

数理経済学特講

複数財に対するオークションの 数理とアルゴリズム

第4回 二部グラフのマッチング問題

塩浦昭義

東京工業大学 経営工学系 准教授

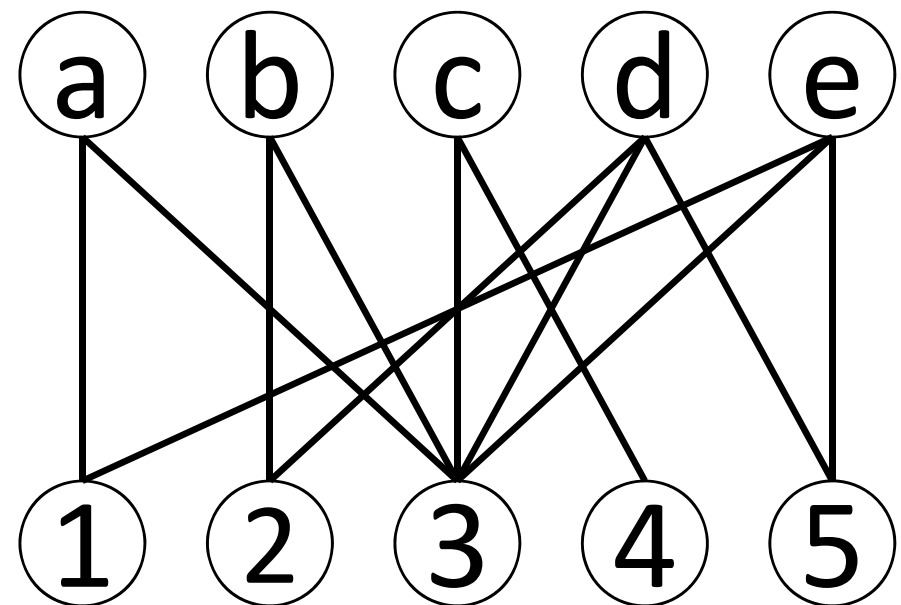
shioura.a.aa@m.titech.ac.jp

二部グラフのマッチング問題

- (グラフ理論における) マッチング問題:
「ひと」や「もの」の間の最適なペアの集合を求める問題
- 例: 入札者への財の割り当て
 - 5人の入札者 $B=\{a, b, c, d, e\}$ 5つの財 $N=\{1, 2, 3, 4, 5\}$
 - 各々の入札者は欲しい財と欲しくない財がある
→ グラフで表現可能

頂点の2つのグループ
B と N の間にのみ, 枝が存在
← **二部グラフ**と呼ばれる

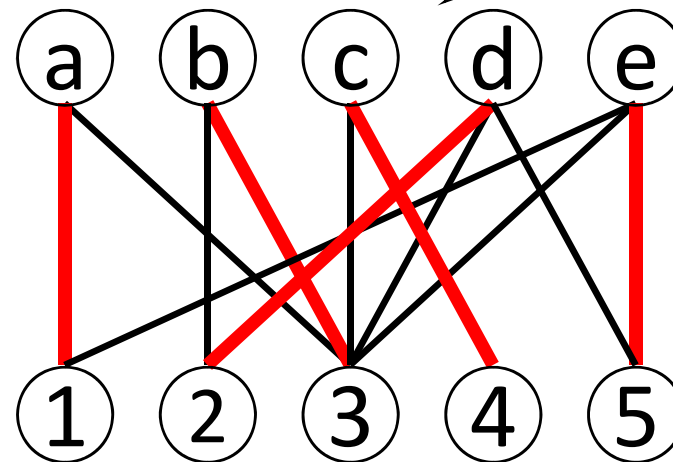
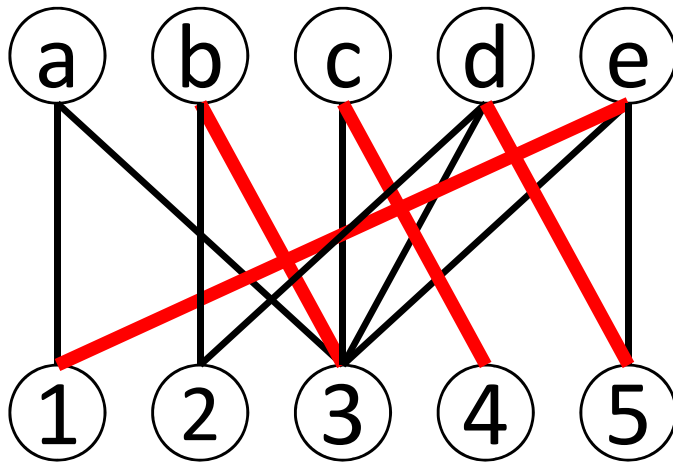
※2部グラフにおいて, B と N の
頂点数は同じでなくても良い



二部グラフの最大マッチング問題

- できるだけ多くの入札者に財を割り当てたい
 - 各入札者には, 高々1つの財が割り当て可能
 - 各財は, 高々一人の入札者に割り当て可能
- ➔ 財の割当を枝集合で表現可能

最大
マッチング



- (グラフの) マッチング:
グラフの枝集合であって, 各頂点に接続する枝の数(次数)が1以下
- 最大(サイズ)マッチング問題:
与えられたグラフのマッチングの中で, 枝数最大のものを求める

応用：均衡価格の判定(その1)

- 複数不可分財のオークションにおいて,
与えられた財の価格が均衡価格か否かを判定

定義：財の価格 $p(j) \in \mathbb{R}_+$ は 均衡価格

↔ 以下の条件を満たす財の配分 $\alpha(i) \in N \cup \{0\}$ が存在
ただし $q(i) \equiv \max_{1 \leq h \leq n} \{v(i, h) - p(h)\}$

① 入札者 i に財 j が割り当てられる ($\alpha(i) = j$)

→ $0 \leq v(i, j) - p(j) = q(i)$

② 入札者 i に財の割り当てがない ($\alpha(i) = 0$) → $q(i) \leq 0$

③ 財 j が誰にも割り当てられない → $p(j) = 0$

- 条件②, ③は対偶をとるとわかりやすい

②' 入札者 i に財の割り当てがある ($\alpha(i) \neq 0$) ← $q(i) > 0$

③' 財 j は誰かに割り当てられる ← $p(j) > 0$

応用：均衡価格の判定(その2)

マッチングを用いて、均衡価格の定義の書き換えが可能

定義：財の価格 $p(j) \in \mathbb{R}_+$ は **均衡価格**

↔ 頂点集合 $B \cup N$, 枝集合 E の二部グラフにおいて,
頂点集合 $B' \cup N'$ をカバーするマッチングが存在

$$B' \equiv \{i \in B \mid q(i) > 0\}, \quad N' = \{j \in N \mid p(j) > 0\},$$

$$E = \{(i, j) \mid i \in B, j \in N, 0 \leq v(i, j) - p(j) = q(i)\}$$

① 入札者 i に財 j が割り当てられる ($\alpha(i) = j$)

$$\rightarrow 0 \leq v(i, j) - p(j) = q(i)$$

②' 入札者 i に財の割り当てがある ($\alpha(i) \neq 0$) ← $q(i) > 0$

③' 財 j は誰かに割り当てられる ← $p(j) > 0$

マッチングの枝は
Eに含まれる

マッチングは
N'をカバーする

マッチングは
B'をカバーする

応用：均衡価格の判定（その3）

- B' , N' を同時にカバーするマッチングが存在
 $\longleftrightarrow$ B' をカバーするマッチングが存在
 & N' をカバーするマッチングが存在（証明は後日）
 - B' をカバーするマッチングの存在判定は
 最大マッチング問題に帰着可能
- (1) 二部グラフ($B, N; E$) から, $B-B'$ に含まれる頂点
 およびそれらに接続する枝を削除
 - (2) 得られたグラフの最大マッチング M を求める
 - (3) $|M| = |B'|$ ならば B' はカバー可能.
 $|M| < |B'|$ ならば B' はカバー不可能.

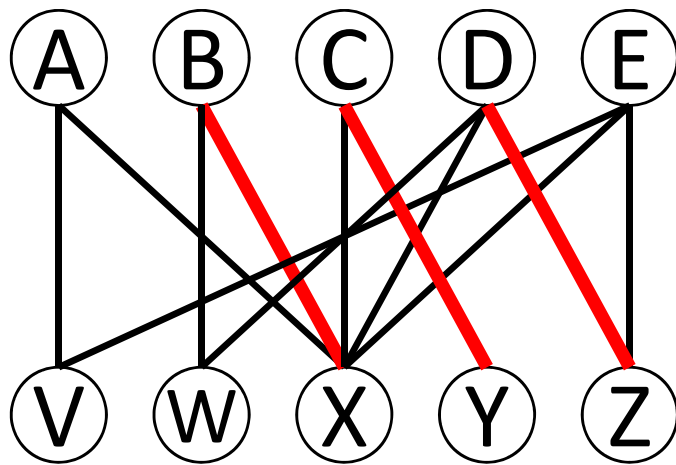
マッチングと交互路

マッチングに関する交互路

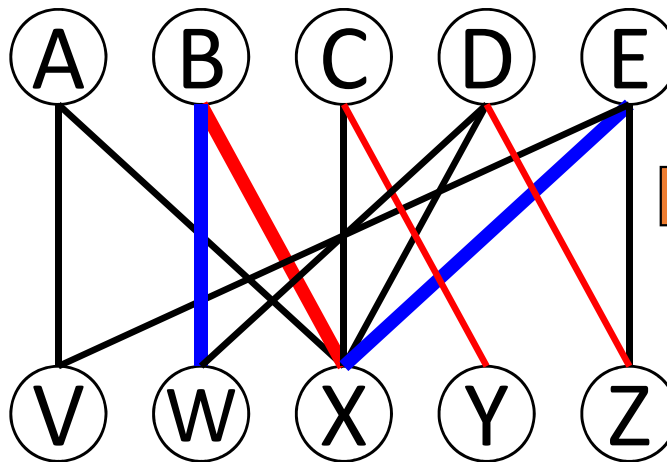
交互路 – 新しいマッチングを作るための「道具」

定義: マッチング M に関する**交互路**

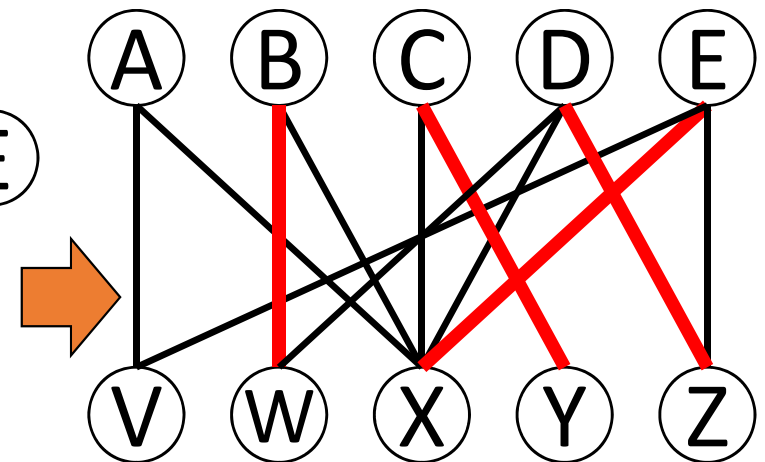
- (i) M の枝と M 以外の枝が交互に現れる路
- (ii) 路の最初の枝が M 以外 $\rightarrow$ 路の始点は, M の枝の端点ではない
- (iii) 路の最後の枝が M 以外 $\rightarrow$ 路の終点は, M の枝の端点ではない



マッチング M



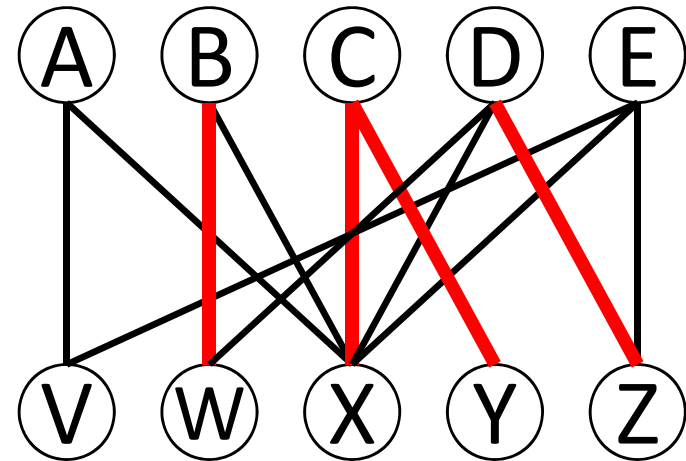
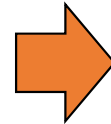
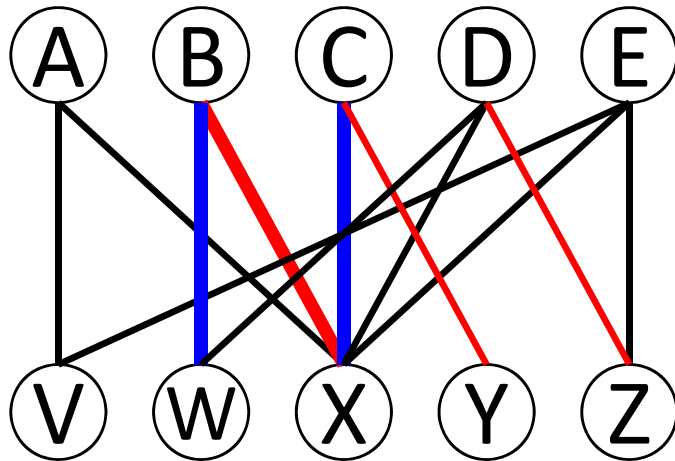
交互路



交互路を使って,
マッチングの枝を
入れ替えした結果

マッチングに関する交互路

交互路でない例

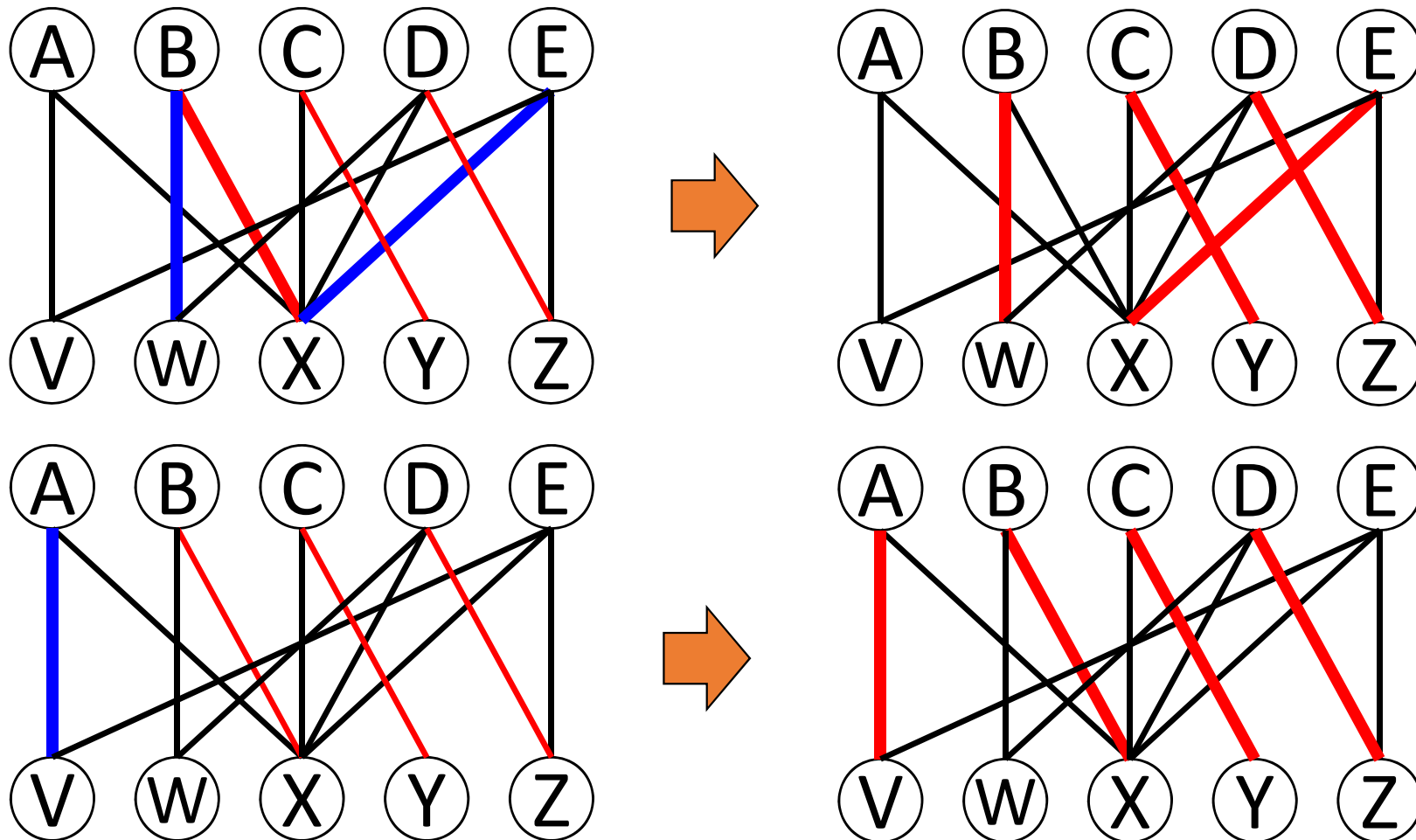


条件(ii)を満たさない
 ➡ 枝を入れ替えても
 マッチングにならない

マッチングに関する交互路

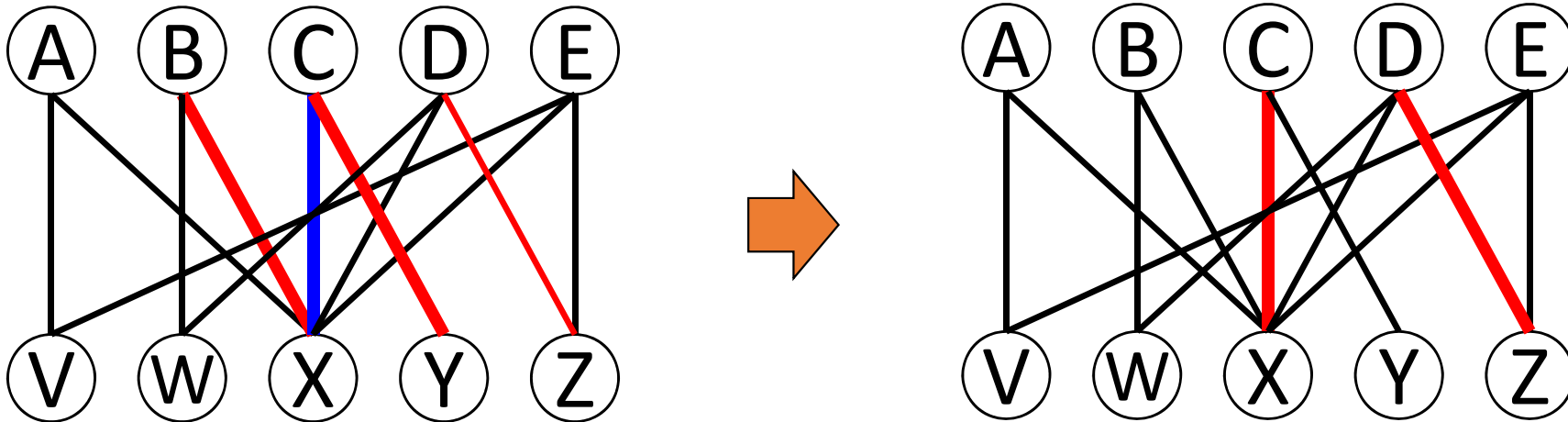
交互路タイプ1: M 以外の枝数 = M の枝数 + 1
 (枝を入れ替えるとマッチングの枝数が1増える)

→ 増加路と呼ばれる

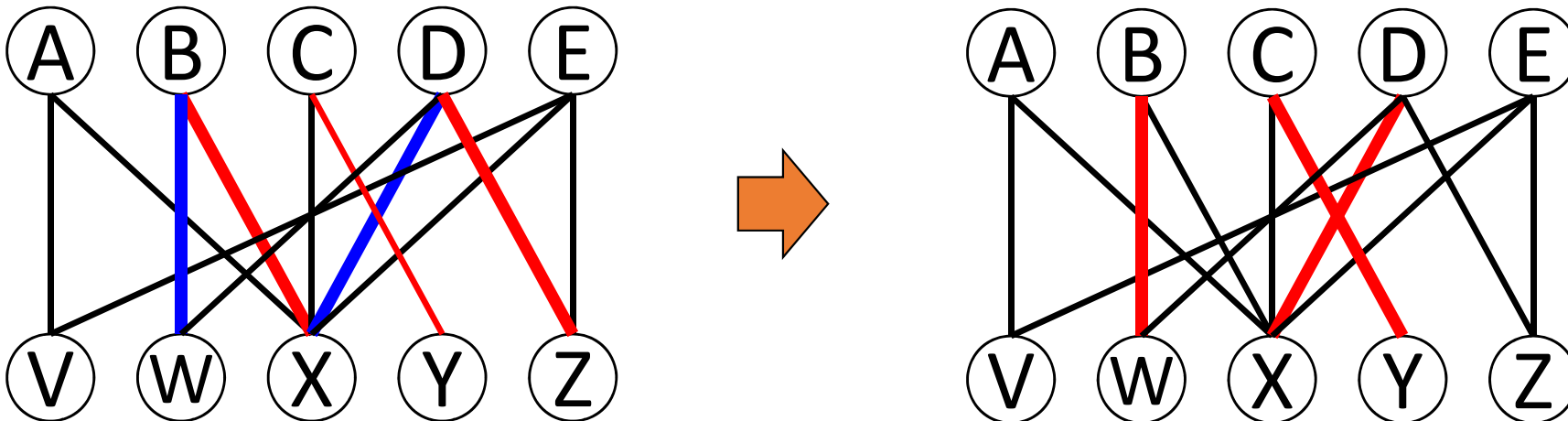


マッチングに関する交互路

交互路タイプ2: M 以外の枝数 = M の枝数 - 1
 (枝を入れ替えるとマッチングの枝数が1減る「減少路」)



交互路タイプ3: M 以外の枝数 = M の枝数
 (枝を入れ替えても、マッチングの枝数是不変)



マッチングに関する交互閉路

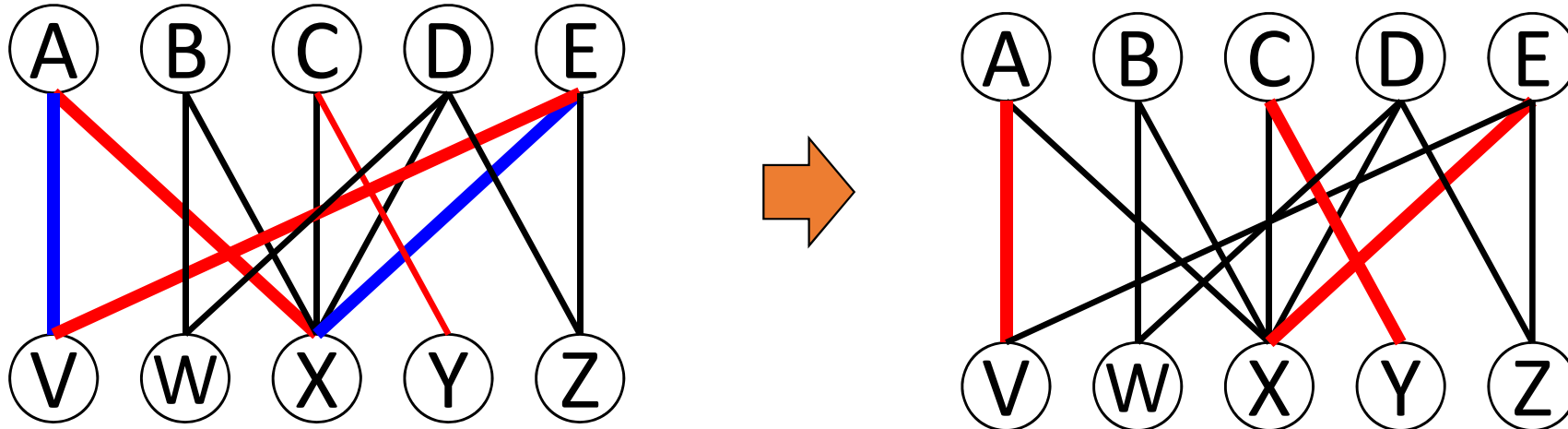
定義: マッチング M に関する **交互閉路**

= M の枝と M 以外の枝が交互に現れる閉路

交互閉路では、「 M 以外の枝数 = M の枝数」

→ 枝を入れ替えても、マッチングの枝数是不変

交互閉路の例



交互路によるマッチングの更新

命題 マッチング M に対し,
 路 P は (M に関する) 交互路
 $\longleftrightarrow P$ を使って M の枝を入れ替えて得られる
 枝集合 $M' = (M \setminus P) \cup (P \setminus M)$ はマッチング

命題 マッチング M に対し,
 M に関する **増加路** が存在 $\rightarrow M$ は最大マッチングではない
 (対偶: M は最大マッチング $\rightarrow M$ に関する **増加路** が存在しない)

逆も成立

定理 マッチング M に対し,
 M に関する **増加路** が存在 $\longleftrightarrow M$ は最大マッチングではない
 (M は最大マッチング $\longleftrightarrow M$ に関する **増加路** が存在しない)

増加路なし $\rightarrow$ 最大マッチング の証明

命題 マッチング M に対し,

M に関する **増加路** が存在 $\leftarrow M$ は最大マッチングではない

(対偶: M は最大マッチング $\leftarrow M$ に関する **増加路** が存在しない)

(証明) 最大マッチングを M^* とおく $\rightarrow |M^*| > |M|$

枝集合 $(M^* \setminus M) \cup (M \setminus M^*)$ に注目

$\rightarrow$ 幾つかの (M に関する)

交互路・交互閉路からなる

($\because$ 各頂点には M の枝が高々1つ,

M^* の枝が高々一つ, 接続しているから)

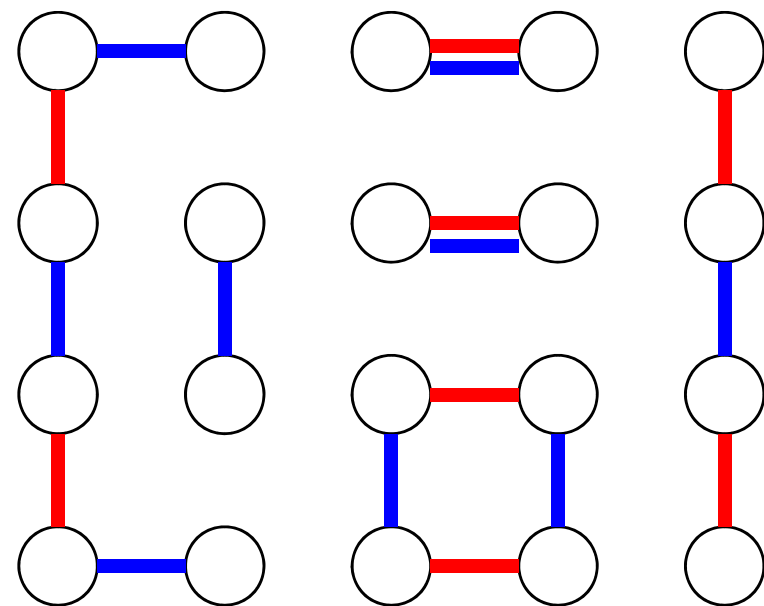
その中に, 増加路が必ず存在

($\because$ タイプ2, 3の交互路, 交互閉路は

いずれも「 M の枝数 $\leq M^*$ の枝数」

$\rightarrow$ 合計すると, $|M^*| \leq |M|$ となり矛盾) ■

※ 減少路は存在しない ($\because M^*$ にとっての増加路に対応するから)



最大マッチングの計算

最大マッチング問題のアルゴリズム

「 M は最大マッチング $\iff M$ に関する増加路が存在しない」
これに基づき、次のアルゴリズムが得られる

ステップ0: 初期マッチングを $M := \emptyset$ とする.

ステップ1: M に関する増加路の存在をチェック.

存在しない $\rightarrow$ 現在の M は最大マッチング(終了)

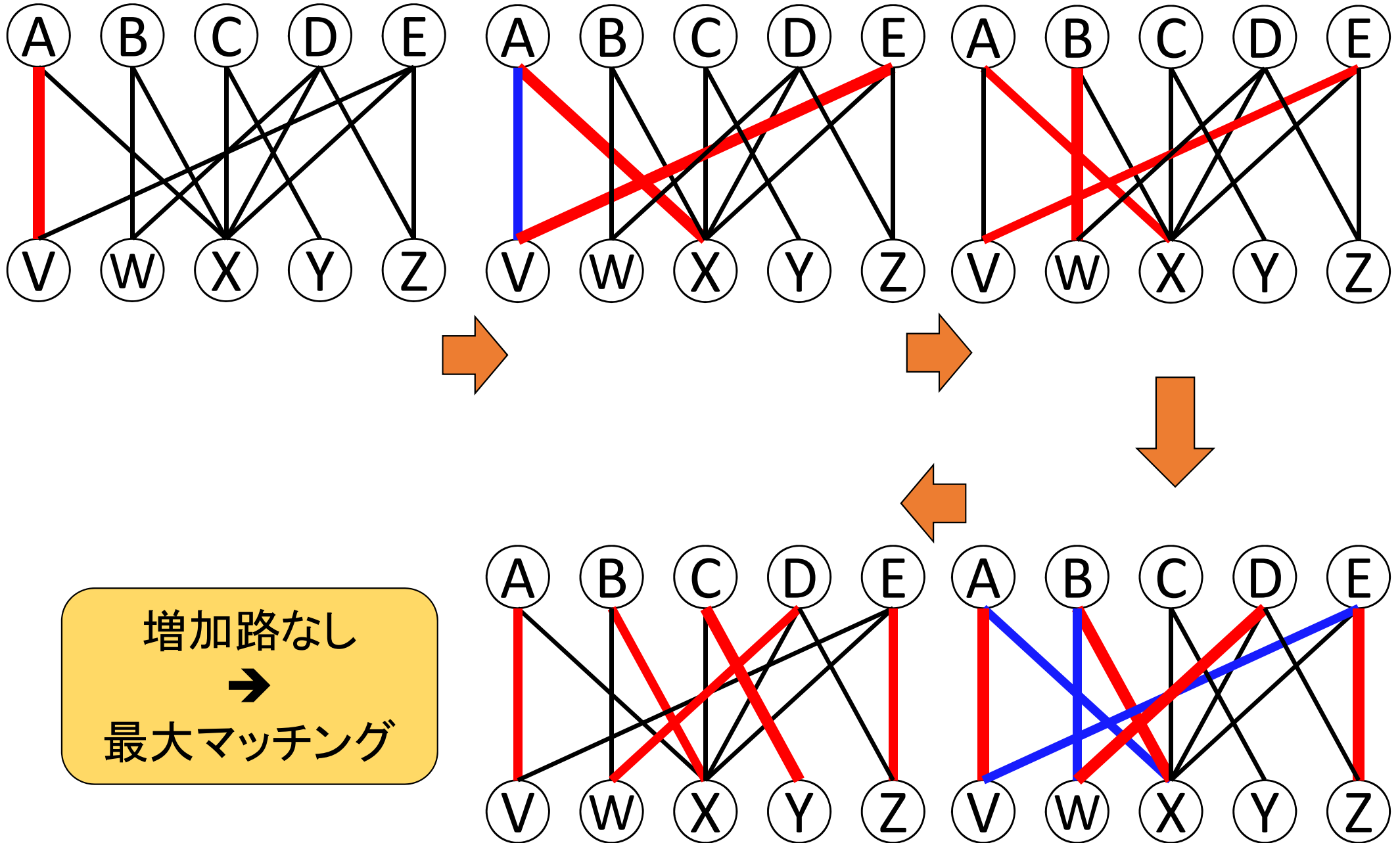
存在する $\rightarrow$ ステップ2へ.

ステップ2: M に関する増加路 P をひとつ見つけ,

P を用いて M を更新 ($M := (M \setminus P) \cup (P \setminus M)$)

ステップ1へ.

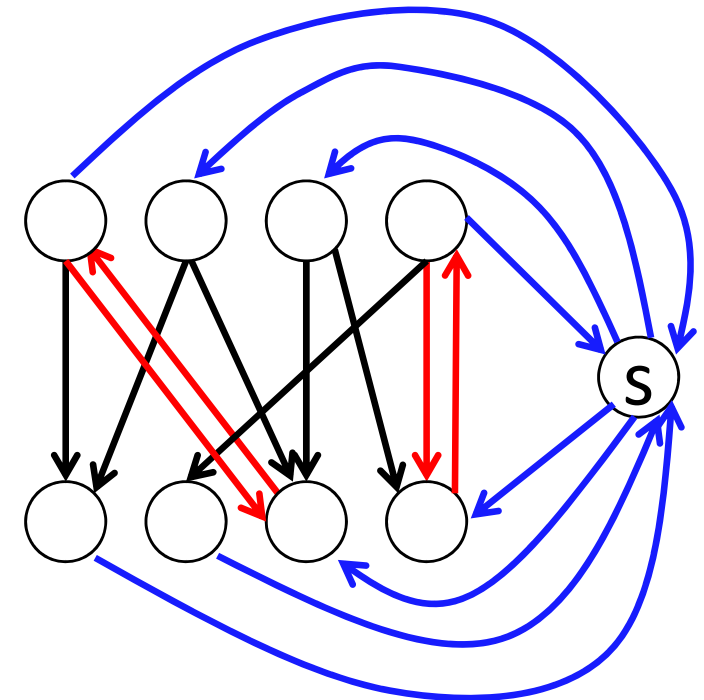
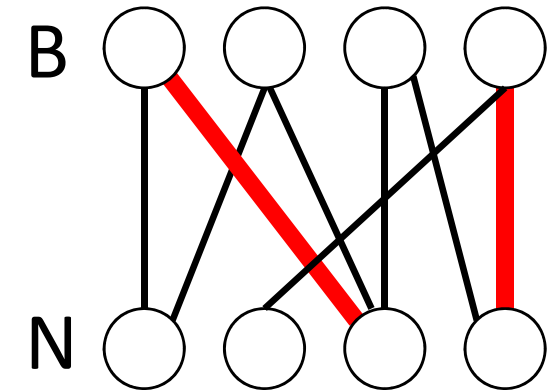
アルゴリズムの実行例



増加路の見つけ方：準備

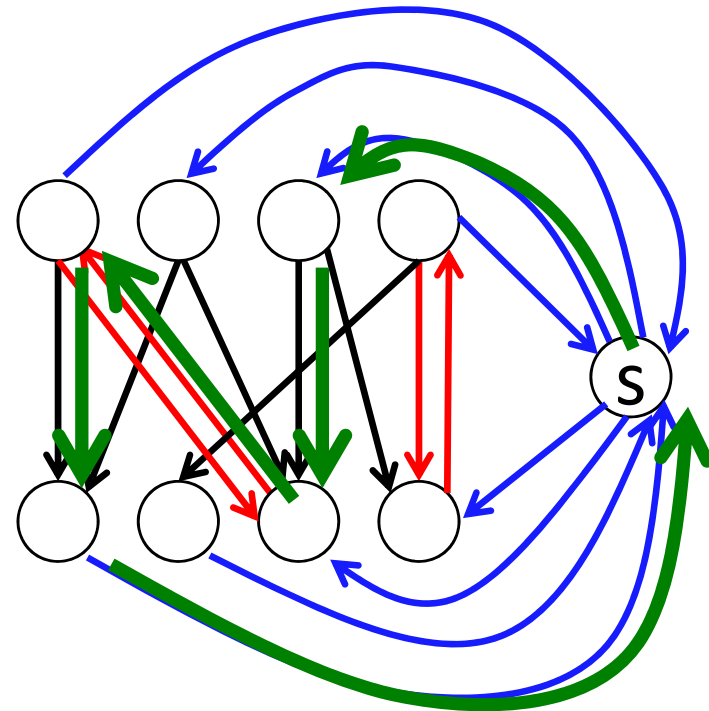
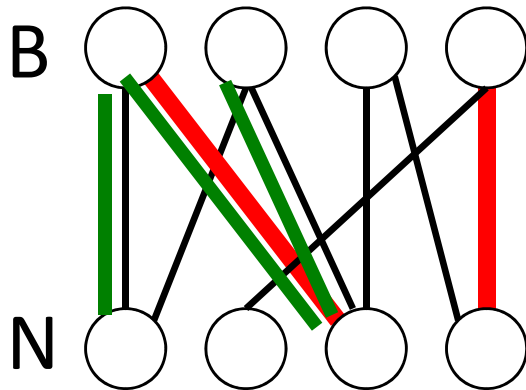
増加路の検出→ある有向グラフでの
有向(閉)路の検出に帰着可能

- 全ての枝を, B 側から N 側に向き付け
- マッチング M の各枝に対し, さらに
逆向きの枝 ($N \rightarrow B$) を追加
- 新たな頂点 s をおく
- B 側の各頂点 u に対し,
 - マッチング枝が接続 $\rightarrow$ 枝 (u, s) をおく
 - 接続なし $\rightarrow (s, u)$ をおく
- N 側の各頂点 v に対し,
 - マッチング枝が接続 $\rightarrow$ 枝 (s, v) をおく
 - 接続なし $\rightarrow (v, s)$ をおく



増加路の見つけ方

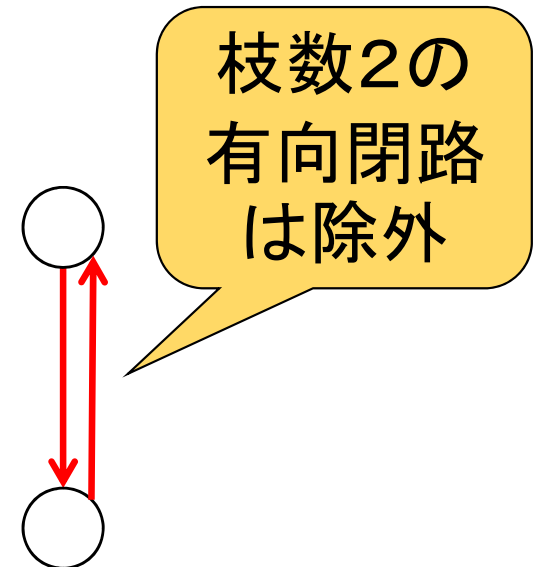
- 新しい有向グラフにおいて,
 頂点 $s \rightarrow B$ 側の頂点 $\rightarrow \dots \rightarrow N$ 側の頂点 $\rightarrow$ 頂点 s の形の有向閉路
 $\longleftrightarrow$ 元の二部グラフの (M に関する) 増加路
 $\therefore$ 有向グラフにおいて, 上記のような有向閉路を求めればよい



有向閉路と交互路の関係

- 新しい有向グラフの有向閉路(のほとんど)は,
元の二部グラフの交互路と対応

- (1) 頂点 $s \rightarrow B$ 側の頂点 $\rightarrow \dots \rightarrow N$ 側の頂点 $\rightarrow$ 頂点 s の形の有向閉路
 $\longleftrightarrow$ 元の二部グラフの増加路(タイプ1)
- (2) 頂点 $s \rightarrow N$ 側の頂点 $\rightarrow \dots \rightarrow B$ 側の頂点 $\rightarrow$ 頂点 s の形の有向閉路
 $\longleftrightarrow$ 元の二部グラフの「減少路」(タイプ2)
- (3) 上記以外で, 頂点 s を通過する有向閉路
 $\longleftrightarrow$ 元の二部グラフのタイプ3の交互路
- (4) 頂点 s を通過しない(枝数4以上の)有向閉路
 $\longleftrightarrow$ 元の二部グラフの交互閉路



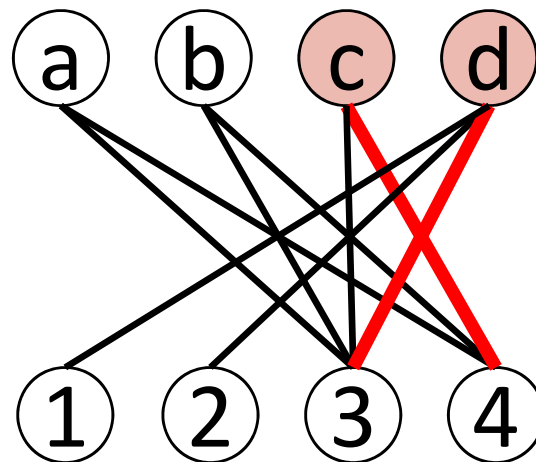
指定頂点をカバーするマッチング

カバーする頂点の制約(片側)

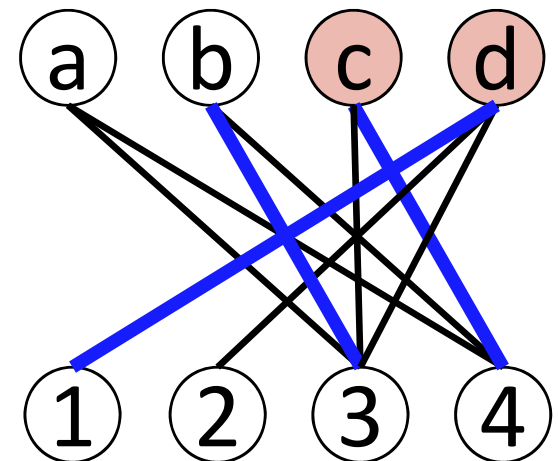
カバーする頂点の制約付きマッチングに関する性質

命題1

- (i) B の部分集合 B' に対し, B' をカバーするマッチングが存在
→ B' をカバーする最大マッチングが存在
- (ii) N の部分集合 N' に対し, N' をカバーするマッチングが存在
→ N' をカバーする最大マッチングが存在



赤いマッチングは
 $B' = \{c, d\}$ をカバーする



青いマッチングは
最大マッチング

カバーする頂点の制約(片側)

命題1

- (i) B の部分集合 B' に対し, B' をカバーするマッチングが存在
→ B' をカバーする最大マッチングが存在
- (ii) N の部分集合 N' に対し, N' をカバーするマッチングが存在
→ N' をカバーする最大マッチングが存在

[証明] M : B' をカバーするマッチング

M に増加路アルゴリズムを適用し, 最大マッチング M^* を求める
増加路アルゴリズムの各反復において,

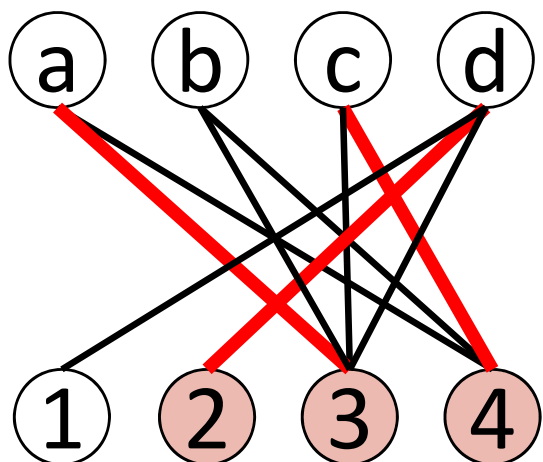
マッチングのカバーする頂点は単調増加(減らない)

∴ M^* のカバーする頂点 $\supseteq M$ のカバーする頂点 $\supseteq B'$ ■

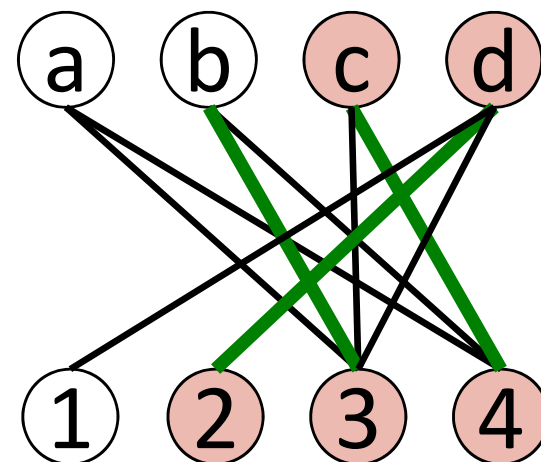
カバーする頂点の制約(両側)

命題2 B の部分集合 B' および N の部分集合 N' に対し,
 B' をカバーするマッチング, および N' をカバーするマッチングが
 それぞれ存在

→ B' と N' を同時にカバーする(最大)マッチングが存在



赤いマッチングは
 $N' = \{2, 3, 4\}$ をカバーする
 最大マッチング



このマッチングは
 $B' = \{c, d\}$, $N' = \{2, 3, 4\}$ を同時に
 カバーする最大マッチング

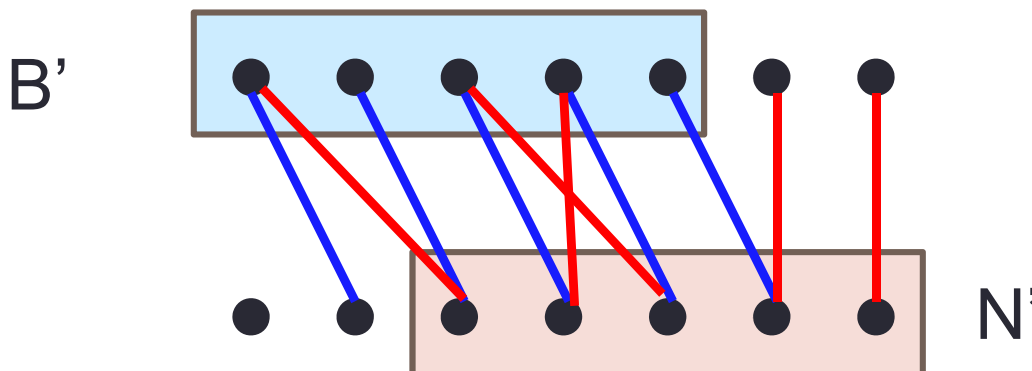
カバーする頂点の制約(両側)

命題2 B の部分集合 B' および N の部分集合 N' に対し,
 B' をカバーするマッチング, および N' をカバーするマッチングが
それぞれ存在

→ B' と N' を同時にカバーする(最大)マッチングが存在

[証明]

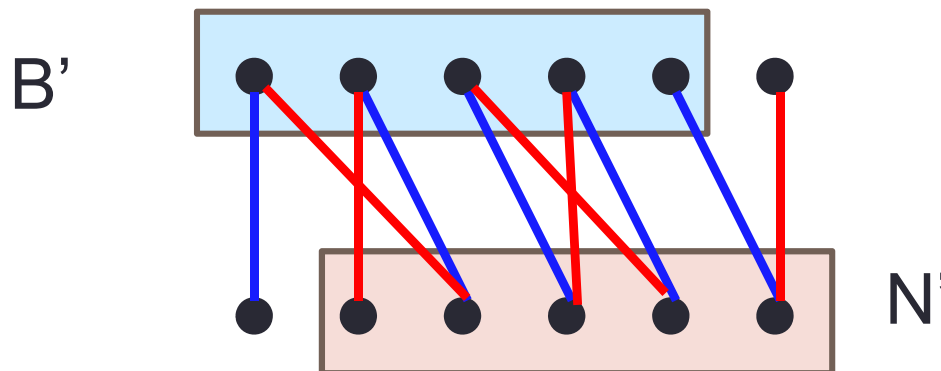
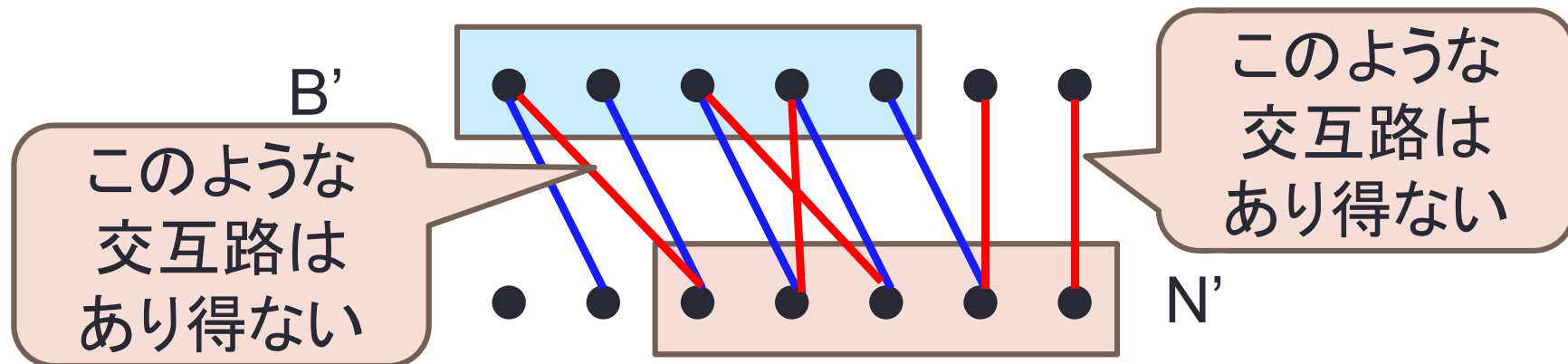
- M : B' をカバーする最大マッチング(命題1(i)より存在)
- そのようなマッチングの中で, カバーする N' の頂点数が最大とする
- M が N' すべてをカバーしていないと仮定 → 矛盾を導く
- M^* : N' すべてをカバーする最大マッチング



カバーする頂点の制約(両側)

[証明のつづき]

- 枝集合 $M \cup M^*$ からなるグラフ $\rightarrow$ 交互路と交互閉路に分解できる
 - M に関する増加路なし, M^* に関する増加路なし
(M, M^* とともに最大マッチングだから)

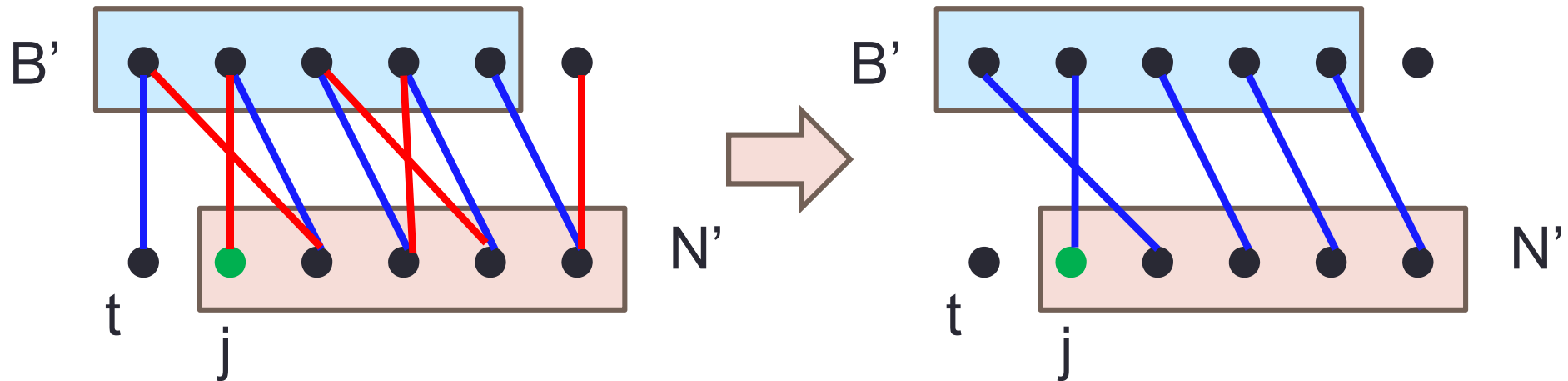


$\therefore$ 交互路の枝数は
すべて偶数

カバーする頂点の制約(両側)

[証明のつづき]

- $j \in N'$: マッチング M でカバーされていない頂点

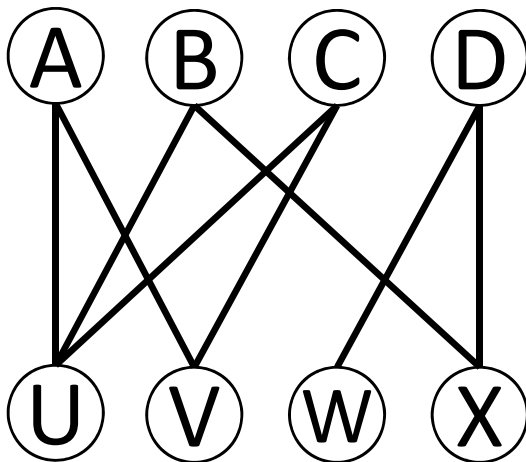


- j はマッチング M^* でカバーされている
 → j から始まる交互路 P が存在, M^* の枝から始まる
- 交互路 P の枝数は偶数
 → 最後の枝は M , 最後の頂点 t は N' に含まれない
 ($\because t \in N' \rightarrow t$ に接続する M^* の枝が存在 $\rightarrow$ 交互路は終わらない)
- $\therefore$ 交互路 P を使って M の枝を入れ替え $\rightarrow j$ がカバーされる
 → カバーできる N' の頂点数が増える(矛盾)

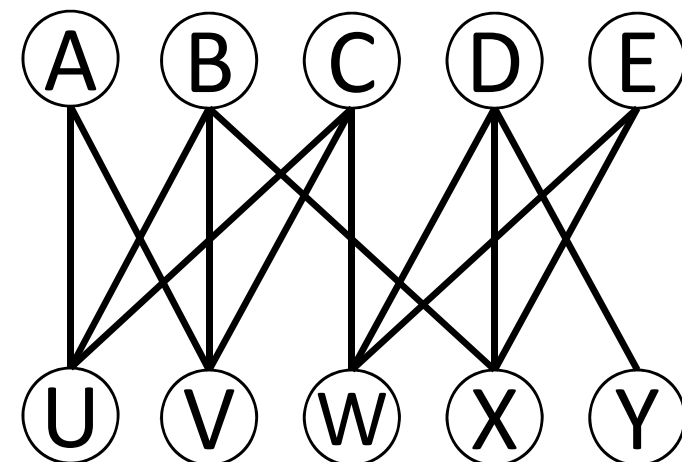
演習問題

問1: 以下のグラフの最大マッチングを, 前のスライドのアルゴリズムにより求めよ. 各反復でのマッチングと使った増加路を(図を使って)明記すること.
(増加路の求め方は, 説明する必要なし)

(1)



(2)



演習問題

問2: 下記のように評価値と財の価格(赤い数字)が与えられたとき, 均衡価格か否か判定したい.

(1) (i), (ii), (iii) それぞれに対し, 判定のための二部グラフを書け.

また, 頂点集合 B' , N' を明記せよ.

(2) 頂点集合 B' および N' をカバーするマッチングが存在するか否か, 調べよ. 存在する場合は具体的に求めること. 存在しない場合はその理由を述べよ.

(i)

	$v(i,j)$	A	B	C
3	①	2	3	6
6	②	6	7	7

(ii)

	$v(i,j)$	A	B	C
1	①	3	1	0
5	②	7	6	7
6	③	1	7	8

(iii)

	$v(i,j)$	A	B	C
0	①	3	1	0
5	②	7	6	7
6	③	1	7	8
0	④	0	0	4