[X,\overline{X}]} c(x,y)$$

である.

定理 1.4.6. $N = (D,u,v,c)$: ネットワーク, f:N のフロー, $K = [X,\overline{X}]$:N のカット, $\mathrm{val}(f) = f^+(X) - f^-(X) \le \mathrm{cap}(K)$

Proof. 仮定より $v \notin X$ であり, 定義1.3.1 の (2) より, $\forall x \in X - \{u\}, f^+(x) - f^-(x) =$

0 である. よって

$$\begin{aligned} f^+(X) - f^-(X) &= \sum_{x\in X} f^+(X) - \sum_{x\in X} f^-(X) \\ &= \sum_{x\in X}(f^+(x) - f^-(x)) \\ &= (f^+(u) - f^-(u)) \\ &= \mathrm{val}(f) \end{aligned}$$

が成り立つ. また, ${}^\forall a \in E(D), 0 \le f(a) \le c(a)$ であるから, 補題 1.4.4 より,

$$\begin{aligned} f^+(X) - f^-(X) &= f(X, \overline{X}) - f(\overline{X}, X) \\ &\le f(X, \overline{X}) \\ &\le c(X, \overline{X}) \\ &= \mathrm{cap}(K) \end{aligned}$$

となり, 示せた. □

1.5 最小カット (Minimum Cut)

定義 1.5.1 (最小カット (Minimum Cut))**.** $N = (D, u, v, c)$ に対して, capacity が最小となるカット K のことを N の最小カット (Minimum cut) という. すなわち ${}^\forall K'$:N 上のカット, $\mathrm{cap}(K) \ge \mathrm{cap}(K')$ である.

一意には定まらないが, 存在する. これはネットワークの定義より従う.

系 1.5.2. *$N = (D, u, v, c)$: ネットワーク, f:N のフロー, K:N のカット. このとき, $\mathrm{val}(f) = \mathrm{cap}(K)$ ならば, f は N の最大フローであり, K は N の最小カットである.*

Proof. $\mathrm{val}(f) = \mathrm{cap}(K)$ とすると, 任意の N のフロー f', 任意の N のカット K' に対して, 定理 1.4.6 より, $\mathrm{val}(f') \le \mathrm{cap}(K) = \mathrm{val}(f) \le \mathrm{cap}(K')$ である. よって, f は N の最大フローであり, K は N の最小カットである. □

系 1.5.3. *$N = (D, u, v, c)$: ネットワーク, f:N のフロー, $K = [X, \overline{X}]$:N のカット. このとき,*

$$({}^\forall a \in [X, \overline{X}], f(a) = c(a)) \wedge ({}^\forall a \in [\overline{X}, X], f(a) = 0)$$

ならば, f は N の最大フローであり, K は N の最小カットである.

Proof. 定理 1.4.6 より,

$$\begin{aligned}\mathrm{val}(f) &= f^+(X) - f^-(X)\\ &= f(X,\overline{X}) - f(\overline{X},X)\\ &= c(X,\overline{X}) - 0\\ &= \mathrm{cap}(K)\end{aligned}$$

よって系 1.5.2 より f は N の最大フローであり, K は N の最小カットである. □

最大フローと最小カットであるための条件を与えている. 特に系 1.5.2 は定理 2.1.1 の十分条件を与えている. 必要条件を与えるために, ここで一つ便利な概念を導入する.

定義 1.5.4 (semipath)**.** 有向グラフ D に対して semipath とは, 以下を満たすような空でない有向グラフ $P=(V,E)$ のことである.

$V=\{\omega_i|i=0,\cdots,k\}, E=\{a_i\in E(D)|a_i=(\omega_{i-1},\omega_i)\vee a_i=(\omega_i,\omega_{i_1}), i=1,\cdots k\}$(各$\omega_i$は異なる)

またこのとき, P を ω_0 から ω_k への semipath (ω_0-ω_k semipath) という. またこのとき E の元について, $a_i=(\omega_{i-1},\omega_i)$ を forward arc , $a_i=(\omega_i,\omega_{i_1})$ を backward arc という.

表記 1.5.5. 上の semipath を $P=(\omega_0,a_1,\omega_1,\cdots,\omega_{k-1},a_k,\omega_k)$ と書き表す.

1.6 f-Augmenting Semipaths

定義 1.6.1. $N=(D,u,v,c)$:ネットワーク, f:N のフロー, $P=(\omega_0,a_1,\omega_1,\cdots,\omega_{k-1},a_k,\omega_k)$:$D$ の semipath とする. P が以下の条件を満たしているとき, P は f-unsaturated な semipath であるという.

(i) $f(a_i)<c(a_i)$ (a_i:forward arc)

(ii) $f(a_i)>0$ (a_i:backward arc)

自明な semipath ($P=(\omega_0)$) は f-unsaturated とする. P が f-unsaturated な u-v semipath であるとき, f-augmenting な semipath であるという.

定理 1.6.2. $N=(D,u,v,c)$: ネットワークとする. このとき, f が最大フローであることと D 上に f-augmenting な semipath が存在しないことは同値である.

Proof. $\Rightarrow$)f を最大フローとし, D 上に f-augmenting な semipath P が存在するとする. $P = (\omega_0, a_1, \omega_1, \cdots, \omega_{k-1}, a_k, \omega_k)$ とすると, $\omega_0 = u, \omega_k = v$ である. P の forward arc a_{i_n} について, $c(a_{i_n}) - f(a_{i_n}) > 0$ であり $n \leq k < \infty$ であるため, $p_1 = \min\{c(a_{i_n}) - f(a_{i_n}) | a_{i_n}\text{:forward arc}\} > 0$ が存在する. 同様に P の backward arc a_{i_m} についても, $f(a_{i_n}) > 0$ であるため $p_2 = \min\{f(a_{i_m}) | a_{i_n}\text{:backward arc}\} > 0$ が存在する. $p = \min\{p_1, p_2\}$ とすれば,

$$f'(a) = \begin{cases} f(a) + p & \text{if } a \text{ is a forward arc on } P \\ f(a) - p & \text{if } a \text{ is a backward arc on } P \\ f(a) & \text{if } a \notin E(P) \end{cases}$$

は N のフローとなり[1],

(i) $f^+(u) + p = f'^+(u)$ (a_1:forward arc)

(ii) $f^-(u) - p = f'^-(u)$ (a_1:backward arc)

より $f^+(u) - f^-(u) < f'^+(u) - f'^-(u)$ *i.e.* $\mathrm{val}(f) < \mathrm{val}(f')$ であるから f が最大フローであることに矛盾する. よって f-augmenting な semipath P は存在しないことがわかる.

$\Leftarrow$)f が D 上 f-augmenting な semipath が存在しないような flow とする. このとき f が最大フローであることを示す. 今 $X = \{x \in V(D) | {}^{\exists}P\text{:}f\text{-unsaturated } u\text{-}x \text{ semipath}\}$ とすると, $u \in X, v \notin X$ であるため, $K = [X, \overline{X}]$ は cut となる. ${}^{\forall}a \in [X, \overline{X}], {}^{\forall}b \in [\overline{X}, X], f(a) = c(a), f(b) = 0$ であるから[2], 系 1.5.3 より f は最大フローである. また, K は最小カットとなっている. $\square$

---

[1] ω_i $(0 < i < k)$ に対して

(i) $f^+(\omega_i) + p = f'^+(\omega_i), f^-(\omega_i) + p = f'^-(\omega_i)$ (a_i, a_{i+1}:forward arc)

(ii) $f^-(\omega_i) + p - p = f'^-(\omega_i)$ (a_i:forward arc, a_{i+1}:backward arc)

(iii) $f^+(\omega_i) - p + p = f'^+(\omega_i)$ (a_i:backward arc, a_{i+1}:forward arc)

(iv) $f^+(\omega_i) - p = f'^+(\omega_i), f^-(\omega_i) - p = f'^-(\omega_i)$ (a_i, a_{i+1}:backward arc)

であるため $f^+(\omega_i) - f^-(\omega_i) = f'^+(\omega_i) - f'^-(\omega_i)$ であり, $x \in V(D) \backslash V(P)$ については明らかに $f^+(x) - f^-(x) = f'^+(x) - f'^-(x)$ である. また, f' は p のとり方から各 arc の容量を超えない. ゆえに f' は flow である.

[2] 任意の $(y, z) \in [X, \overline{X}]$ について, $y \in X$ より f-unsaturated u-y semipath が存在し, $z \in \overline{X}$ より f-unsaturated u-z semipath が存在しない. $f(y, z) < c(y, z)$ とすると, f-unsaturated u-y semipath に $f(y, z), z$ を加えたものは f-unsaturated u-z semipath になり矛盾する. よって $f(y, z) = c(y, z)$ である. 同様に, 任意の $(w, x) \in [\overline{X}, X]$ についても $f(w, x) = 0$ が言える.

2 最大フロー最小カット定理

2.1 maximum flow minimum cut theorem

定理 2.1.1 (maximum flow minimum cut theorem)**.** ネットワーク $N = (D, u, v, c)$ に対して, 最大フローと最小カットの値は一致する. すなわち, f:N のフロー, K:N のカットに対して,

$$f\text{: 最大フロー} \wedge K\text{: 最小カット} \Leftrightarrow \mathrm{val}(f) = \mathrm{cap}(K)$$

Proof. $\Leftarrow$) 系 1.5.2 より従う.

$\Rightarrow$)f: 最大フロー, K: 最小カットとする. 定理 1.6.2 より, D 上に f-augmenting な semipath は存在せず, $X = \{x \in V(D) | {}^{\exists}P{:}f\text{-unsaturated } u\text{-}x \text{ semipath}\}$ とすると $K' = [X, \overline{X}]$ は最小カットとなり,

$$f(a) = \begin{cases} c(a) & \text{if } a \in K' \\ 0 & \text{if } a \in [\overline{X}, X] \end{cases}$$

である. よって系 1.5.3 より $\mathrm{val}(f) = \mathrm{cap}(K') = \mathrm{cap}(K)$ となり示せた. □

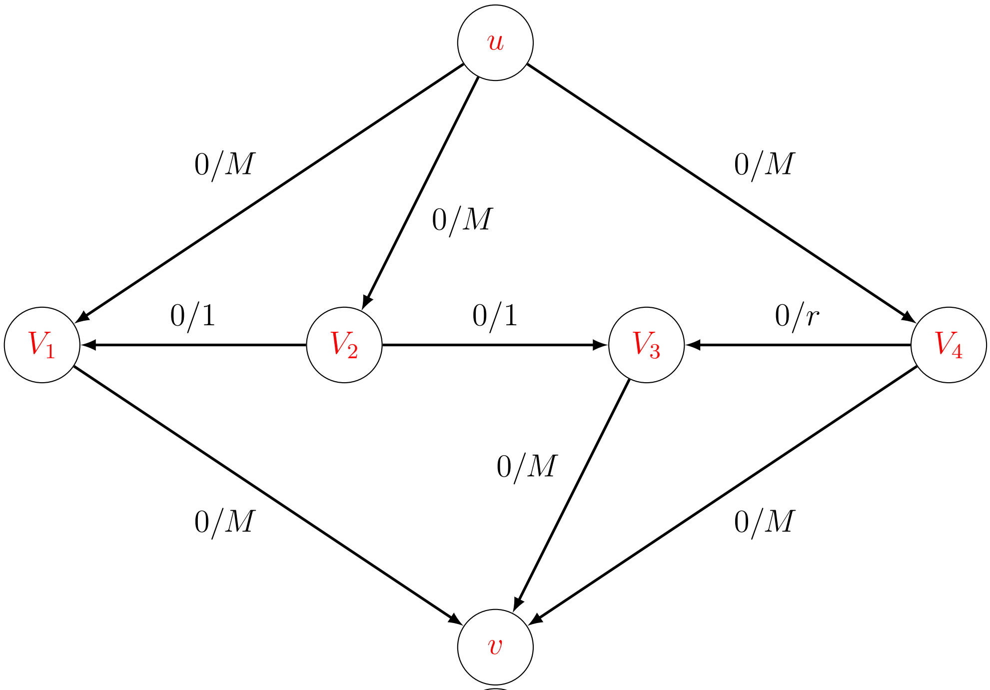
2.2 The Ford-Fulkerson Algorithm

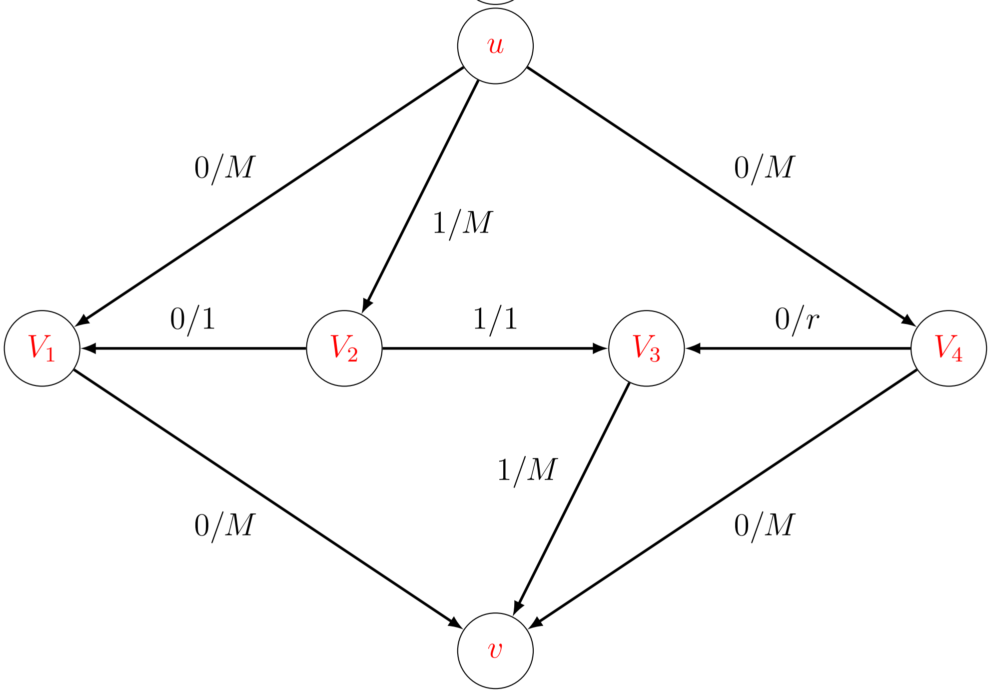
$N = (D, u, v, c)$: ネットワークとする.

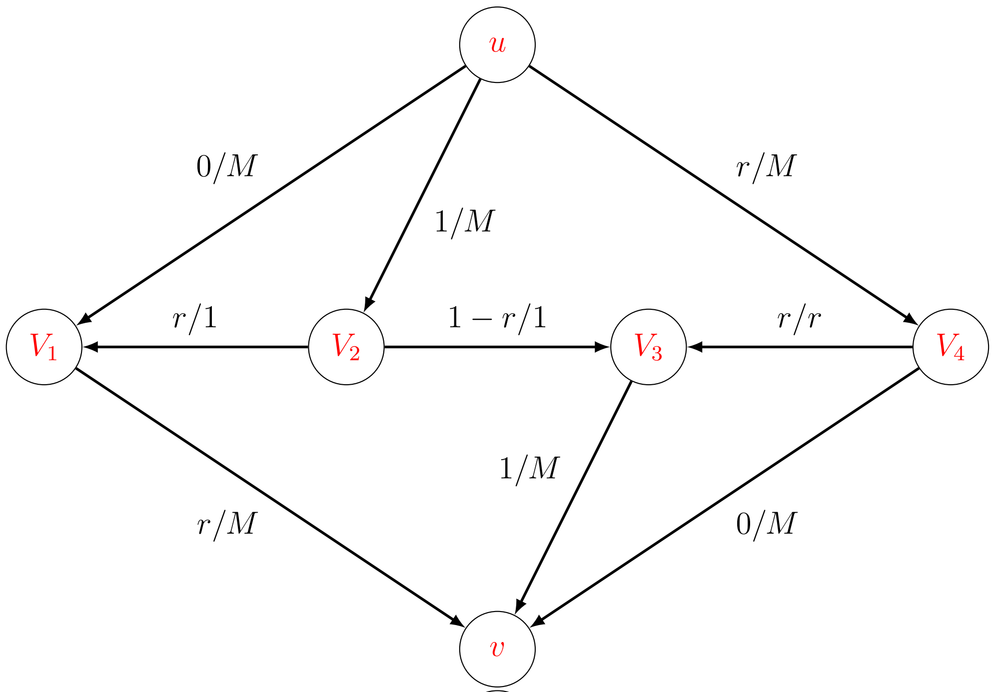
1. flow f を一つとる.
2. f-augmenting な semipath を見つける. 見つけれなかった場合は終了する.
3. 定理 1.6.2 の $\Rightarrow$ で作ったように f' を構成する.
4. $f = f'$ として Step 2 に戻る.

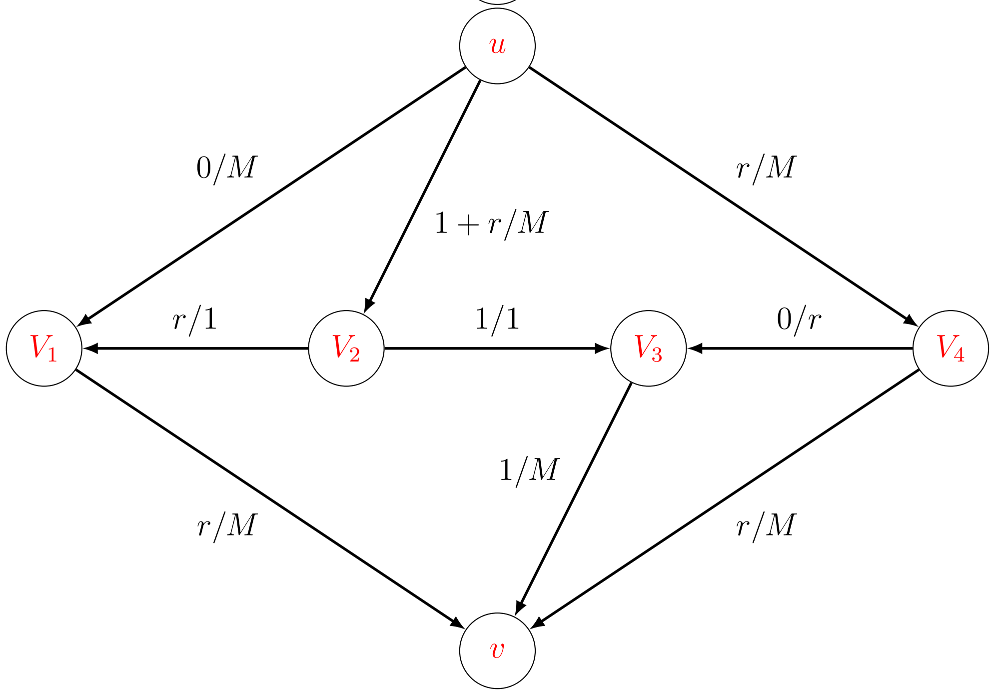
しかしこのアルゴリズムは欠点が多い. 一つはグラフと f-augment semipath の選び方によって, 計算量がとても大きくなるという点. もう一つは容量が無理数だとアルゴリズムが止まらなくなるという点だ.

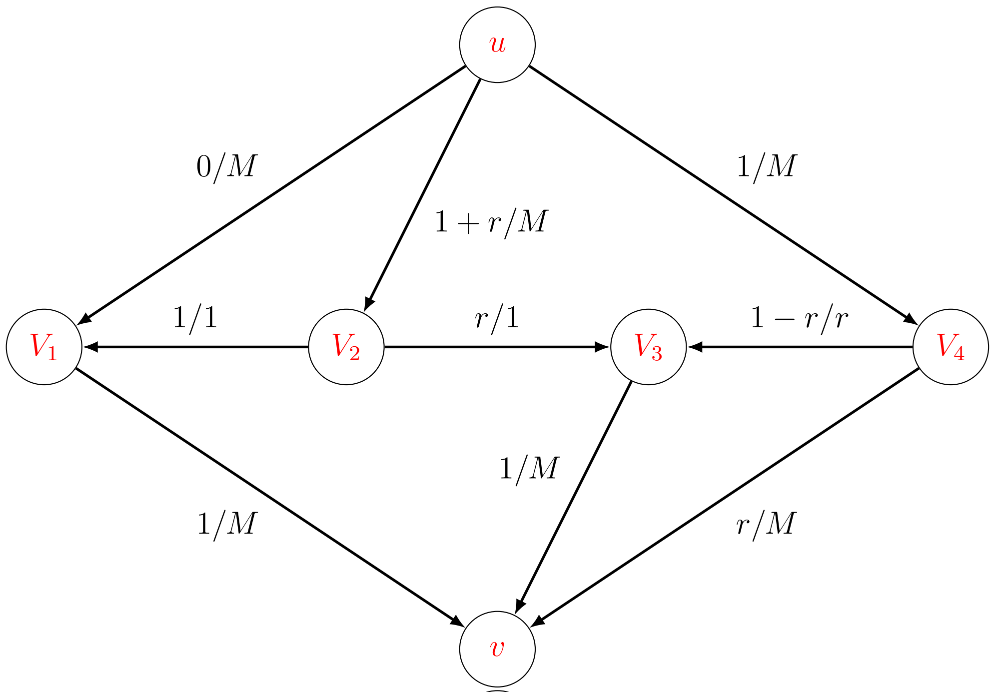
$r = \frac{\sqrt{5}-1}{2}$ とすると,


u
0/M
0/M
0/M
V1
0/1
V2
0/1
V3
0/r
V4
0/M
0/M
0/M
v

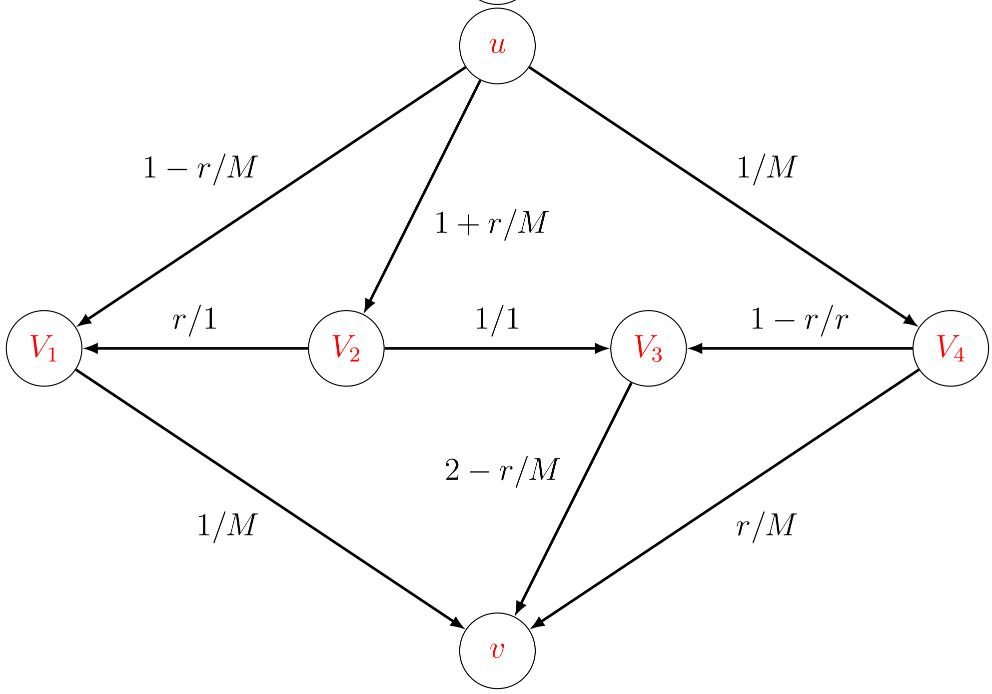

u
0/M
1/M
0/M
V1
0/1
V2
1/1
V3
0/r
V4
1/M
0/M
0/M
v


u
0/M
1/M
r/M
V1
r/1
V2
1 − r/1
V3
r/r
V4
1/M
r/M
0/M
v


u
0/M
1 + r/M
r/M
V1
r/1
V2
1/1
V3
0/r
V4
1/M
r/M
r/M
v



u
$0/M$
$1+r/M$
$1/M$
V_1
$1/1$
V_2
$r/1$
V_3
$1-r/r$
V_4
$1/M$
$1/M$
r/M
v



u
$1-r/M$
$1+r/M$
$1/M$
V_1
$r/1$
V_2
$1/1$
V_3
$1-r/r$
V_4
$2-r/M$
$1/M$
r/M
v

2.3 The Edmonds-Karp Algorithm

$N = (D, u, v, c)$: ネットワークとする. 各点にラベルを付け, ラベル付けされているがスキャンされていない頂点のリスト L を用意する. ラベルは 2 つの値のペアである.

1. flow f を一つとる. N の intermediate vertex w において f-unsaturated な u-w semipath P が存在するときに, P の直前の頂点 x について $(x, w) \in E(P)$ であればラベルは $(x+, \epsilon(w))$, $(w, x) \in E(P)$ であればラベルは $(x-, \epsilon(w))$ とする.

2. u のラベルは $(-, \infty)$ とし, u をリスト L に加える.

3. $L = \emptyset$ ならばとめ. そうでなければ L の最初の元 x(ラベル $(z+, \epsilon(x))$ or $(z+, \epsilon(x))$ を持つ) について,

   3.1. flow f を一つとる.

   3.2. ラベルを付ける. u は $(-, \infty)$ とし, リスト L に加える.

4. v がラベルを持っている場合, Step 5 へ, そうでなければ 3 へ行く.

5. v がラベルを持っている場合, Step 5 へ, そうでなければ 3 へ行く.

6. ラベルを削除し, L から頂点を全て削除し, Step 2 に戻る.