グラフとネットワーク(第10回)

http://coconut.sys.eng.shizuoka.ac.jp/gnB/

安藤和敏 (静岡大学工学部)

2004.12.09

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 $l\colon A \to \mathbb{R}$ 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 で示されるグラフ G と枝長関数 l を考える.

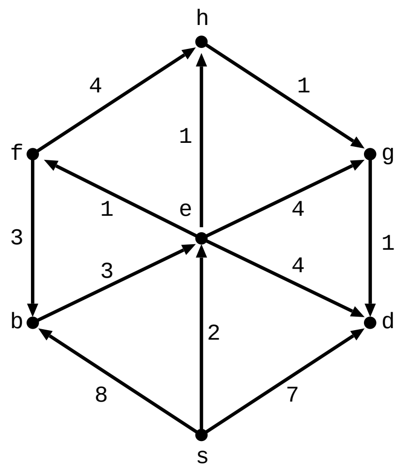
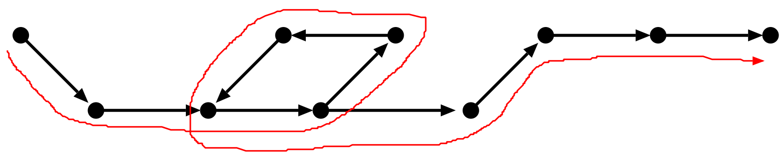


図 2.1: G と l



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 $p\colon V \to \mathbb{R}$ 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. □

注意 2.a: 補題 *2.1* によれば, 「道 P が l に関して点 u から点 v への最短路であれば, どのようにポテンシャル $p\colon V \to \mathbb{R}$ を定めても P が l_p に関して点 u から点 v への最短路である」ということがわかる. 同じように, 「道 P があるポテンシャル p に対する l_p に関しての点 u から点 v への最短路であれば, P は l に関して u から v への最短路である」. □

補題 2.2: 点 u から点 v への有向道 P に対して, 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 が非負関数であり, かつ, P 上の各枝 a に対して $l_p(a) = 0$ であるとすると, P は点 u から点 v への最短路である.

(証明) l_p は非負関数, すなわち, $l_p(a) \geq 0$ $(a \in A)$ であるから, u から v への任意の有向道 P' に対して $l_p(P') \geq 0$. その一方で $l_p(P) = 0$ であるから, このポテンシャル p で定まる l_p に関して P は u から v への最短路である.

注意 2.a より, P は l に関して u から 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)

Require: 単純な有向グラフ $G = (V, A)$, 枝長関数 $l\colon A \to \mathbb{R}_+$.

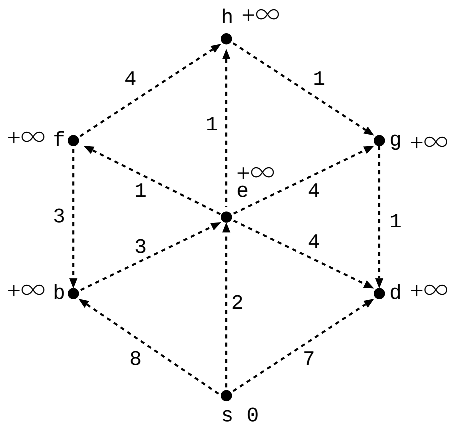
Ensure: v_0 からその他の各点 v への最短路, 及び, 最短路長.

1: $U \leftarrow \{v_0\},\ W \leftarrow \emptyset,\ p(v_0) \leftarrow 0,\ p(u) \leftarrow +\infty\ (u \in V \setminus \{v_0\})$.

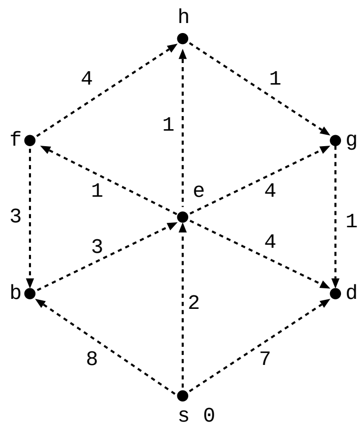
2: $U = \emptyset$ ならば停止.
そうでなければ, U の点のなかで p の値が最小であるものを 1 つ選び, それを w とする. 点 w から出る枝 $a = (w, x)$ で $x \notin W$ であるような各枝に対して, 以下の (*) を実行する.
  (*) $p(x) > p(w) + l(w, x)$ ならば
    $q(x) \leftarrow w,\ p(x) \leftarrow p(w) + l(w, x)$,
    $U \leftarrow U \cup \{x\}$.

3: $W \leftarrow W \cup \{w\},\ U \leftarrow U \setminus \{w\}$ として Step 2 に行く.

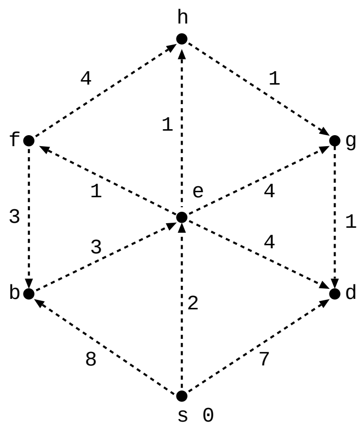
アルゴリズムの動作を例題を用いて見てみよう.



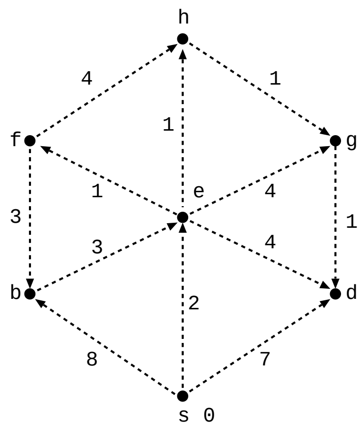
Step 1 の終了時



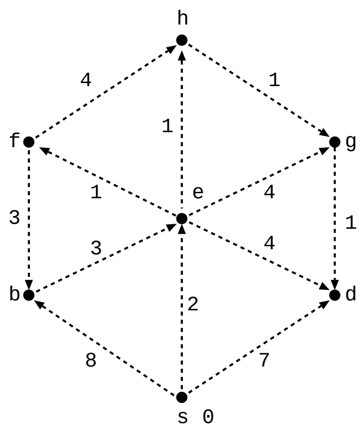
1 回目の Step 2 の終了時



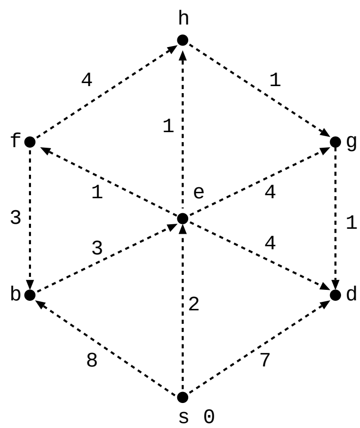
2 回目の Step 2 の終了時



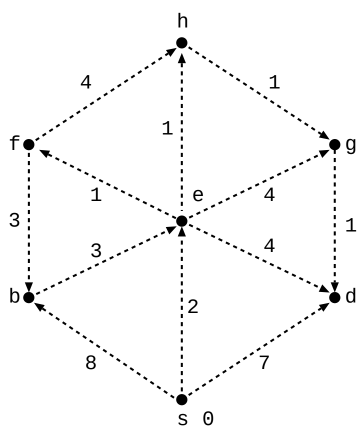
3 回目の Step 2 の終了時



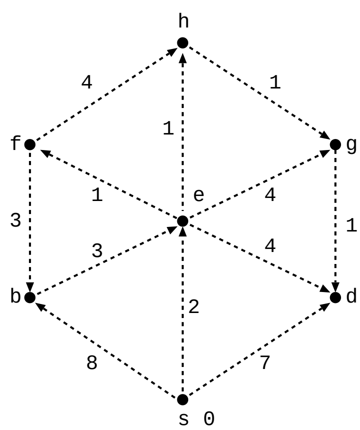
4 回目の Step 2 の終了時



5 回目の Step 2 の終了時



6 回目の Step 2 の終了時



7 回目の Step 2 の終了時

補題 2.3: ダイクストラ法の実行によって点が $v_0, v_1, v_2, \cdots$ の順に W に取り込まれたとすると, 最終的に得られる p に対して

$$0 = p(v_0) \leq p(v_1) \leq p(v_2) \leq \cdots \tag{2.13}$$

が成り立つ.

(証明) ダイクストラ法実行中の W に対して,

$$\max\{p(u) \mid u \in W\} \leq \min\{p(u) \mid u \in V \setminus W\} \tag{2.14}$$

が成り立つことを帰納法で証明する.

第 1 回目の Step 2 と Step 3 が終った時点では, $W = \{v_0\}$ である. また p は, v_0 に隣接する点 u に対しては, $p(u) = l(v_0, u)$ であり, その他の点 v に対しては, $p(u) = +\infty$ である. ゆえに,

$$\begin{aligned} \max\{p(u) \mid u \in W\} &= \max\{p(u) \mid u \in \{v_0\}\} \\ &= p(v_0) \\ &= 0 \\ &\leq \min\{l(v_0, u) \mid u \text{ は } v_0 \text{ に隣接する点 }\} \\ &= \min\{p(u) \mid u \in V \setminus W\}. \end{aligned}$$

$i \geq 1$ 回目の Step 2 と Step 3 が終った時点の W に対して (2.14) が成り立つと仮定して, $i+1$ 回目の Step 2 と Step 3 が終った時点の W でも (2.14) が成り立つことを証明しよう. $i+1$ 回目の Step 2 において w が選ばれたとする. p が更新される前に

$$p(w) = \min\{p(u) \mid u \in U\}$$

であったし, $W \cup U$ 以外の点 u では $p(u) = +\infty$ であるから,

$$p(w) = \min\{p(u) \mid u \in V \setminus W\}.$$

よって,

$$\max\{p(u) \mid u \in W\} \leq p(w) \leq \min\{p(u) \mid u \in V \setminus (W \cup \{w\})\}. \tag{2.15}$$

ゆえに,

$$\max\{p(u) \mid u \in W \cup \{w\}\} \leq \min\{p(u) \mid u \in V \setminus (W \cup \{w\})\}. \tag{2.16}$$

この回の Step 3 で $W \leftarrow W \cup \{w\}$ という更新が行われるので, 式 (2.14) が, $i+1$ 回目の Step 3 が終ったときも成り立つ. □

補題 2.4: ダイクストラ法の実行中に得られる W, U, p, q に対し,

(i) G の部分グラフ $G_W = (W \cup U, \delta^+ W)$ 上で, (2.1) で定義される l_p: $\delta^+ W \to \mathbb{R}$ は $\delta^+ W$ 上で非負であり, 各点 $u \in (W \cup U) \setminus \{v_0\}$ に対して $l_p(q(u), u) = 0$ が成り立つ.

(ii) すべての点 $u \in (W \cup U) \setminus \{v_0\}$ に対して定まる枝 $(q(u), u)$ の全体は点 v_0 を根とする有向木である.

(証明) (i) $w \in W$ として, w から出る任意な枝 (w, x) を考える. w はある回の Step 3 において W に加えられたはずである. その回の Step 2 を考えよう.

もし $x \in W$ ならば, 補題 2.3 によって $p(x) \leq p(w)$ であるから,

$$l_p(w, x) = l(w, x) + p(w) - p(x) \geq 0.$$

もし $x \notin W$ ならば, その回のステップ 2 において, (*) $p(x) > p(w) + l(w, x)$ ならば

$q(x) \leftarrow w,\ p(x) \leftarrow p(w) + l(w, x),$

$U \leftarrow U \cup \{x\}.$

を実行したはずである. (*) の直後では, $p(x) > p(w) + l(w, x)$ だったとしてもそうでなかったとしても, $l_p(w, x) \geq 0$ が成り立っている.

$p(x)$ はその後の Step 2 と Step 3 の繰返しにおいて増えることはないし, $p(w)$ は変化しないことに注意すると, その後のいかなる時点における p に対しても

$$l_p(w, x) = l(w, x) + p(w) - p(x) \geq 0$$

が成り立つことがわかる.

Step 2 において, $q(x) \leftarrow w$ の更新が起こるのと同時に $p(x) \leftarrow p(w) + l(w, x)$ という更新も行われるしたがって,

$$\begin{array}{rcl} l_p(q(x), x) & = & l_p(w, x) \\ & = & l(w, x) + p(w) - p(x) \\ & = & l(w, x) + p(w) - p(w) - l(w, x) \\ & = & 0. \end{array}$$

(ii) 任意な $u \in (W \cup U) \setminus \{v_0\}$ から出発して, 枝 $(q(u), u)$ を逆にたどっていく道を考えよう. この道は, v_0 に到達するか, あるいは, 最初の点 u に戻るかのどちらかであるが, 補題 2.3 における点の添字付けを考えると, 最初の点に戻るということはあり得ない. □