アルゴリズム論(第12回)

http://coconut.sys.eng.shizuoka.ac.jp/ds/07/

安藤和敏 (静岡大学工学部)

2007.07.02

2.3. 最小費用フロー

2.3.1. 最小費用フロー問題

最大フロー問題のときのように，特別な 2 点 s^+ と s^- が指定されている有向グラフ $G = (V, A)$ を考える．ここでは，各枝 $a \in A$ に対して容量 $c(a)$ の他に，枝 a に 1 単位のフローを流すのにかかる**費用** $\gamma(a)$ が与えられている．このネットワークを $\mathcal{N} = (G = (V, A), s^+, s^-, c, \gamma)$ と書く．

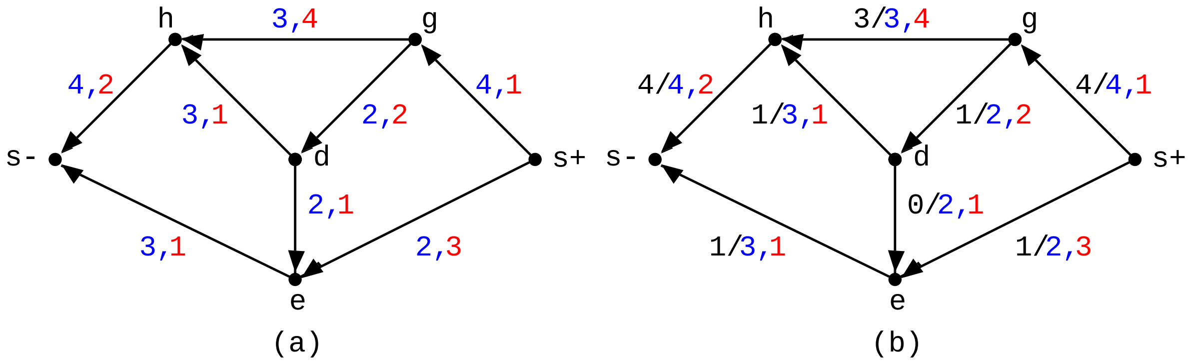

図 2.1: (a) 最小費用流問題で考えるネットワーク $\mathcal{N} = (G = (V, A), s^+, s^-, c, \gamma)$．各枝 $a \in A$ に付された数値は $c(a), \gamma(a)$ を表す; (b) $\mathcal{N}$ 中のフロー φ．各枝 $a \in A$ に付された数値は $\varphi(a), c(a), \gamma(a)$ を表す．φ の費用は, $1\times4+3\times1+4\times3+2\times1+1\times1+2\times4+1\times1+1\times0 = 31$.

ネットワーク $\mathcal{N} = (G = (V, A), s^+, s^-, c, \gamma)$ と実数 $\hat{v}$ が与えられたとき，流量が $\hat{v}$ であるようなフロー φ の中で，フローの費用

$$\sum_{a \in A} \gamma(a)\varphi(a)$$

が最小になるものを**最小費用フロー** (minimum cost flow) と呼び，最大費用フローを求める問題を**最小費用フロー問題** (minimum-cost flow problem) と呼ぶ．

2.3.2. 補助グラフと最適性の条件

最小費用フロー問題を解くためのアルゴリズムには, プライマル法, ネットワークシンプレックス法, などがあるが, ここでは, **プライマル法**と**プライマル-デュアル法**について学ぶ.

これらのアルゴリズムでは, 以下に説明する補助ネットワークを用いる.

ネットワーク $\mathcal{N} = (G = (V, A), s^+, s^-, c, \gamma)$ 上のフロー φ に対して, 補助ネットワーク $\mathcal{N}_\varphi$ は以下のように構成される. 枝集合 A_φ は, 最大フロー問題の補助ネットワークと同様に,

$$\begin{aligned} A_\varphi &= A_\varphi^+ \cup A_\varphi^-, && (2.33)\\ & A_\varphi^+ = \{a | a \in A, \varphi(a) < c(a)\} && (2.34)\\ & A_\varphi^- = \{\bar{a} | a \in A, 0 < \varphi(a)\} \quad (\bar{a}\text{: } a \text{ の逆向き枝}) && (2.35) \end{aligned}$$

で与えられ, 容量関数 $c_\varphi\colon A_\varphi \to \mathbb{R}$ も最大フロー問題の補助ネットワークと同様に,

$$c_\varphi(a) = \begin{cases} c(a) - \varphi(a) & \text{if } a \in A_\varphi^+ \text{ のとき,} \\ \varphi(a) & \text{if } \bar{a} \in A_\varphi^- \text{ のとき.} \end{cases} \tag{2.36}$$

で定義される. ここでは, 新たに費用関数 $\gamma_\varphi\colon A_\varphi \to \mathbb{R}$ を

$$\gamma_\varphi(a) = \begin{cases} \gamma(a) & \text{if } a \in A_\varphi^+ \text{ のとき,} \\ -\gamma(a) & \text{if } \bar{a} \in A_\varphi^- \text{ のとき.} \end{cases} \tag{2.37}$$

で定義する.

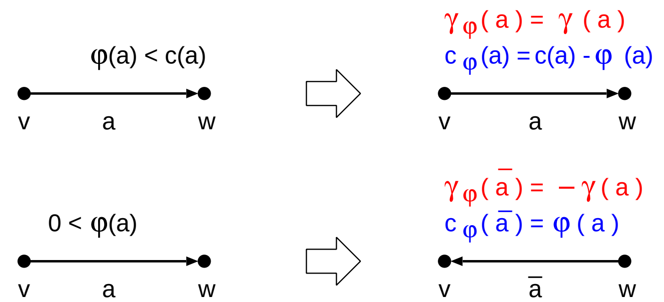

図 2.2: 補助ネットワークの作り方

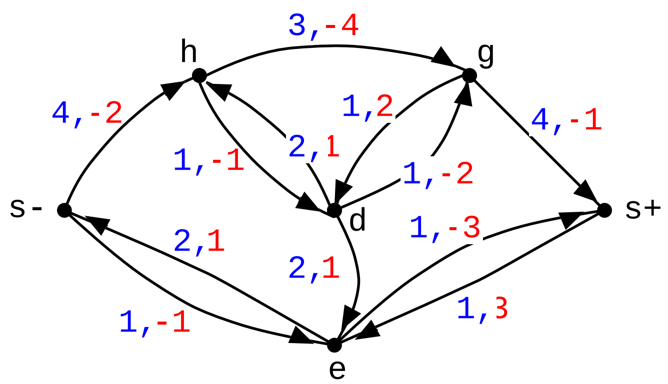

図 2.3: 補助ネットワーク $\mathcal{N}_\varphi$ の例

最小費用フローは補助ネットワークを用いて, 以下のように特徴付けられる.

定理 2.12: 2端子ネットワーク $\mathcal{N}=(G=(V,A),s^+,s^-,c,\gamma)$ 中のフロー φ がそれと同じ流量をもつ $\mathcal{N}$ 中のすべてのフローのうちで費用最小であるための必要十分条件は, φ に関する補助ネットワーク $\mathcal{N}_\varphi$ 中に負の長さの有向閉路が存在しないことである.

上の定理の必要性の証明は簡単である. $\mathcal{N}$ 中のフロー φ に対して,

φ は同じ流量をもつ $\mathcal{N}$ 中のすべてのフローのうちで費用最小である
$\Longrightarrow \mathcal{N}_\varphi$ 中に負の長さの有向閉路が存在しない

を示せすかわりに, その対偶

$\mathcal{N}_\varphi$ 中に負の長さの有向閉路が存在する
$\Longrightarrow \varphi$ は同じ流量をもつ $\mathcal{N}$ 中のすべてのフローのうちで費用最小でない

を示そう. $\mathcal{N}_\varphi$ 中に負の長さの有向閉路 Q が存在すれば, Q に沿ってフローを一巡させれば, φ と同じ流量をもち $(v^*(\varphi)=v^*(\varphi'))$, 費用が φ の費用より小さいフロー φ' を作ることができる. 図 2.4 を見よ. したがって, φ は費用最小ではない.

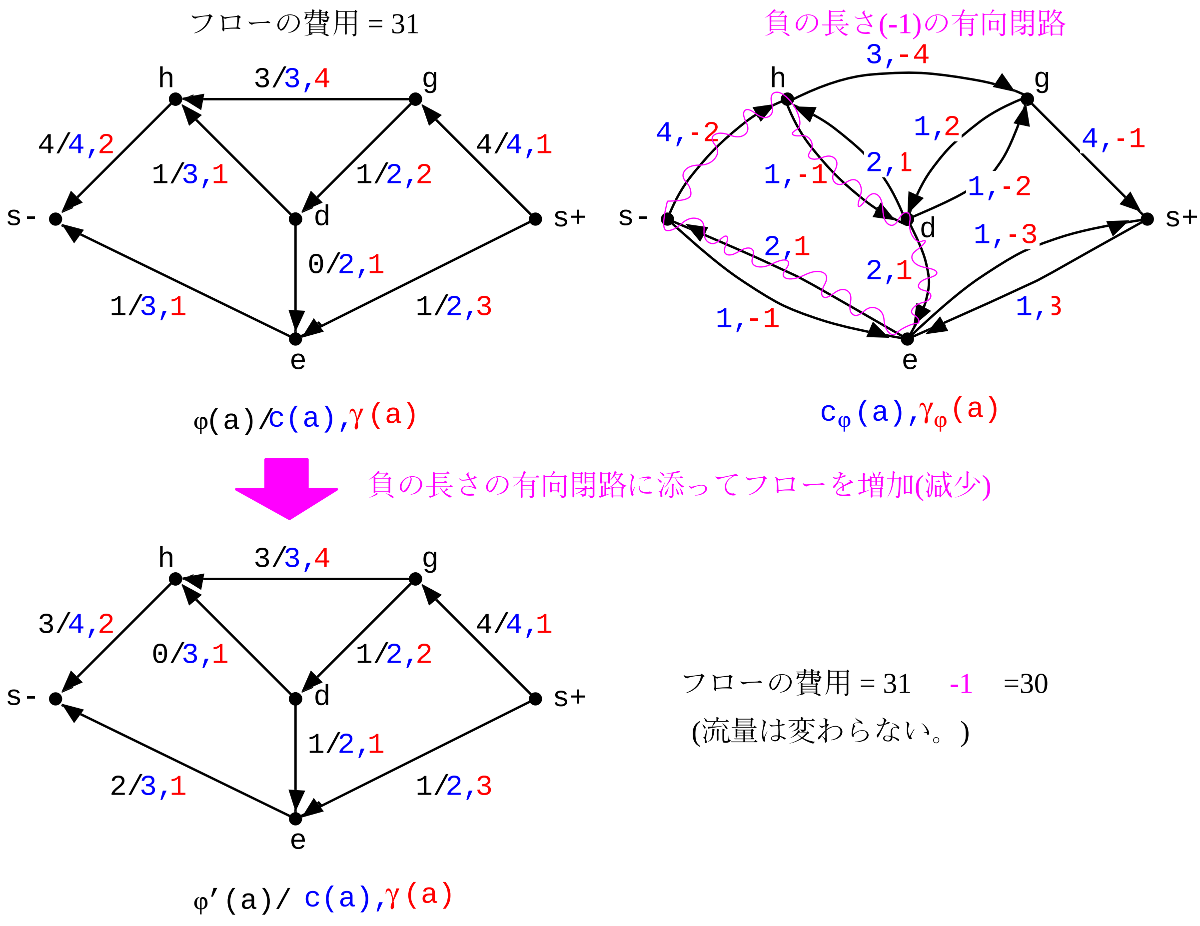

図 2.4: 定理 2.12 の必要性の証明の考え方

十分性

$\mathcal{N}_\varphi$ 中に負の長さの有向閉路が存在しない
$\Longrightarrow \varphi$ は同じ流量をもつ $\mathcal{N}$ 中のすべてのフローのうちで費用最小である

の証明は, 少し難しいので省略する.

流量が $\hat{v}$ であるような適当なフローを φ とする.
補助ネットワーク $\mathcal{N}_\varphi$ を作る.
while $\mathcal{N}_\varphi$ 中に有向閉路 Q が存在する **do**
  $d \leftarrow \min\{c_\varphi(a) \mid a \in Q\}$.
  for Q の各枝 a について **do**
    if a が A_φ^+ の枝 **then**
      $\varphi(a) \leftarrow \varphi(a) + d$.
    else if a が A_φ^- の枝 **then**
      $\varphi(\bar{a}) \leftarrow \varphi(\bar{a}) - d$.
    end if
  end for
  補助ネットワーク $\mathcal{N}_\varphi$ を作り直す.
end while

図 2.5: プライマル法

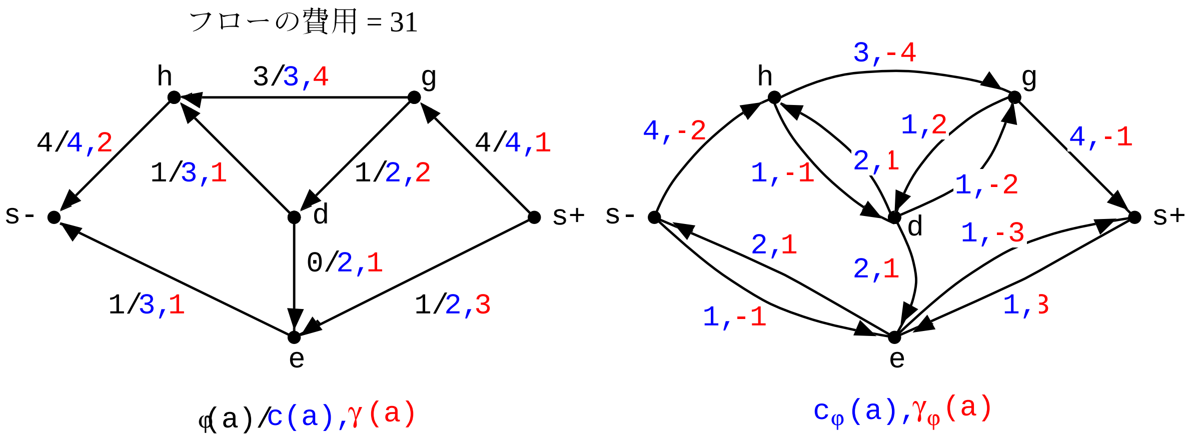

図 2.6: プライマル法の動き (最初のフロー φ と, $\mathcal{N}_\varphi$ の有向閉路)

2.3.3. 最小費用フロー問題に対するアルゴリズム

[プライマル法]

プライマル法は, 定理 2.12 に基づいており, 適当な初期フロー φ から出発して, $\mathcal{N}_\varphi$ の中に負の長さの有向閉路が存在する限り, その有向閉路にそってフロー φ を更新してゆくというアルゴリズムである.

プライマル法の各繰り返しにおいて, 負の長さの有向閉路を見付けなければならない. 補助ネットワークの枝には長さが負のものもあるので, 負の長さの有向閉路を見付けるためには, ベルマン-フォード法を用いなければならない. (もちろん, 紙の上で解くときには,「直観」で負の長さの有向閉路を見付けてもよい. ただし, 負の長さの有向閉路が存在しないということの保証を得るためには, 教科書の定理 2.4 の条件を満すポテンシャルを見付けなければならない.)

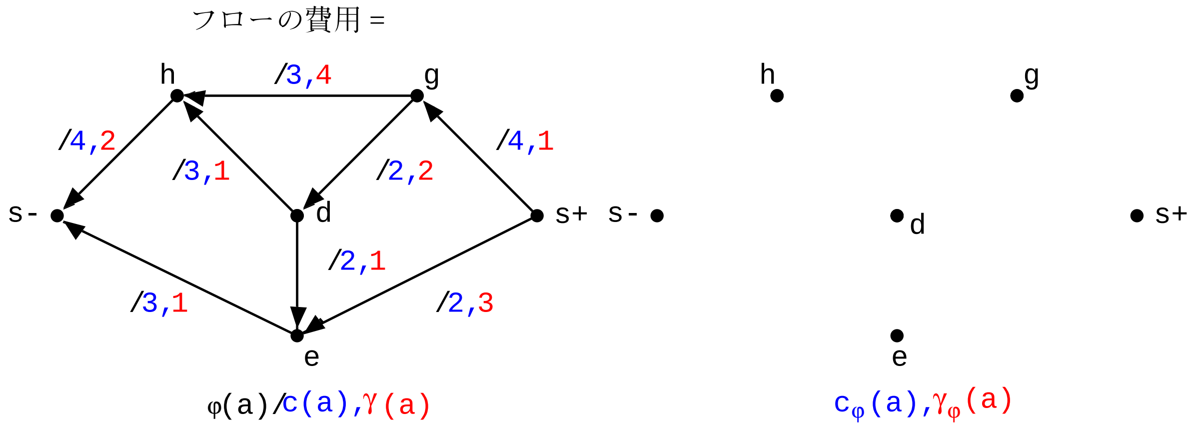

図 2.7: プライマル法の動き (フロー φ と, $\mathcal{N}_\varphi$ の有向閉路)

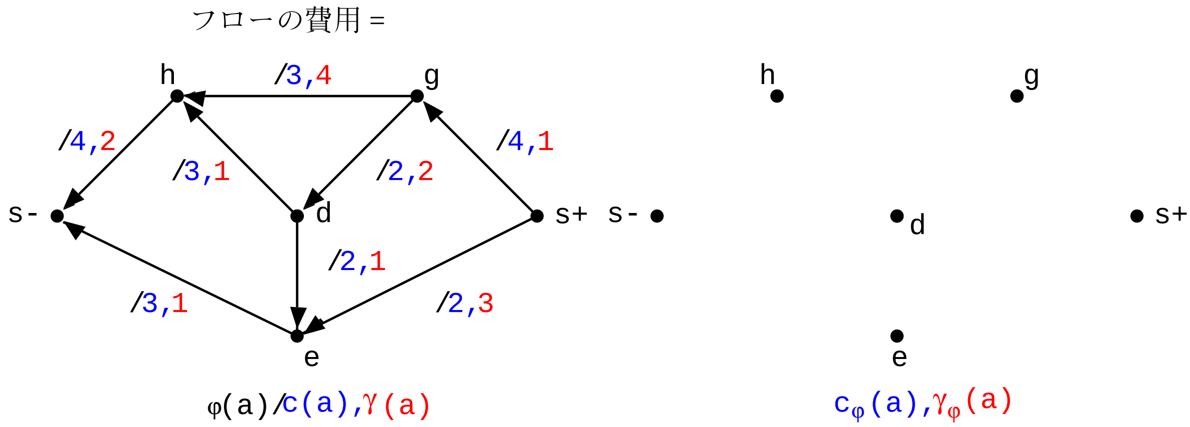

図 2.8: プライマル法の動き (フロー φ と, $\mathcal{N}_\varphi$ の有向閉路)

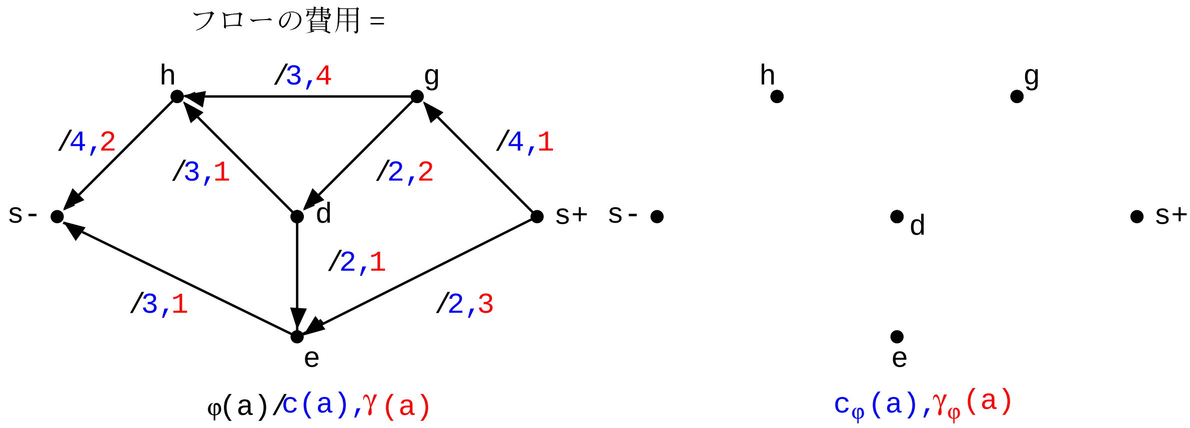

図 2.9: プライマル法の動き (フロー φ と, $\mathcal{N}_\varphi$ の有向閉路)