オペレーションズ・リサーチ基礎

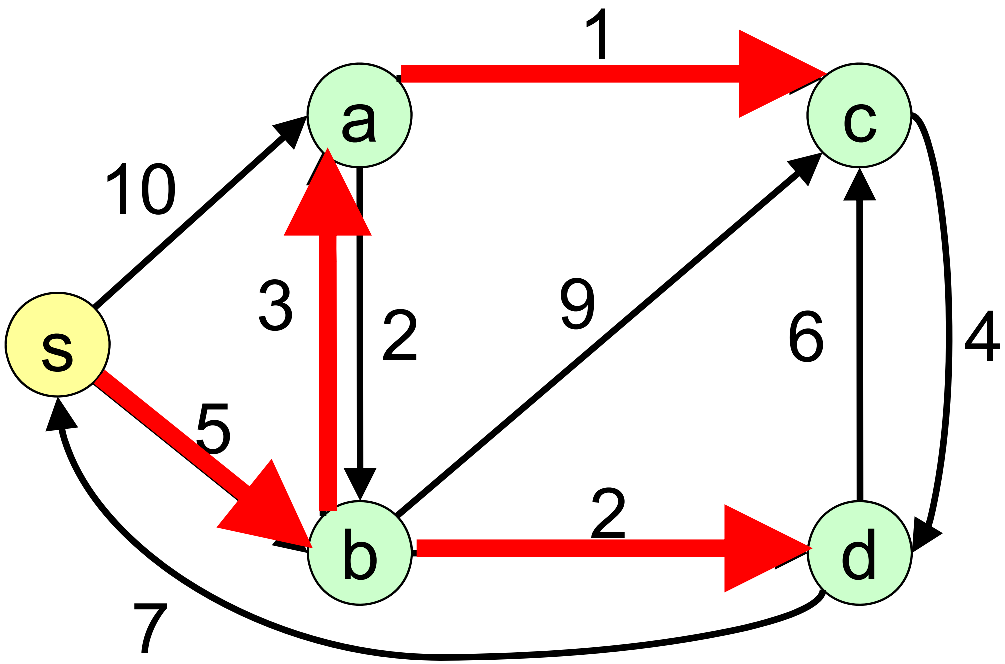
ネットワーク最適化：最短路問題
負の枝長を許した場合

塩浦昭義
東京工業大学 経営工学系
shioura.a.aa@m.titech.ac.jp

単一始点全終点 最短路問題

枝の長さが
負の場合も扱う

- 入力 : 有向グラフ G=(V, E)
  各枝の長さ $\ell(e)$ $(e \in E)$, 始点 s∈V
- 出力 : s から すべての頂点 v への最短路P(v)とその長さd(v)



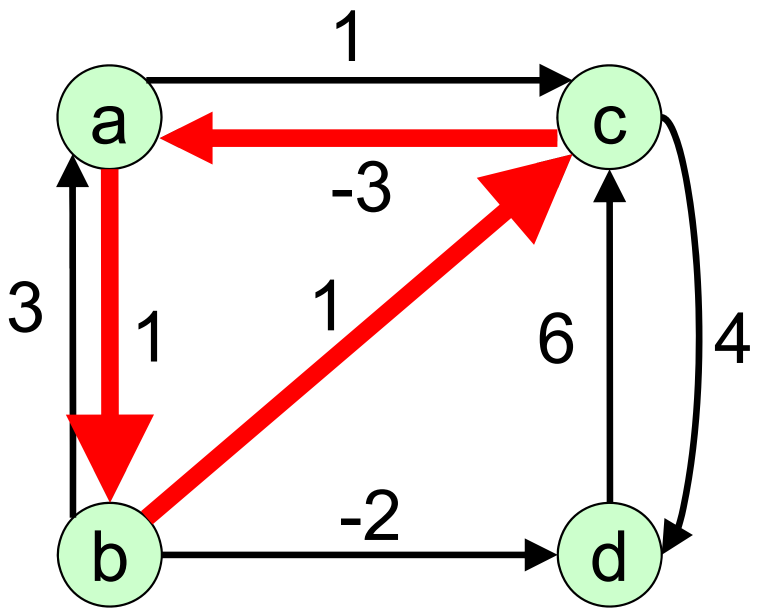
以降で使う仮定 :
s から各頂点 v への路が存在
（存在しない場合は
枝(s,v)を追加，その長さを
十分大きい正数とする）

注意 : 路は，同じ頂点，
同じ枝を何回通っても可

関連する問題：グラフの負閉路の検出

- 入力：有向グラフ G=(V, E)
  各枝の長さ $\ell(e)$ $(e \in E)$
- 出力：グラフに負閉路が「存在する」または「存在しない」の答え
  存在するときは，負閉路を求める

（負閉路 = 閉路のうち，枝の長さの和が負のもの）



応用：通貨両替の問題（鞘取，さやとり）

- 手持ちのお金をうまく両替して，儲けることは可能か？

入力：各国の通貨（JPY, EUR, USD, GBPなど）
　　　通貨の両替レート（1USD=120JPYなど）

出力：手持ちのお金を増やす両替方法は存在するか？

from\to	100JPY	EUR	USD	GBP
100JPY	1	0.76	0.82	0.55
EUR	1.3	1	1.1	0.70
USD	1.2	0.9	1	0.65
GBP	1.8	1.4	1.5	1

100 JPY
→0.76 EUR
→076x1.1 USD
→0.76x1.1x1.2
=100.32 JPY

応用:通貨両替の問題(続き)

グラフを使って表現

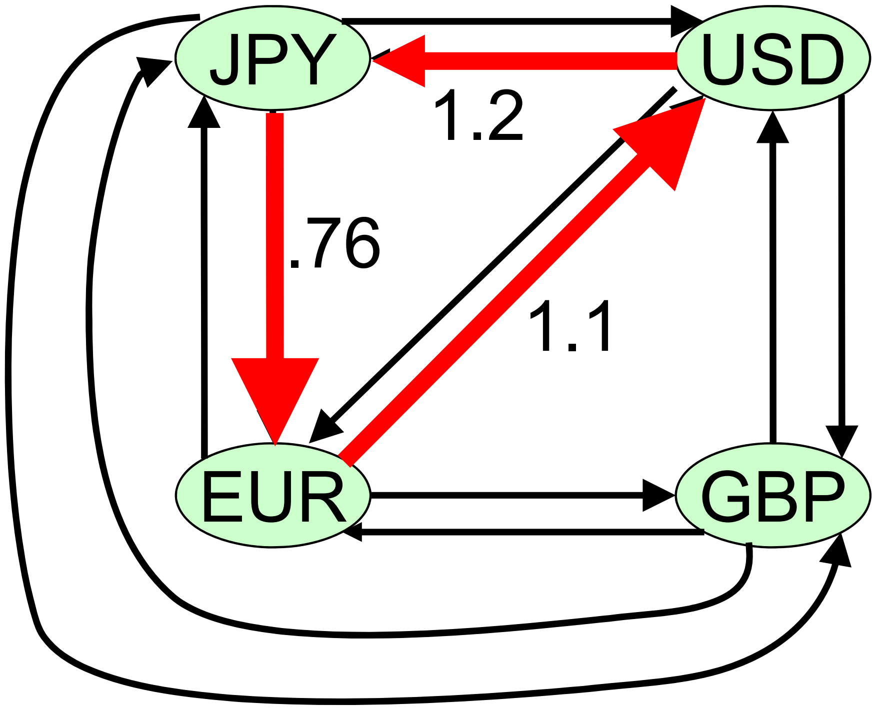
通貨Xから通貨Yへの両替←→枝(X,Y)

両替レート = 枝(X,Y)の重み

➔元の通貨に戻る両替の組合せ←→閉路

金額の変化率 = 閉路に含まれる枝の重みの積

この値>1🡸🡺金額増加



100 JPY
→0.76 EUR
→076x1.1 USD
→0.76x1.1x1.2
=100.32 JPY

閉路の重みを長さに変換して,
「重みの積>1🡸🡺長さの和が負」が成り立つようにする
(ヒント: log ab
=log a + log b)

関連する問題：差分不等式系

- 入力：n 個の変数 $p_1, p_2, \ldots, p_n$ からなる，次の形の不等式系

$$p_i - p_j \le \alpha_{ij}, \quad \beta_i \le p_i \le \gamma_i$$

- 出力：不等式系に解が「ある」または「ない」の答え
  存在するときは，解を求める

具体例

$p_2 - p_1 \le 1,\ p_3 - p_1 \le 1,$
$p_3 - p_2 \le 3,\ p_4 - p_2 \le 5,$
$p_4 - p_3 \le 4,\ p_3 - p_4 \le 6,$
$p_1 = 0,\ p_4 \le 5$

$\Rightarrow$

解あり
$(p_1, p_2, p_3, p_4) = (0,1,1,5)$

応用：プロジェクトスケジューリング(PERT)

- 幾つかの作業からなるプロジェクト（例：自動車製造）
- 各作業の開始時間をどのように設定すべきか？

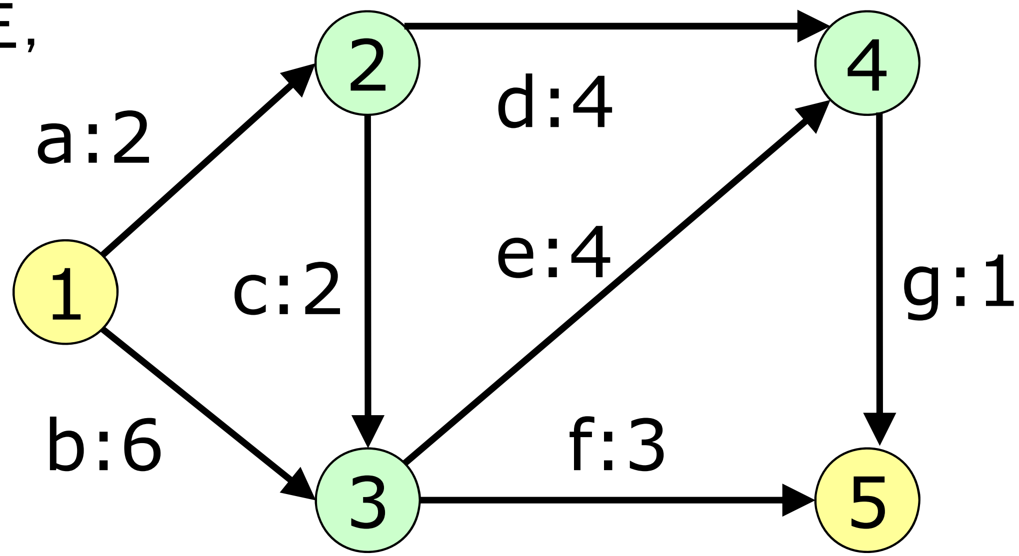
具体例：7つの作業a,b,...,g （グラフの枝に対応）

5つの「イベント」1,2,3,4,5（グラフの頂点に対応）

- イベント1から開始，イベント5で終了
- 作業aは2時間が必要，作業bは6時間，．．．
- 作業a が終わるとイベント2が発生，
  作業c,dの開始可能

作業b, c の両方が終わると
イベント3が発生，
作業e,fの開始可能

注意
このグラフには
閉路が存在
しない



プロジェクトスケジューリング(つづき)

実現可能なスケジュールを立てたい

➔ 各イベント v の発生時間を t(v) （変数）とおき,

差分不等式で表現

例：イベント1はプロジェクト開始 0時間後に発生 ➔ $t(1)=0$

イベント2はイベント1が発生してから少なくとも2時間後に発生

➔ $t(2) - t(1) \geqq 2$

イベント4はイベント2が発生してから少なくとも4時間後に発生

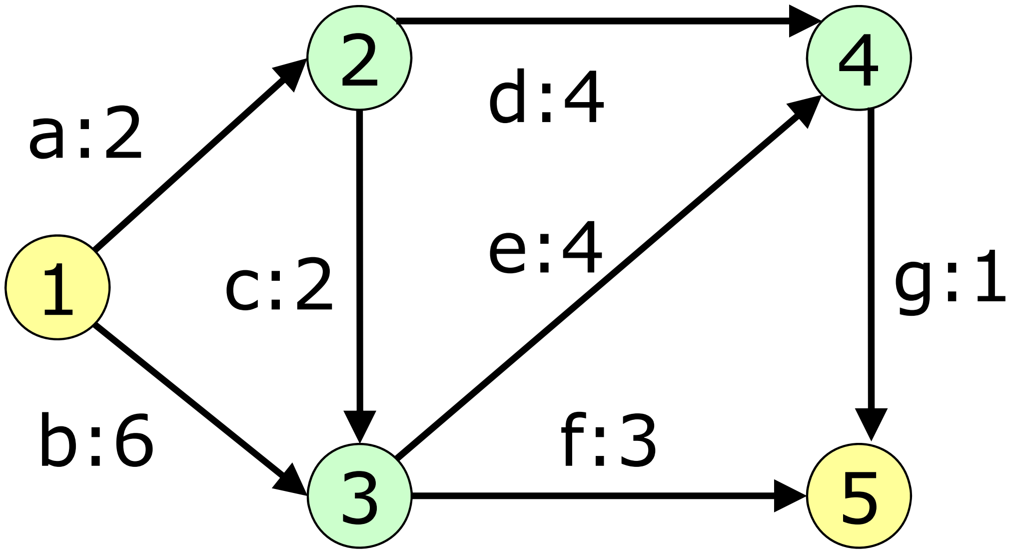
➔ $t(4) - t(2) \geqq 4$

同様にして,

$t(3)-t(1) \geqq 6$, $t(3)-t(2) \geqq 2$,

$t(4)-t(3) \geqq 4$, $t(5)-t(3) \geqq 3$,

$t(5)-t(4) \geqq 1$

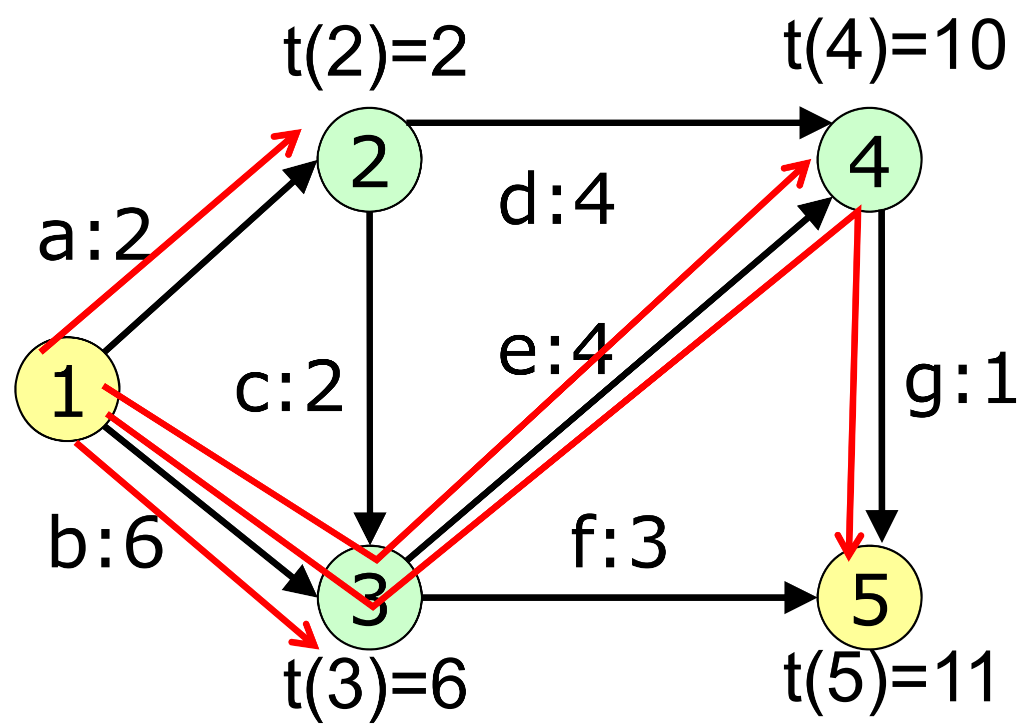


プロジェクトスケジューリング(つづき)

全ての作業を最も早く終えるには，各作業をいつ開始すればよいか？

➔ 各作業の所要時間を枝長と見なして，
開始イベント1から各イベント v への最長路を計算，
その長さを t(v) とおく.

各頂点への最長路の計算方法：
元のグラフの最長路
＝ 枝長の負号を反転したときの
最短路
という事実(＋α)を使えば可能



最短路の計算

ベルマン・フォードのアルゴリズム

(R. Bellman(1958), L. Ford, Jr.(1956))

- 始点 s から各頂点 v への
  最短路長 d(v)（および最短路 P(v)）を求める，　または
- 負閉路が存在することを検出する

アルゴリズムのアイディア：各反復で，次の値を計算

$d_k(v)$ = s から v への，枝数が k 以下の路の長さの最小値

このような路を場合分け：

(1) 枝数≦ k-1 ➔ $d_k(v) = d_{k-1}(v)$

(2) 枝数＝ k ➔ ひとつ手前の頂点を u とおくと，

$$d_k(v) = d_{k-1}(u) + \ell(u, v)$$

∴ 次の再帰式が成立

$$d_k(v) = \min[d_{k-1}(v),\ \ \min\{d_{k-1}(u) + \ell(u, v) | (u, v) \in E\}]$$

ベルマン・フォードのアルゴリズム

再帰式に基づき，$d_k(v)$ を計算するアルゴリズム
n = |V| （=頂点の総数）とおく

手順0： $d_0(s) = 0, d_0(v) = +\infty$ ($\forall v \neq s$) とおく．k=1とする．

手順1： 各頂点 v に対し，以下の式で $d_k(v)$ を計算

$$d_k(v) = \min[d_{k-1}(v),\ \min\{d_{k-1}(u) + \ell(u,v) | (u,v) \in E\}]$$

手順2： k < n ならば k:=k+1とおいて手順1へ戻る．

k = n ならば手順3へ．

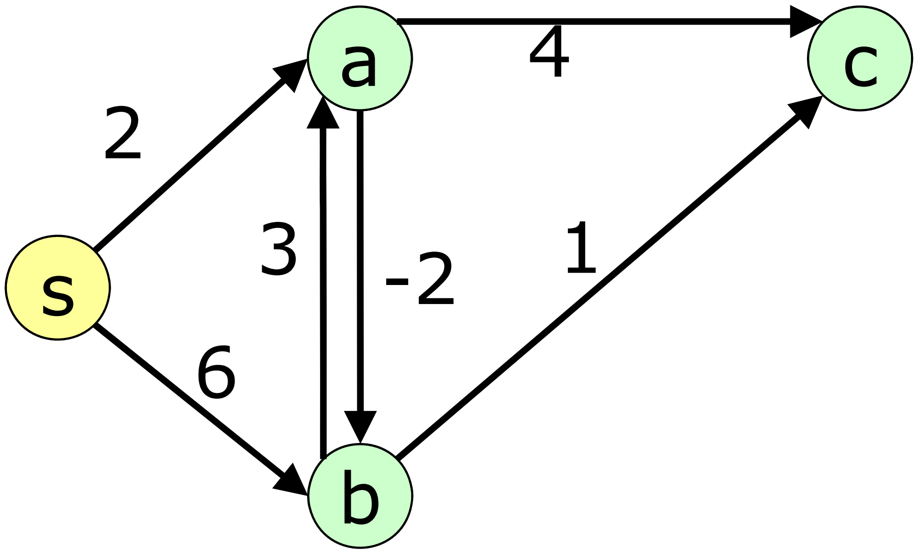
手順3：ある v に対して $d_n(v) < d_{n-1}(v)$が成立➔「負閉路存在」

全ての v に対して $d_n(v) = d_{n-1}(v)$が成立

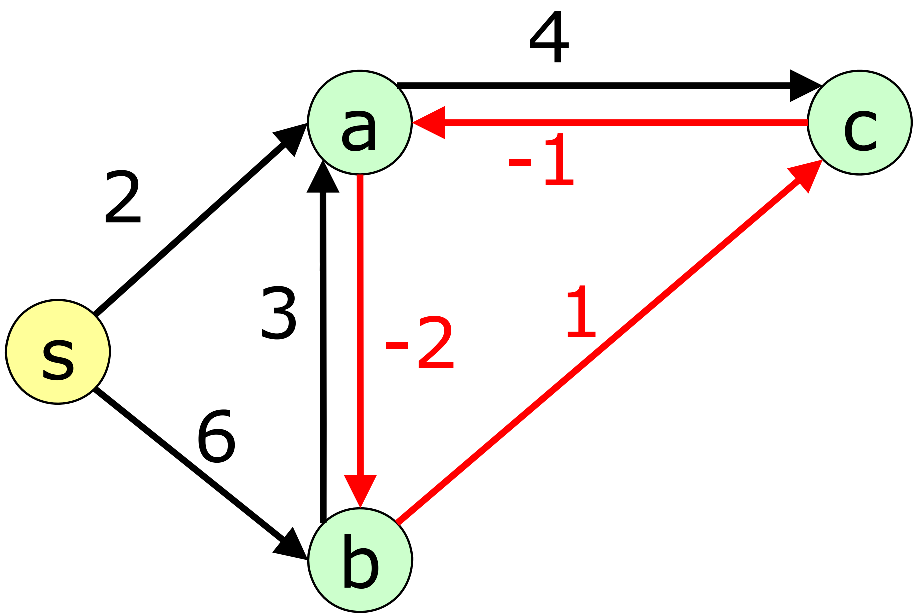
➔最短路長 $d_{n-1}(v)$ を出力

最短路長だけでなく，最短路を計算することも
（若干の修正により）可能

実行例



	k=0	1	2	3	4
s	0	0	0	0	0
a	∞	2	2	2	2
b	∞	6	0	0	0
c	∞	∞	6	1	1



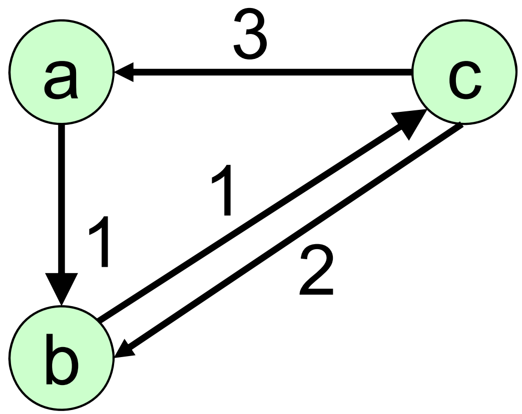
	k=0	1	2	3	4
s	0	0	0	0	0
a	∞	2	2	2	0
b	∞	6	0	0	0
c	∞	∞	6	1	1

最短路の性質

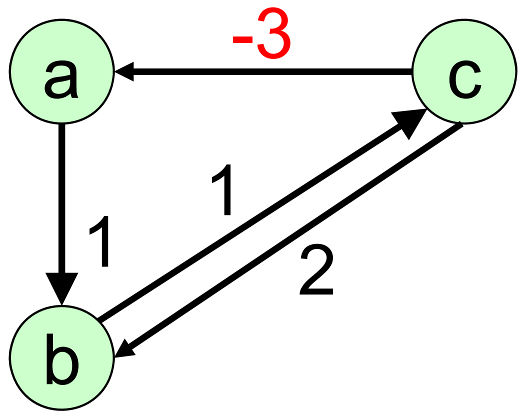
ポテンシャル

最短路問題における便利な道具

定義: 実数 $p(v)\,(v \in V)$ はポテンシャル
⟷ 各枝(u,v)に対し $p(v) - p(u) \le \ell(u,v)$ を満たす



$p(a) = 3, p(b) = 2, p(c) = 0$ はポテンシャル



※ポテンシャルは存在しないこともある
左のグラフにおいて,

$$p(a) - p(c) \le -3, p(c) - p(b) \le 1,$$
$$p(b) - p(a) \le 1$$

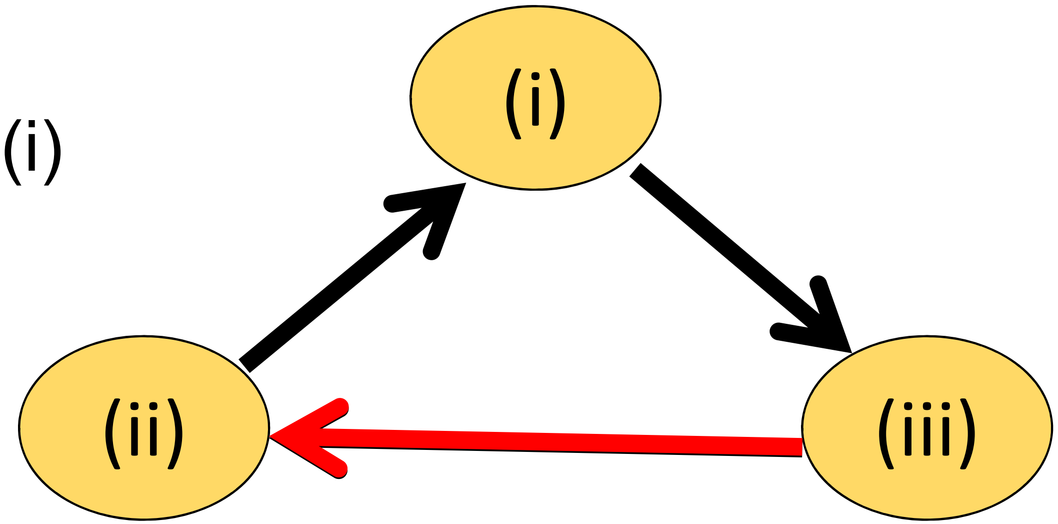
は解をもたない

最短路と負閉路とポテンシャル

与えられたグラフにおける最短路，負閉路，ポテンシャルの存在に関して，以下の関係が成り立つ，

定理：次の (i), (ii), (iii) は等価
(i) 始点 s から各頂点への最短路が存在
(ii) 負閉路が存在しない
(iii) ポテンシャルが存在

これからの流れ：
(iii)➔(ii), (i)➔(iii), (ii)➔(i)
を順番に証明



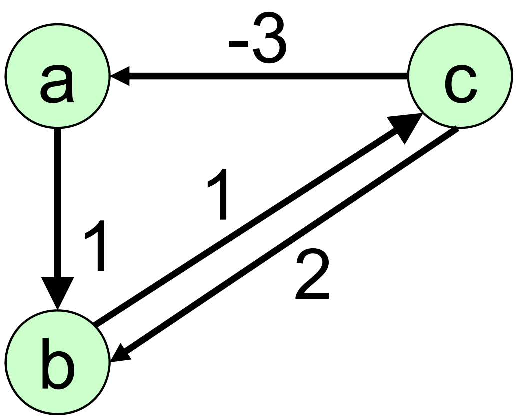
ポテンシャル存在➔負閉路が非存在

定義: 実数 $p(v)(v \in V)$ はポテンシャル
⟵➔各枝(u,v)に対し $p(v) - p(u) \le \ell(u,v)$ を満たす

命題 ポテンシャルが存在➔負閉路は存在しない
(対偶:負閉路が存在➔ポテンシャルは存在しない)

(証明) 任意の閉路C の長さが非負であることを示せば良い.
不等式 $p(v) - p(u) \le \ell(u,v)$ $((u,v) \in C)$ を辺々足す

➔ $0 = \sum_{(u,v)\in C}[p(v) - p(u)]$
$\le \sum_{(u,v)\in C} \ell(u,v)$ =閉路の長さ ■



左のグラフにおいて,
$p(a) - p(c) \le -3, p(c) - p(b) \le 1,$
$p(b) - p(a) \le 1$
は解をもたない

最短路と負閉路とポテンシャル

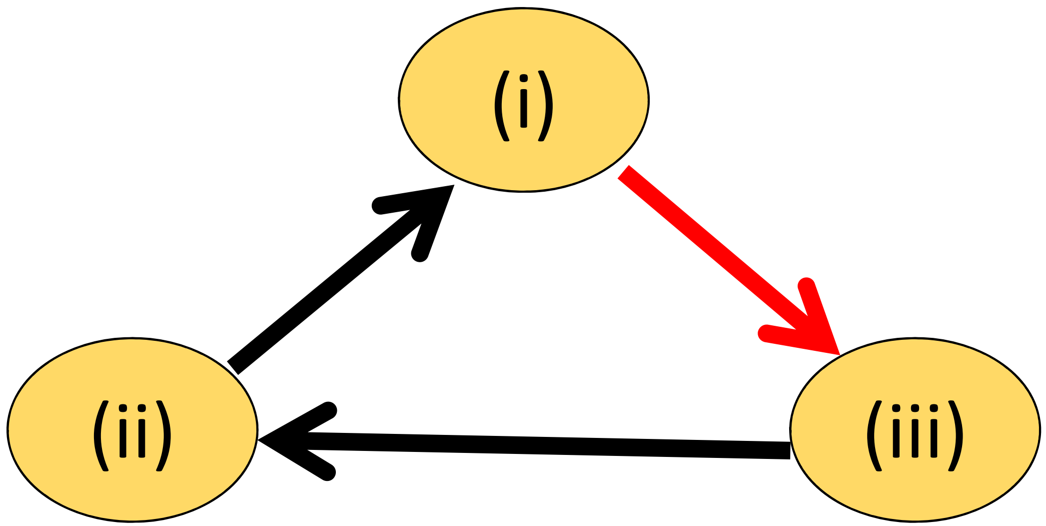
与えられたグラフにおける最短路，負閉路，ポテンシャルの存在に関して，以下の関係が成り立つ，

定理：次の (i), (ii), (iii) は等価

(i) 始点 s から各頂点への最短路が存在

(ii) 負閉路が存在しない

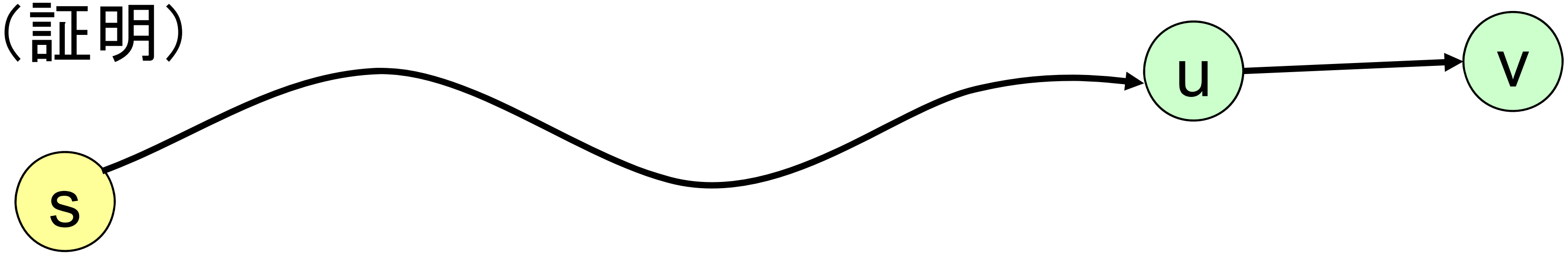
(iii) ポテンシャルが存在



最短路存在➔ポテンシャル存在

命題 頂点sから各頂点への最短路が存在すると仮定.
このとき，ポテンシャルが存在する.
とくに，頂点 v への最短路長を d(v) とすると,
p(v)=d(v) はポテンシャル

（証明）



P: 頂点 s から u への最短路 → d(u)= P の長さ
➔ $\tilde{P} = P \cup \{(u,v)\}$ は s から v への路,
その長さ$= d(u) + \ell(u,v)$
$\geq$ v への最短路長=d(v) ■

最短路と負閉路とポテンシャル

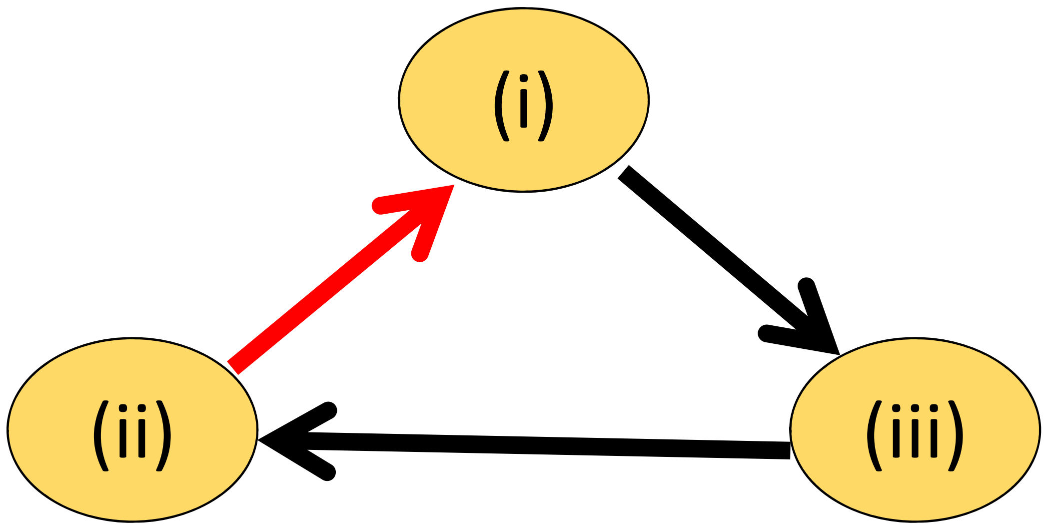
与えられたグラフにおける最短路，負閉路，ポテンシャルの存在に関して，以下の関係が成り立つ，

定理：次の (i), (ii), (iii) は等価

(i) 始点 s から各頂点への最短路が存在

(ii) 負閉路が存在しない

(iii) ポテンシャルが存在



負閉路が不存在➔最短路が存在

定理 : 有向グラフに負閉路が存在しない
➔ 始点から各頂点 v へ，枝数≦|V|-1 の最短路が存在

（証明）以下では n = |V| とおく.
P*: s から v への，枝数 n-1 以下の路の中で最短（←必ず存在）

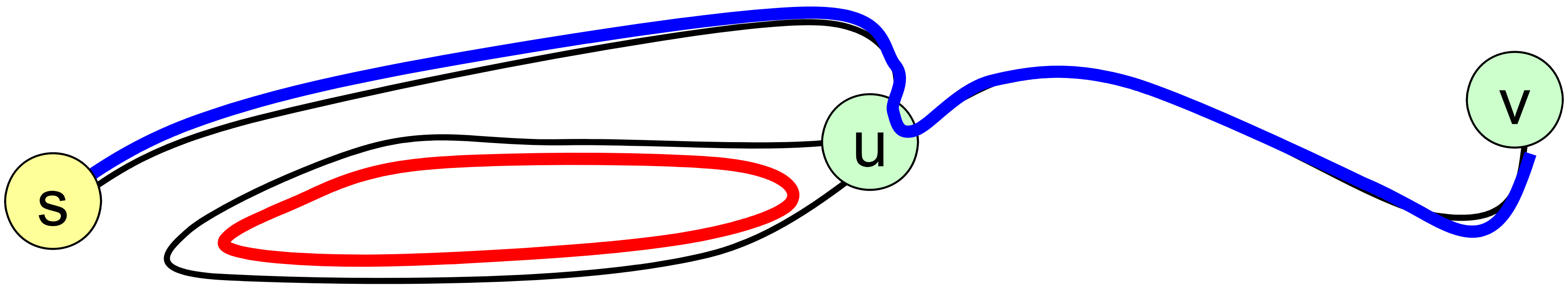
次の命題を示せば良い.
s から v への，枝数が n 以上の任意の路 P に対し $\ell(P^*) \le \ell(P)$

背理法 :
枝数が n 以上のある s から v への路 P に対し
$\ell(P^*) > \ell(P)$ が成り立つと仮定.
そのような路が複数存在したら，枝数が最も少ないものを P とする.

負閉路が不存在➔最短路が存在（つづき）

定理：有向グラフに負閉路が存在しない
➔ 始点から各頂点 v へ，枝数≦|V|-1 の最短路が存在

（証明のつづき）
P は s から v への路，枝数≧n
➔ ある頂点が必ず2回現れる（u とする）



u から u への部分路は閉路C ➔長さ $\ell(C)$ は非負
閉路を削除して得られる路を P’ とおく
➔ P’ は s から v への路，長さ $\ell(P') = \ell(P) - \ell(C) < \ell(P) < \ell(P^*)$
P’ の枝数 < P の枝数　（矛盾）　■

最短路の計算：正当性

ベルマン･フォードのアルゴリズム

n = |V|とおく

手順0: $d_0(s) = 0, d_0(v) = +\infty$ $(\forall v \neq s)$ とおく．k=1とする.

- s から枝数0でたどり着けるのは s のみなので

手順1: 各頂点 v に対し，以下の式で $d_k(v)$ を計算

$$d_k(v) = \min[d_{k-1}(v),\ \min\{d_{k-1}(u) + \ell(u,v) | (u,v) \in E\}]$$

手順2: k < n ならば k:=k+1とおいて手順1へ戻る.

k=n ならば手順3へ.

手順3: ある v に対して $d_n(v) < d_{n-1}(v)$が成立➔「負閉路存在」

全ての v に対して $d_n(v) = d_{n-1}(v)$が成立

➔最短路長 $d_{n-1}(v)$ を出力

出力結果が正しいことを証明

出力結果が正しいことを証明

アルゴリズムの正当性(その1)

再帰式　$d_k(v) = \min[d_{k-1}(v),\ \min\{d_{k-1}(u) + \ell(u,v) | (u,v) \in E\}]$

命題：負閉路が存在する ➔ 負閉路に含まれるある頂点 v に対して $d_n(v) < d_{n-1}(v)$ が成立

（証明）閉路Cに含まれる枝を $(v_1, v_2), (v_2, v_3), \dots, (v_k, v_1)$ とおく.

C は負閉路なので, $\ell(v_1, v_2) + \ell(v_2, v_3) + \cdots + \ell(v_k, v_1) < 0.$

再帰式より $d_n(v_{j+1}) \le d_{n-1}(v_j) + \ell(v_j, v_{j+1})$ $(\forall j = 1,2,\dots,k)$

（ただし, $v_{k+1} = v_1$ とする）

不等式を辺々足すと,

$$\sum_{j=1}^{k} d_n(v_j) = \sum_{j=1}^{k} d_n(v_{j+1})$$
$$\le \sum_{j=1}^{k} d_{n-1}(v_j) + \sum_{j=1}^{k} \ell(v_j, v_{j+1}) < \sum_{j=1}^{k} d_{n-1}(v_j)$$

∴ ある v_j に対して $d_n(v_j) < d_{n-1}(v_j)$ が成立　■

アルゴリズムの正当性(その2)

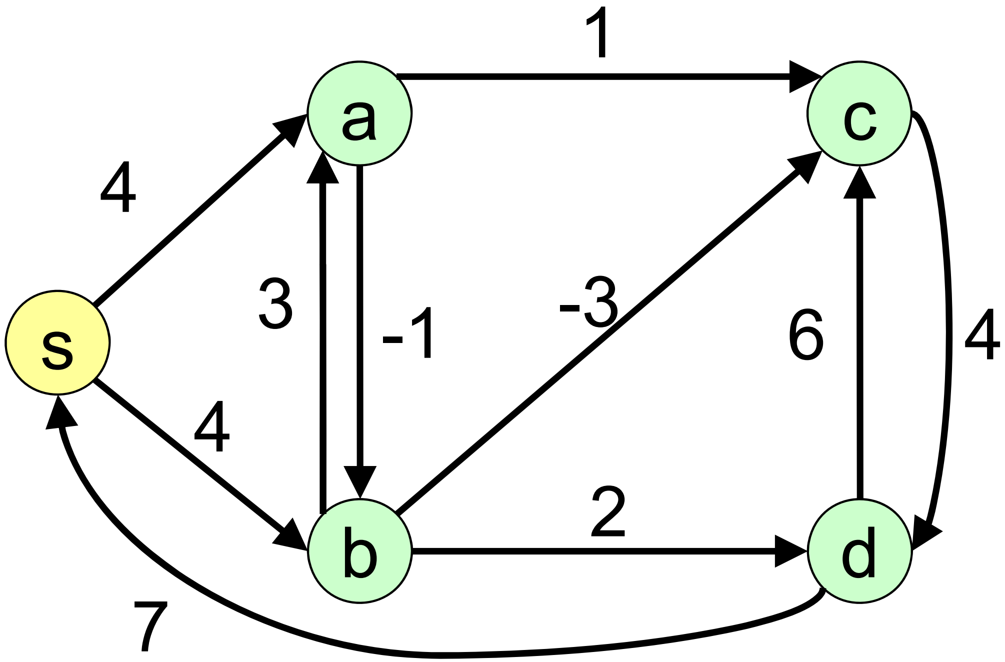
命題:負閉路が存在しない ➔任意の頂点 vに対して
$d_{n-1}(v)$ は頂点 v への最短路長に等しく, $d_n(v) = d_{n-1}(v)$ 成立

(証明) 各頂点 v への, 枝数≦ n-1 の最短路が存在(定理より)
➔ $d_{n-1}(v)$ は頂点 v への最短路長に等しい
さらに, 全ての v に対して $d_n(v) = d_{n-1}(v)$ が成立 ■

定理:有向グラフに負閉路が存在しない
➔ 始点から各頂点へ, 枝数≦|V|-1 の最短路が存在

レポート問題

問1：下記のグラフにおけるs から各頂点への最短路長を，ベルマン・フォードのアルゴリズムで計算せよ．



	k=0	1	2	3	4	5
s	0					
a	+∞					
b	+∞					
c	+∞					
d	+∞					

問2： ベルマン・フォードのアルゴリズムにおいて，最短路を計算するにはどのように修正したら良いか，説明せよ．