# グラフとネットワーク (第10回)

http://coconut.sys.eng.shizuoka.ac.jp/gn/08/

安藤和敏 (静岡大学工学部)

2008.12.04

## 2.1. 木と道

### 2.1.1. 最短路問題 (ダイクストラ法)

有向グラフ $G = (V, A)$ 上の各枝 $a \in A$ に対して，その長さ $l(a)$ を指定する枝長関数 $l: A \to \mathbb{R}$ が与えられているとする．このようなネットワークを $\mathcal{N} = (G = (V, A), l)$ と書くことにする．

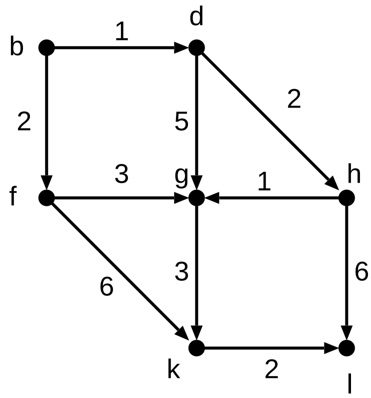
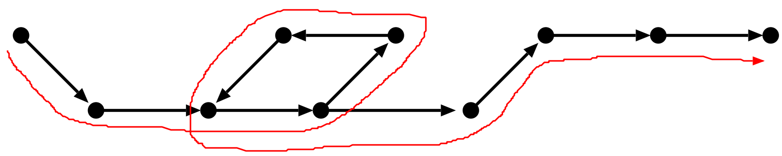



図 2.1: $G$ と $l$



$G$ の中の有向道

$$P = (v_0, a_1, v_1, a_2, v_2, a_3, \cdots, a_k, v_k)$$

に対して，$\sum_{i=1}^{k} l(a_i)$ を $P$ の**長さ**と呼ぶ．**最短路問題** (shortest path problem) とは，与えられた 2 点 $u, v \in V$ に対して，$u$ から $v$ への長さが最小の有向道を見出す問題である．

点集合上で定義される関数 $p: V \to \mathbb{R}$ を**ポテンシャル** (potential) と呼ぶ．

**補題 2.1**: 任意なポテンシャル $p\colon V \to \mathbb{R}$ に対し, 関数 $l_p\colon A \to \mathbb{R}$ を

$$l_p(a) = l(a) + p(\partial^+ a) - p(\partial^- a) \tag{2.11}$$

によって定義する. $\mathcal{N}$ の中の点 $u$ から $v$ への任意な有向道 $P$ に対して, 関数 $l$ と $l_p$ に関する $P$ の長さをそれぞれ $l(P)$ と $l_p(P)$ とすると,

$$l_p(P) = l(P) + p(u) - p(v) \tag{2.12}$$

が成り立つ. □

**補題 2.2**: $P$ を点 $u$ から点 $v$ への有向道とする. もし, あるポテンシャル $p\colon V \to \mathbb{R}$ が存在して, (2.1) で定義される $l_p$ が

(i) グラフ $G$ の全ての枝 $a \in A$ に対して $l_p(a) \geq 0$, かつ,

(ii) $P$ 上の全ての枝 $a$ に対して, $l_p(a) = 0$

を満足するならば, $P$ は点 $u$ から点 $v$ への最短路である. □

すべての枝の長さが非負, すなわち, $l(a) \geq 0$ $(a \in A)$, であるときに使える解法として **ダイクストラ法** が有名である. これは, 与えられた 1 点から残りのすべての点への最短路を求める.

---

**Algorithm 1** ダイクストラ法 (始点を $v_0$ とする)

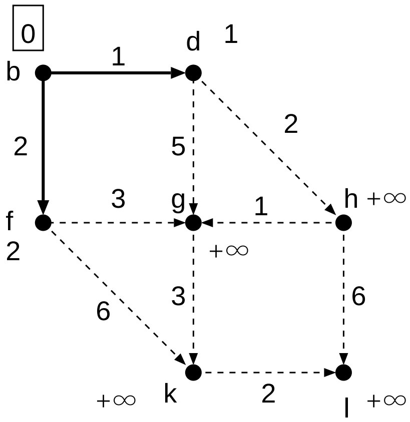
---

**入力:** 単純な有向グラフ $G = (V, A)$, 枝長関数 $l\colon A \to \mathbb{R}_+$.
**出力:** $v_0$ からその他の各点 $v$ への最短路, 及び, 最短路長.

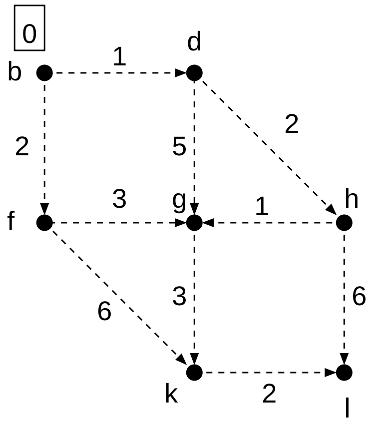
1: $U \leftarrow \{v_0\},\ W \leftarrow \emptyset,\ p(v_0) \leftarrow 0,\ p(u) \leftarrow +\infty\ (u \in V \setminus \{v_0\})$.
2: $U = \emptyset$ ならば停止.
そうでなければ, $U$ の点のなかで $p$ の値が最小であるものを 1 つ選び, それを $w$ とする. 点 $w$ から出る枝 $a = (w, x)$ で $x \notin W$ であるような各枝に対して, 以下の (*) を実行する.
  (*) $p(x) > p(w) + l(w, x)$ ならば
    $q(x) \leftarrow a,\ p(x) \leftarrow p(w) + l(w, x)$,
    $U \leftarrow U \cup \{x\}$.
3: $W \leftarrow W \cup \{w\},\ U \leftarrow U \setminus \{w\}$ として Step 2 に行く.

---

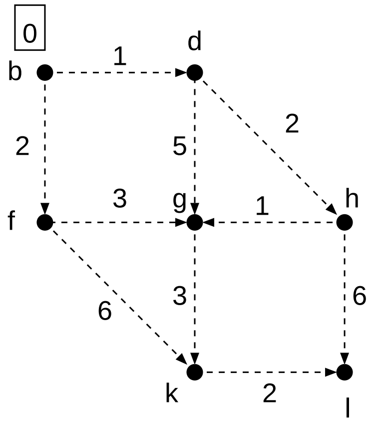
図 2.1 で示されるグラフ $G$ と枝長関数 $l$ を例題として, アルゴリズムの動作を見てみよう. ここで, $v_0 = b$ とする.



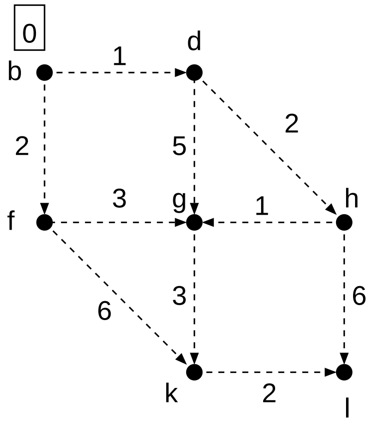

第 1 回目の Step 3 終了時



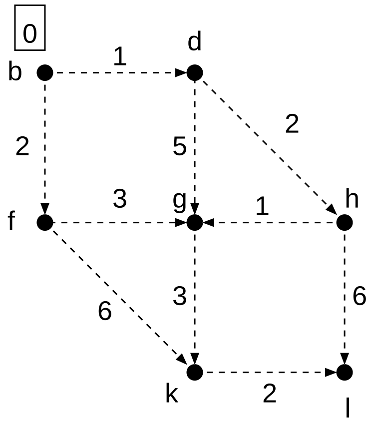

第 2 回目の Step 3 終了時



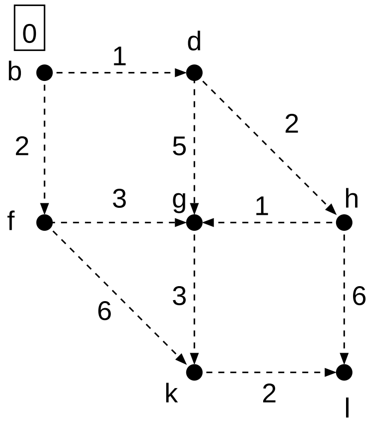

第 3 回目の Step 3 終了時



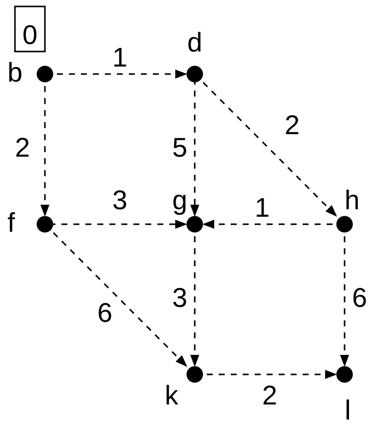

第 4 回目の Step 3 終了時




第 5 回目の Step 3 終了時




第 6 回目の Step 3 終了時




第 7 回目の Step 3 終了時