グラフとネットワーク (第 11 回)

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

安藤和敏 (静岡大学工学部)

2005.01.13

2.1. 木と道

2.1.1. 最短路問題 (ベルマン-フォード法)

図 2.1 (教科書の図 2.8 と同じ) のように, 負の長さを持つ枝があるネットワークに対しては, ダイクストラ法は必ずしも最短路を見出さない. 本日の講義では, 長さが負であるような枝があるネットワークにおいても, 最短路を見出すアルゴリズム (ベルマン-フォード法) について紹介する. 負の長さの枝の存在は非現実的に思えるかも知れないが, もっと

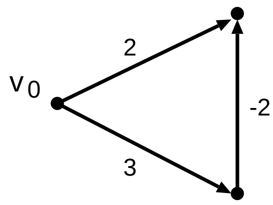


図 2.1: 負の長さの枝があるネットワーク

複雑な問題を解くためにそのようなネットワークを扱う必要がある.

図 2.2 のように, ネットワークに負の長さの有向閉路が存在する場合, 最短路問題は解を持たない. では, 負の長さの有向閉路の存在は許すけれども, 問題を最短路として初等

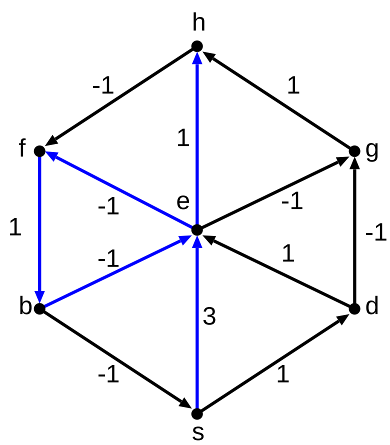


図 2.2: 負の長さの有向閉路が存在するネットワーク

的な有向道 (⇒ p. 13) の中で最短なものを見付けることとしたらどうだろう. しかし, それば解くのが非常に困難な問題になる.

ここでは, 問題を

> 「負の長さの閉路を見付けるか, または, 始点からそれ以外の全ての点への最短路を見付ける」

と設定しよう．この問題を解くアルゴリズムとしては, ベルマン-フォード法 (アルゴリズム 1 を見よ) が知られている．

ベルマン-フォード法を図 2.3 のグラフに対して動かしてみよう．

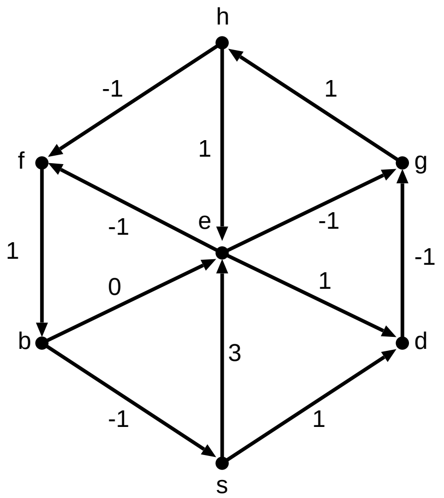


図 2.3: ネットワーク $\mathcal{N} = (G = (V, A), l)$

アルゴリズム 1 ベルマン-フォード法 (始点を v_0 とする)

入力: ネットワーク $\mathcal{N} = (G = (V, A), l)$.

出力: もしあれば負の長さの有向閉路を, そうでなければ v_0 からその他の各点 v への最短路, 及び, 最短路長.

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

2: 各枝 $(v, w) \in A$ に対して,

  (*) $p(w) > p(v) + l(v, w)$ ならば

   $p(w) \leftarrow p(v) + l(v, w),\ q(w) \leftarrow v.$

3: (i) Step 2 で p の更新 (*) が全くされなければ停止する．

  (ii) p が更新されたとき,

   (a) $k < n(= |V|)$ ならば $k \leftarrow k + 1$ として Step 2 へ戻り,

   (b) $k = n$ ならば停止する (このとき負の長さの有向閉路が存在する).

始点を $v_0 = s$ としてベルマン-フォード法を実行すると, Step 1 の終了時には, 図 2.4 のようになる．

講義中で示すように, これに続くステップを実行した結果を示せ．ただし, Step 2 で枝を調べる順序は

s から出る枝, b から出る枝, d から出る枝, e から出る枝, f から出る枝, g から出る枝, h から出る枝

とする．

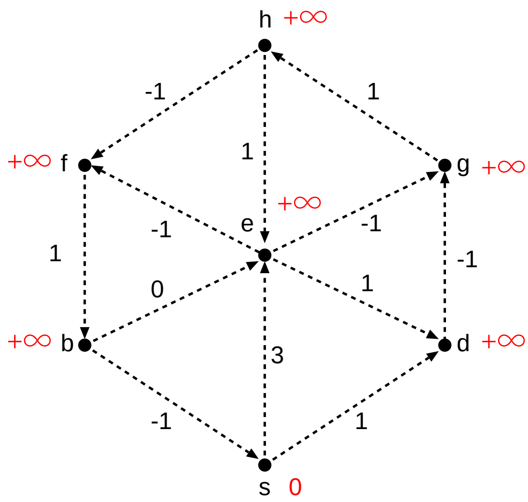
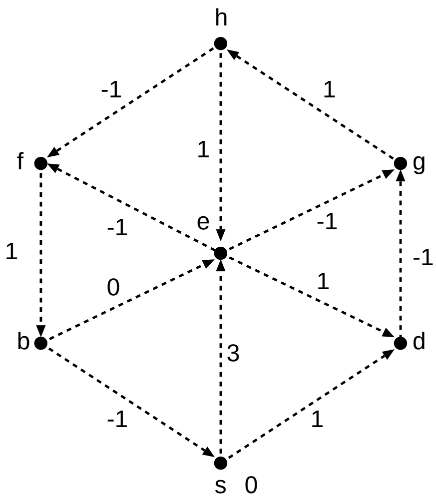
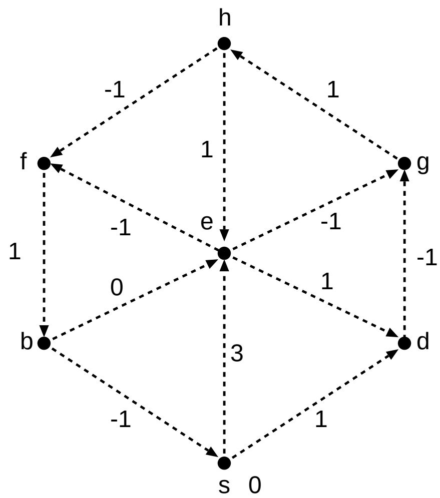


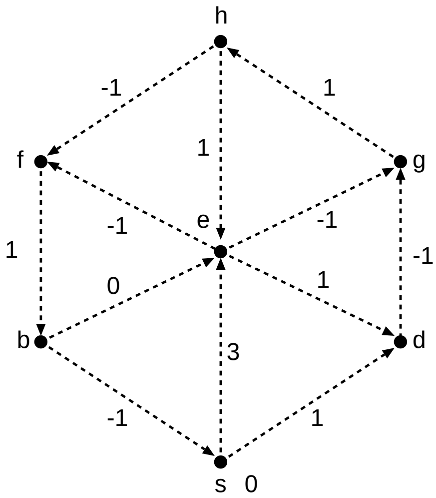
図 2.4: Step 1 終了時



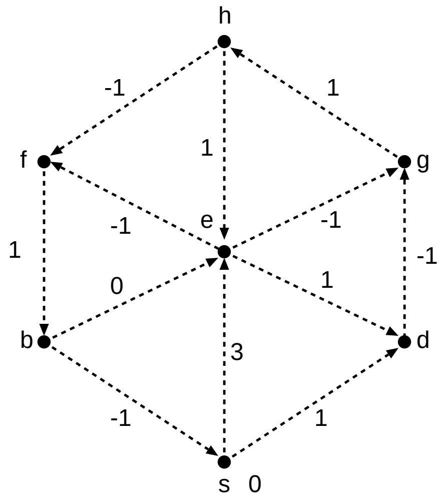
1 回目の Step 2 終了時



2 回目の Step 2 終了時



3 回目の Step 2 終了時



4 回目の Step 2 終了時

Step 3 (i) でベルマン-フォード法が終了したとき, 全ての枝 $(v,w) \in A$ に対して $p(v) + l(v,w) \geq p(w)$ が成り立つ. 即ち,

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

さらに, $(q(u), u)$ $(u \in V \setminus \{v_0\})$ は v_0 を根とする有向木を成し, この有向木上の枝 (v, w) に対しては,

$$l_p(v, w) = l(v, w) + p(v) - p(w) = 0.$$

補題 2.2(⇒ p. 43) から, v_0 からこの有向木上の道が最短路である.