グラフとネットワーク (第11回)

http://coconut.sys.eng.shizuoka.ac.jp/gn/06/

安藤和敏 (静岡大学工学部)

2007.01.23

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 (べき乗法とベルマン-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.

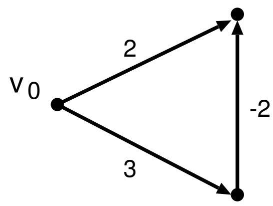

図 2.1: 負の長さの枝があるネットワーク

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.

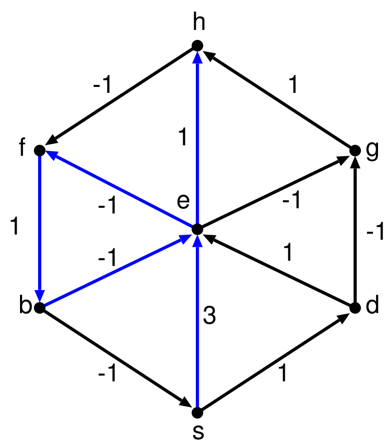

図 2.2: 負の長さの有向閉路が存在するネットワーク

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, べき乗法とベルマン-フォード法が知られている.

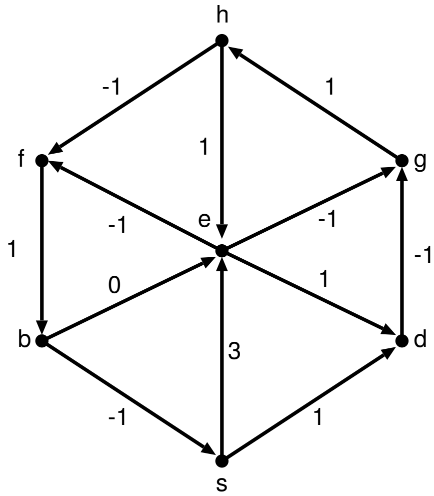

図 2.3: ネットワーク $\mathcal{N} = (G = (V, A), l)$

2.2. ベルマン-フォード法

ベルマン-フォード法 (アルゴリズム 1) は, べき乗法の改良版と考えることができる.

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).

ベルマン-フォード法を, 始点を $v_0 = s$ として図 2.4 のグラフに対して実行した結果は表 2.1 のようになる. さらに, 実行結果を図で表現すると, 図 2.5 のようになる.

Step 2 における枝の選択の順番は任意であるが, 例えば次のような順序で枝を調べてみよう.

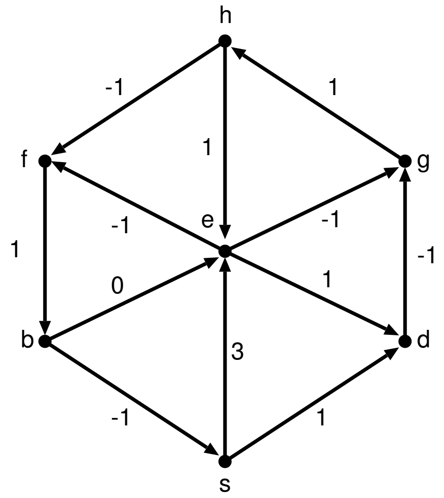

図 2.4: ネットワーク $\mathcal{N}=(G=(V,A),l)$

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.

この順序で枝を調べた場合, ベルマン-フォード法の実行の経過は表 2.1 のようになる.

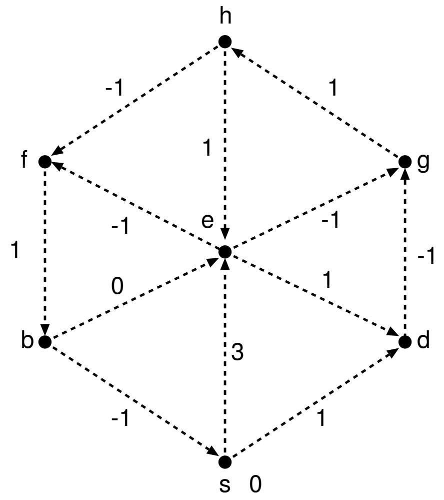

図 2.5: p と q の図的表現 (各自で記入せよ).

表 2.1: ベルマン-フォード法の動き (各自で記入せよ).

		s	b	d	e	f	g	h
	p	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
	q	—	—	—	—	—	—	—
$k = 1$	p							
	q							
$k = 2$	p							
	q							
$k = 3$	p							
	q							
$k = 4$	p							
	q							
$k = 5$	p							
	q							
$k = 6$	p							
	q							
$k = 7$	p							
	q							