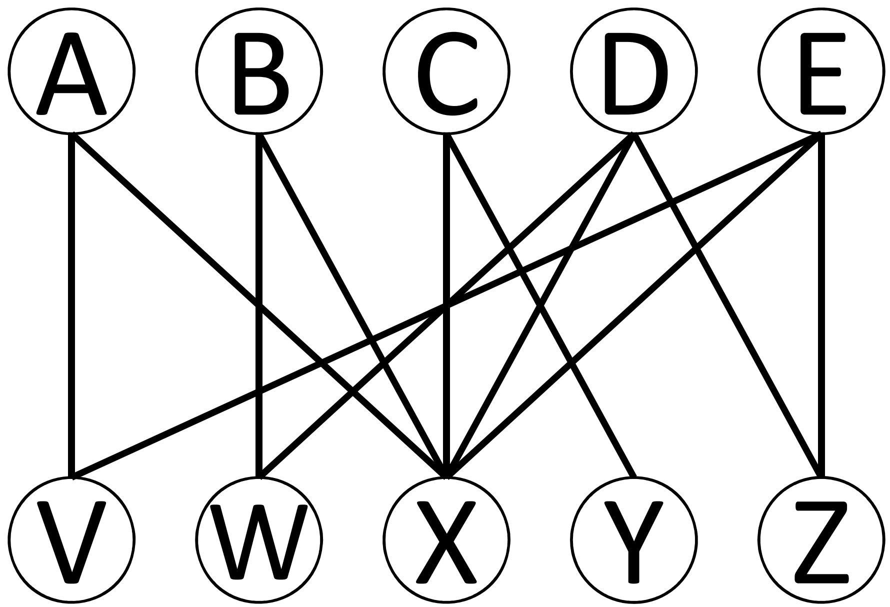
経営経済のための最適化理論
第3回
最大マッチング問題

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

マッチング問題

- 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：労働者の仕事への割り当て
  - 5人の労働者 A, B, C, D, E, 5つの仕事 V, W, X, Y, Z
  - 各々の労働者は従事可能な仕事と不可能な仕事がある
    ➔ グラフで表現可能



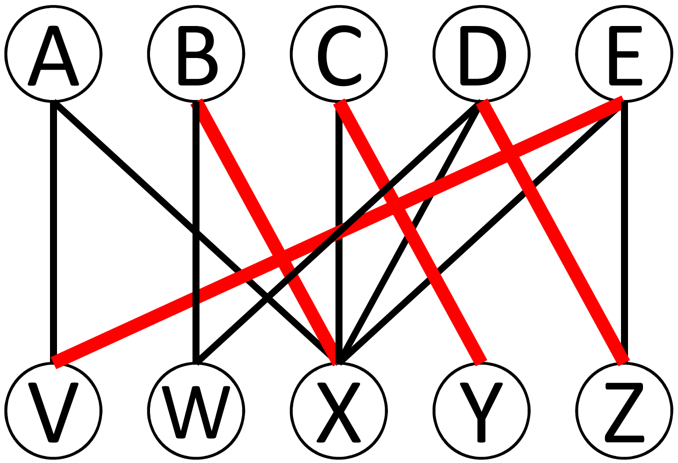
頂点の2つのグループ
V_1={A, B, ..., E} と V_2={V, W, ..., Z}
の間にのみ，枝が存在
← 二部グラフと呼ばれる

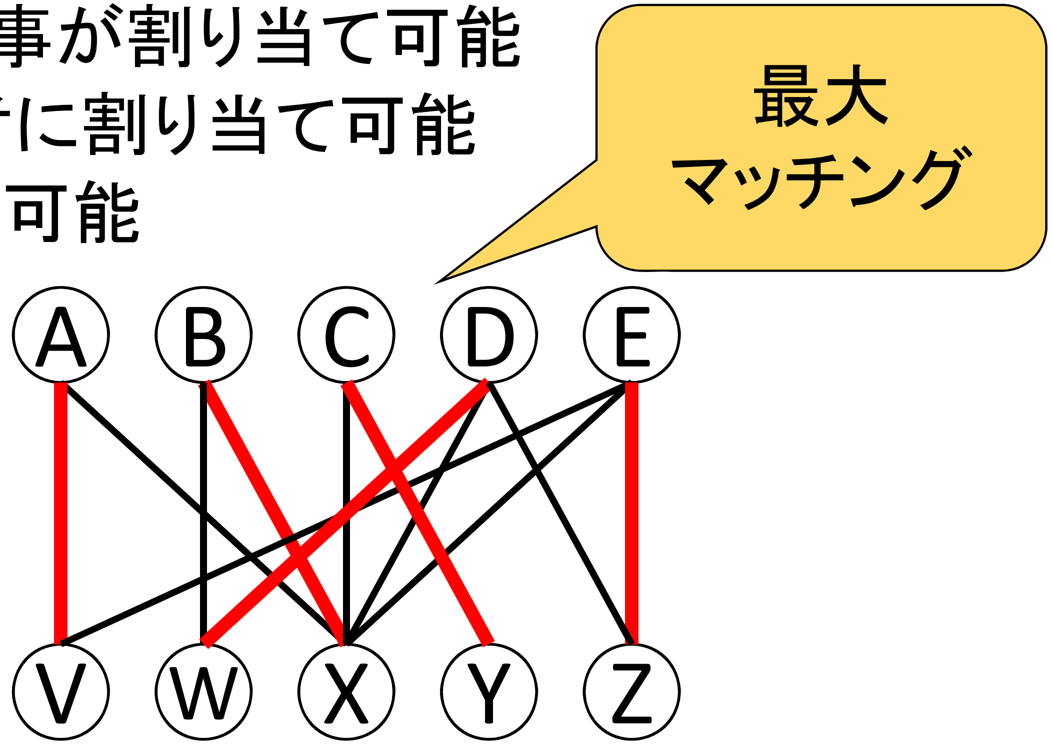
※2部グラフにおいて，V_1とV_2の頂点数は同じでなくても良い

最大マッチング問題

- できるだけ多くの労働者に仕事を割り当てたい
  - 各労働者には，高々1つの仕事が割り当て可能
  - 各仕事は，高々一人の労働者に割り当て可能

  ➔ 仕事の割当を枝集合で表現可能





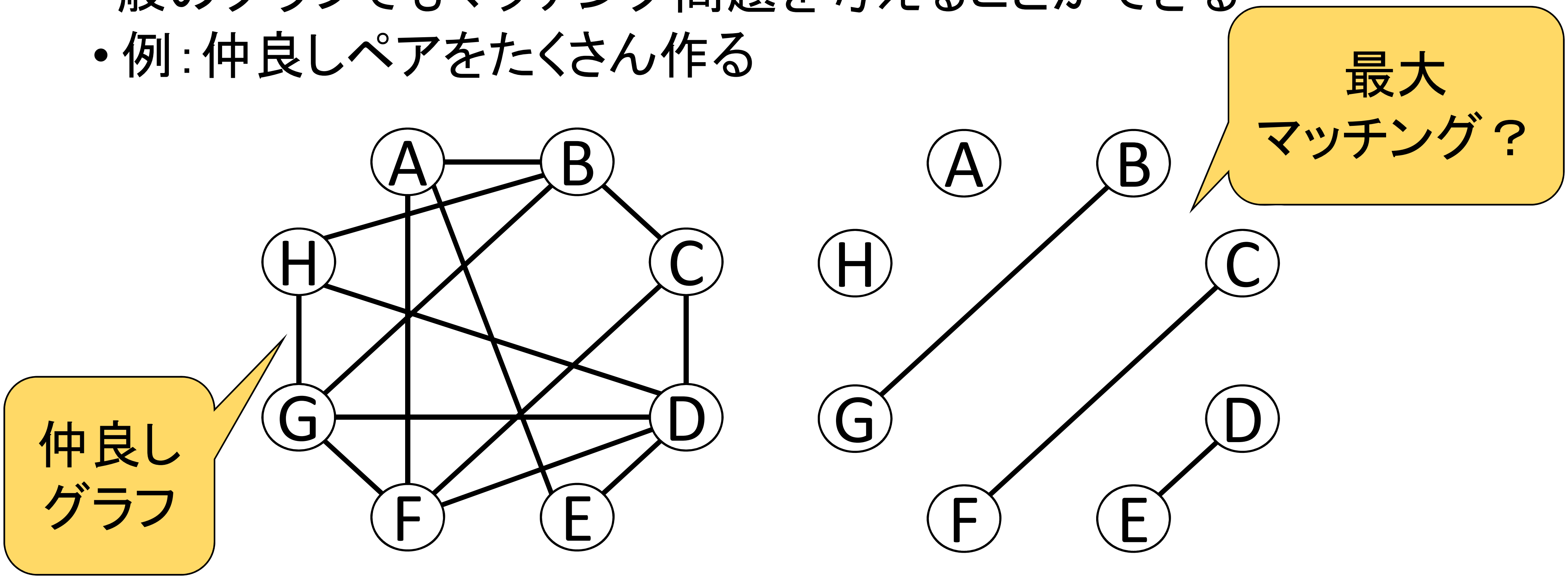
- グラフのマッチング：

  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

一般グラフのマッチング問題

- 一般のグラフでもマッチング問題を考えることができる
  - 例：仲良しペアをたくさん作る



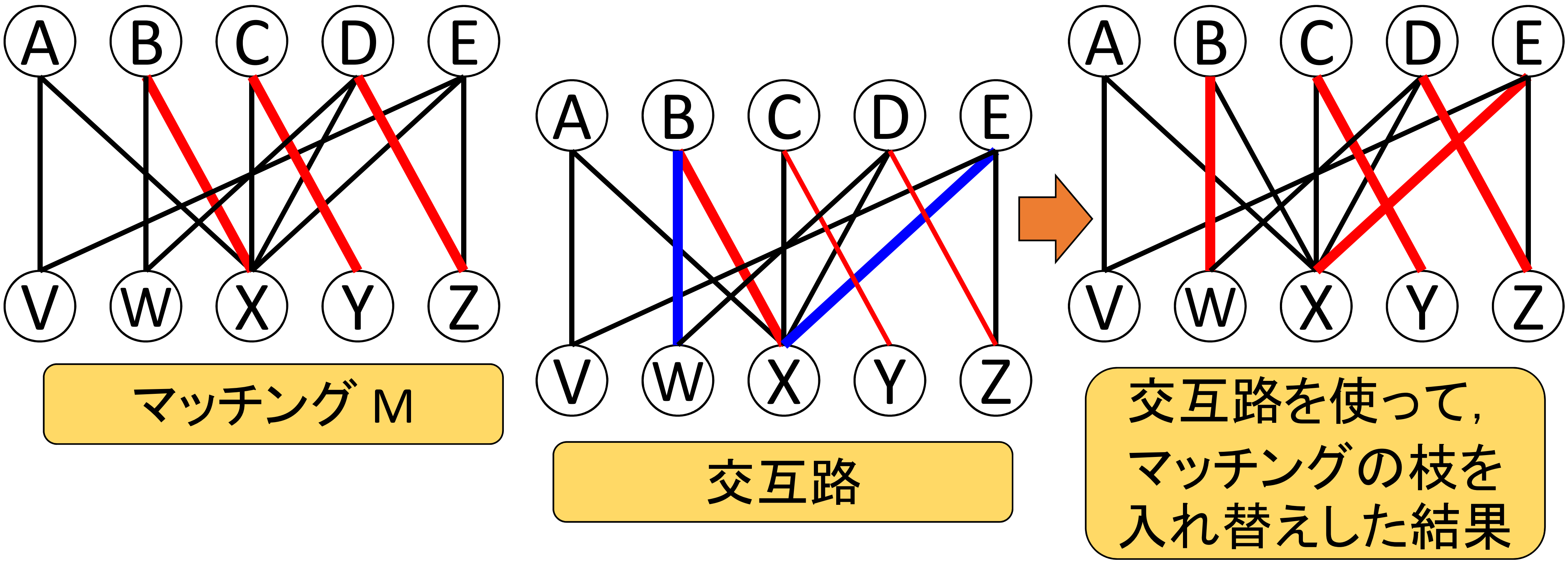
- グラフのマッチング：
  グラフの枝集合であって，各頂点に接続する枝の数（次数）が１以下
- 最大（サイズ）マッチング問題：
  与えられたグラフのマッチングの中で，枝数最大のものを求める

マッチングと交互路

マッチングに関する交互路

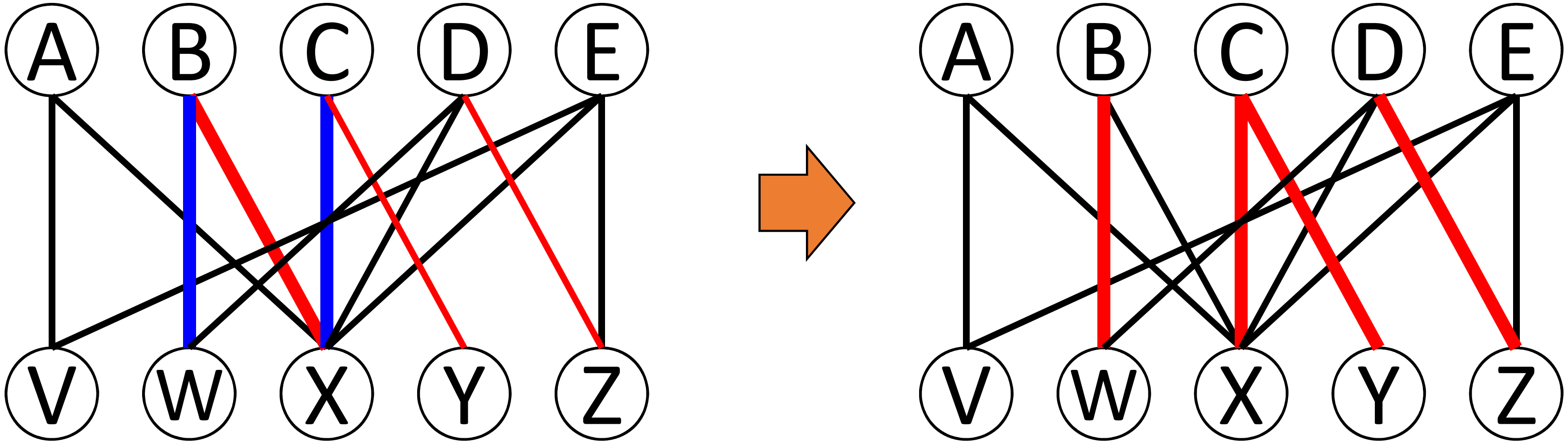
交互路 – 新しいマッチングを作るための「道具」

定義：マッチング Mに関する交互路 ←→ 次の2条件を満たす路 P
(a) 路 P には Mの枝とM以外の枝が交互に現れる
(b) 路 P に入っている M の枝を削除し，Pに入っているM以外の枝を追加すると，別のマッチングが得られる



マッチングに関する交互路

交互路でない例



条件(b)を満たさない
枝を入れ替えても
　マッチングにならない

交互路の定義の書き換え

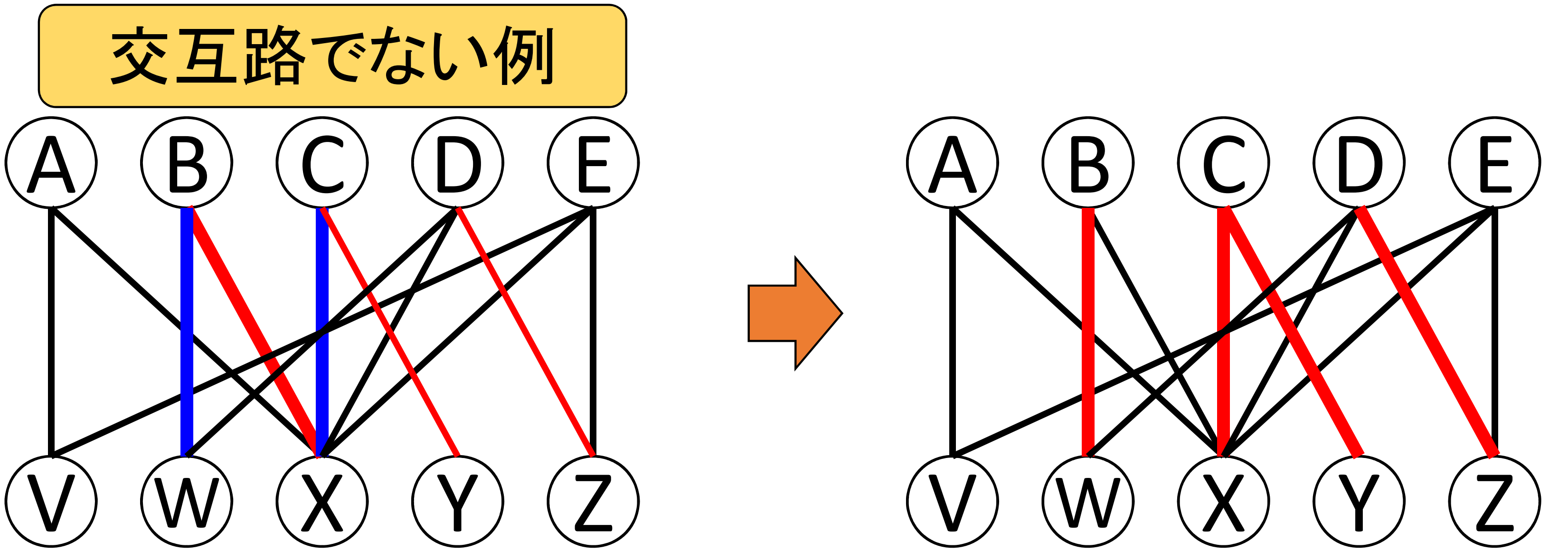
定義：マッチング Mに関する交互路 ←→ 次の2条件を満たす路 P
(a) 路 P には Mの枝とM以外の枝が交互に現れる
(b) 路 P に入っている M の枝を削除し，Pに入っているM以外の枝を追加すると，別のマッチングが得られる

⇩

条件(b)を書き換えると. . .

定義：マッチング Mに関する交互路 ←→ 次の3条件を満たす路 P
(a) 路 P には Mの枝とM以外の枝が交互に現れる
(b’) Pの最初の枝はM以外の枝のとき，
Pの始点にはMの枝が接続していない
(b’’) Pの最後の枝はM以外の枝のとき，
Pの終点にはMの枝が接続していない

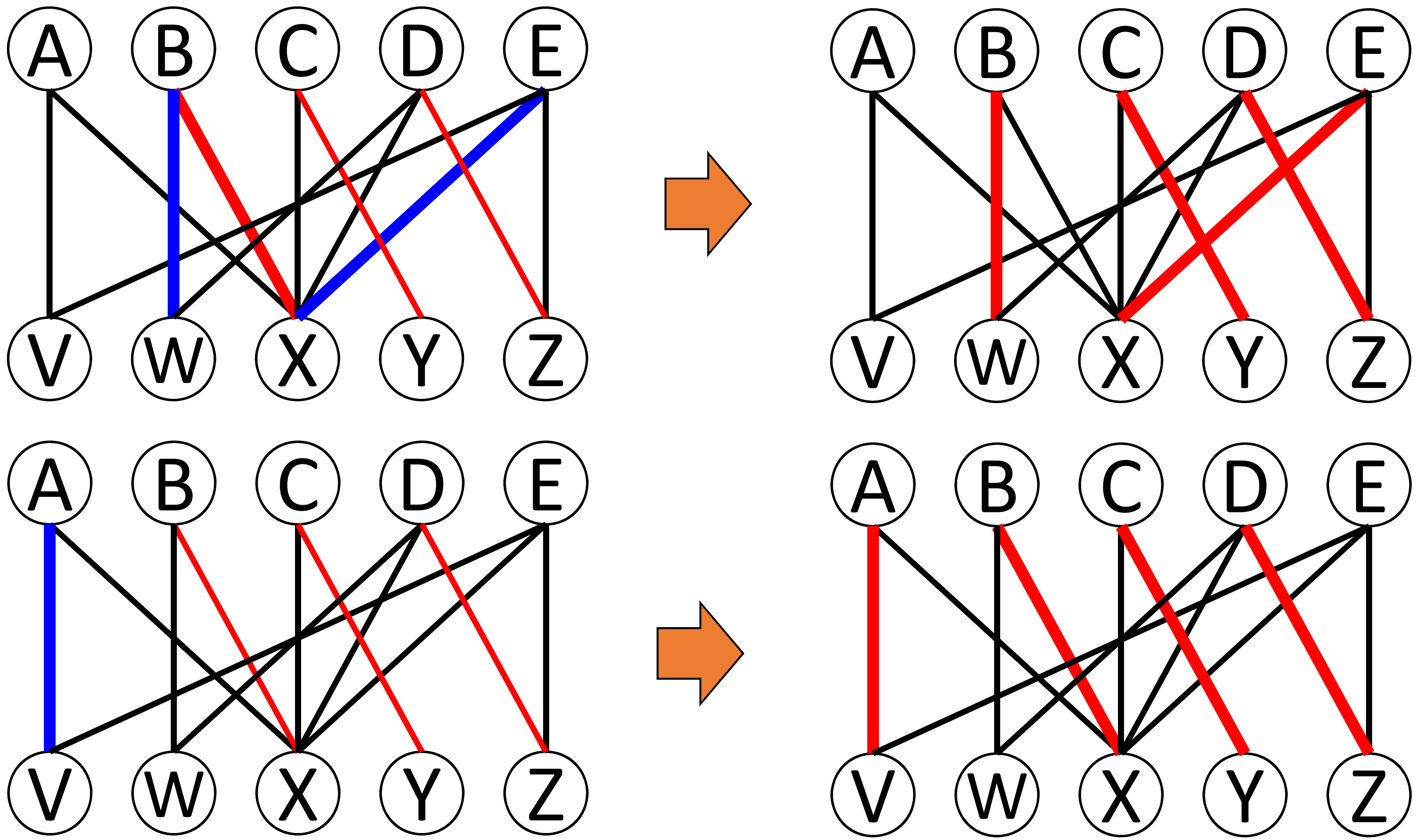
マッチングに関する交互路



交互路 P = {(C,X), (X,B), (B,W)}
最初の枝 (C,X) は M 以外の枝
始点 C にはマッチングの枝 (C,Y) が接続
∴ 条件(b’) を満たさない
このような場合，枝を入れ替えても
マッチングにならない

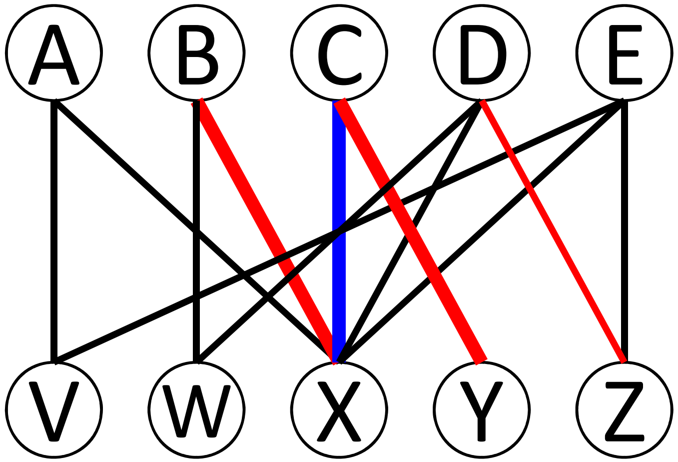
マッチングに関する交互路

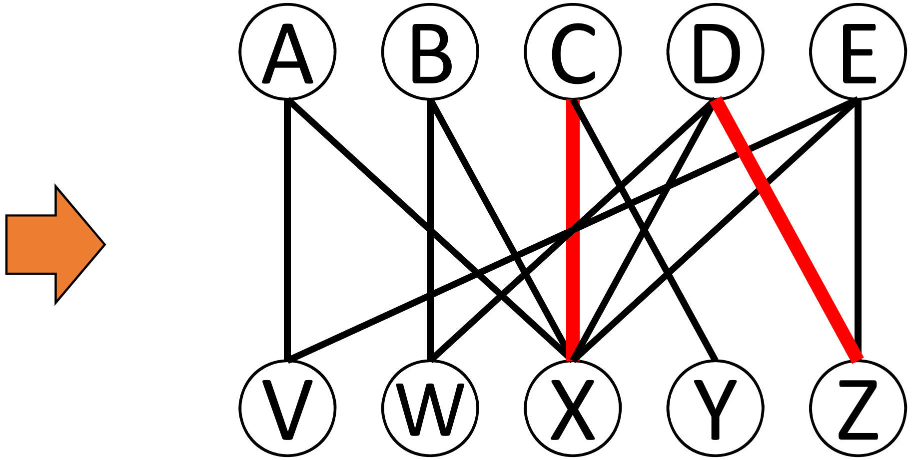
交互路タイプ1：M以外の枝数＝Mの枝数＋1
（枝を入れ替えるとマッチングの枝数が1増える）
➔ 増加路と呼ばれる



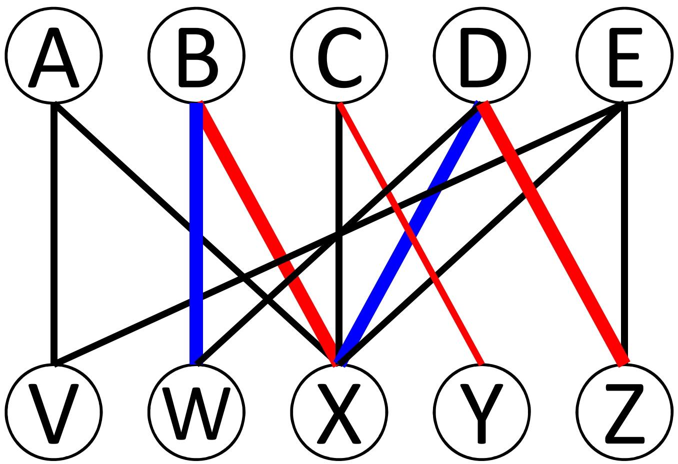
マッチングに関する交互路

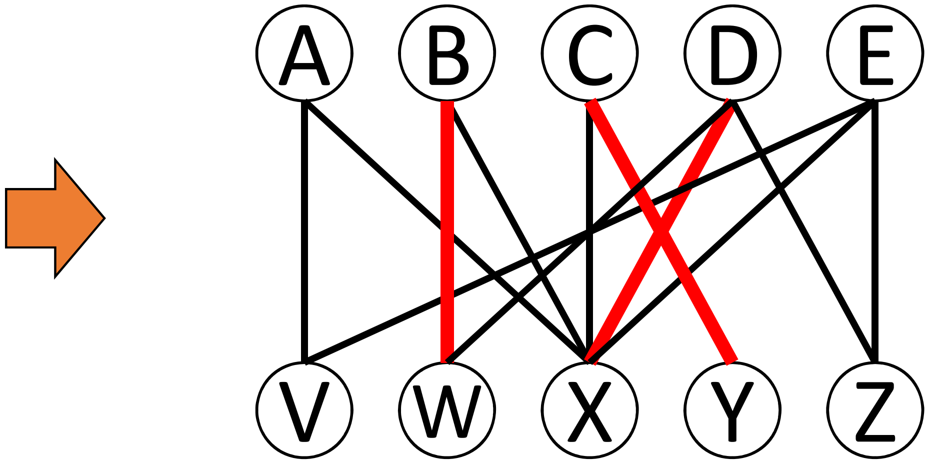
交互路タイプ2：M以外の枝数＝Mの枝数−1
（枝を入れ替えるとマッチングの枝数が1減る）





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，マッチングの枝数は不変）



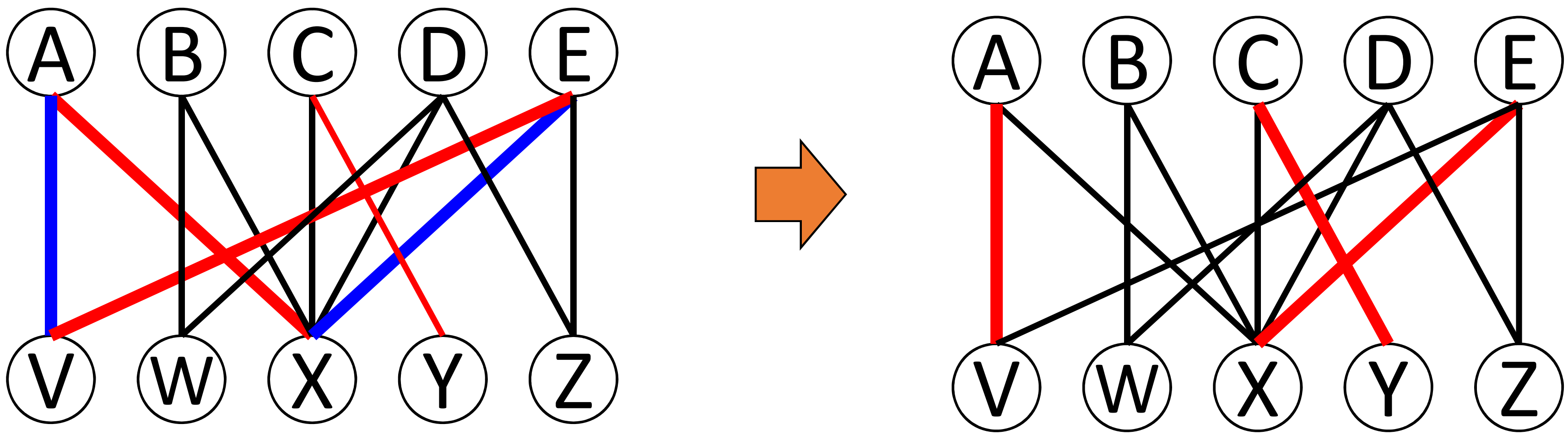


マッチングに関する交互閉路

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, マッチングの枝数は不変

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
⇔Pを使ってMの枝を入れ替えると，新しいマッチングM’ができる

$$M' = (M \setminus P) \cup (P \setminus M)$$

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

逆も成立する

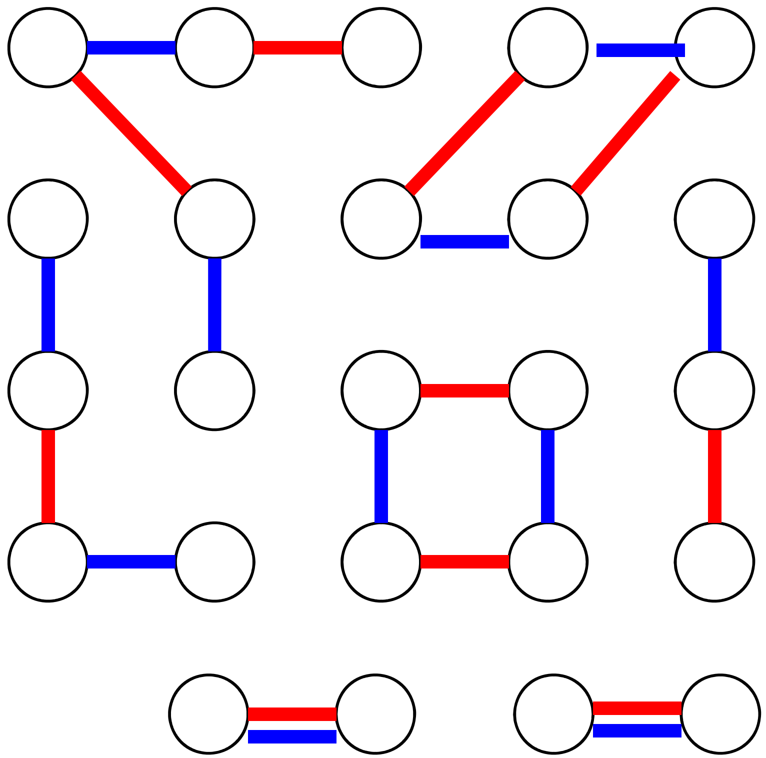
命題 マッチングMに対し，
Mに関する増加路が存在⇐Mは最大マッチングではない
（対偶：Mは最大マッチング⇐ Mに関する増加路が存在しない）

2つのマッチングに関する性質

命題 同じグラフの2つのマッチング M, M’ に対し,
差集合 $(M' \setminus M) \cup (M \setminus M')$ は交互路, 交互閉路に分解可能.
とくに, |M’|>|M| ➔ M に関する増加路を含む.

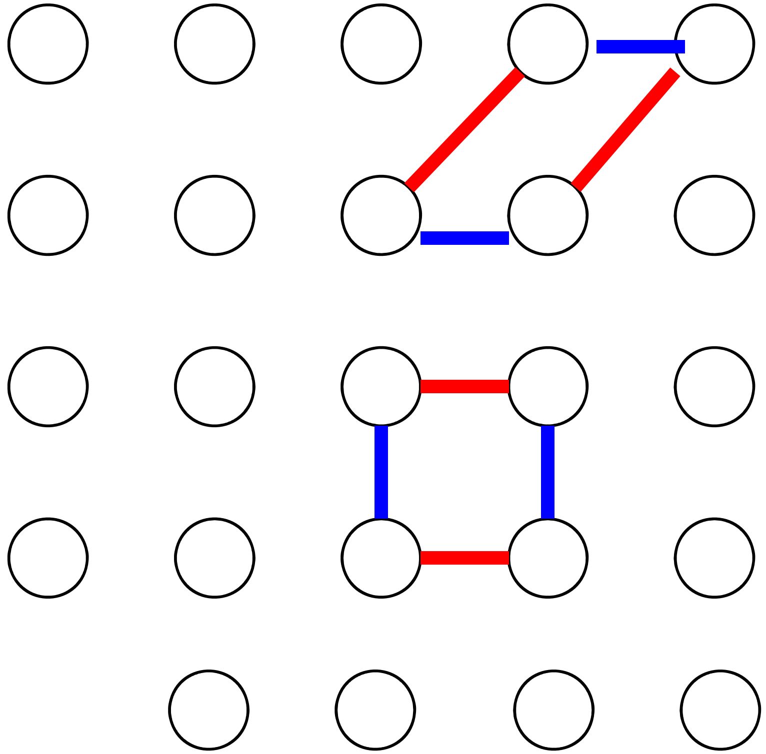
(証明) $\tilde{E} = (M' \setminus M) \cup (M \setminus M')$ とおく.
グラフの各頂点には, M と M’の枝が
　高々1本ずつ接続する.

Mまたは M’の枝のみが接続する頂点 v_0を
　任意に選び, 　v_0 から $\tilde{E}$ の枝をたどる
→ Mの枝と M’ の枝が交互に現れる路
　　が見つかる
→ $\tilde{E}$ から削除.



同様の操作を繰り返す → 交互路 $P_1, P_2, \ldots, P_k$ が得られる

２つのマッチングに関する性質



（証明のつづき）前ページの操作の結果，
各頂点にはMおよびM'の両方の枝が接続
（もしくは何も接続しない）

$\tilde{E}$ の枝の接続する頂点 v_0を
任意に選び， v_0 から $\tilde{E}$ の枝をたどる
→ Mの枝とM' の枝が交互に現れる路
が見つかる
→ $\tilde{E}$ から削除.

同様の操作を繰り返す → 交互閉路 $C_1, C_2, \ldots, C_h$（分解終了）

２つのマッチングに関する性質

（証明のつづき）$\tilde{E} = (M' \setminus M) \cup (M \setminus M')$ を分解して

交互路 $P_1, P_2, \ldots, P_k$

交互閉路 $C_1, C_2, \ldots, C_h$ が得られた.

得られた交互閉路 C_j に対し，$|C_j \cap M'| = |C_j \cap M|$

$|M'| - |M| = |M' \setminus M| - |M \setminus M'|$

$= \sum_{i=1}^{k} |P_i \cap M'| + \sum_{j=1}^{h} |Q_j \cap M'| - \sum_{i=1}^{k} |P_i \cap M| - \sum_{j=1}^{h} |Q_j \cap M|$

$= \sum_{i=1}^{k} (|P_i \cap M'| - |P_i \cap M|)$

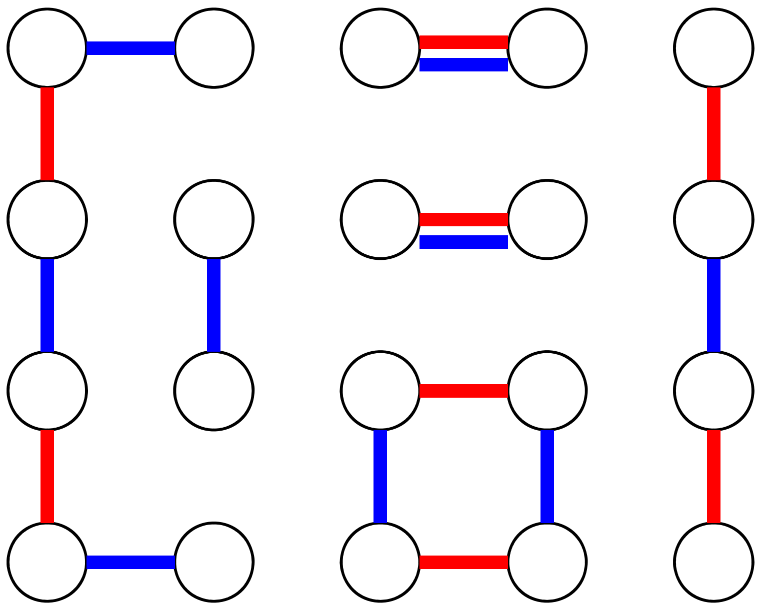
$\therefore$ $|M'| - |M| > 0$ ならば，ある i に対し $|P_i \cap M'| > |P_i \cap M|$

つまり，P_i はM に関する増加路　■

交互路によるマッチングの最適性条件

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

（証明）Mは最大ではない ➔ |M*| > |M|を満たすマッチングをM*とおく
枝集合$(M^* \setminus M) \cup (M \setminus M^*)$に注目
➔ 前のスライドの命題より,
　　M に関する増加路が存在 ■



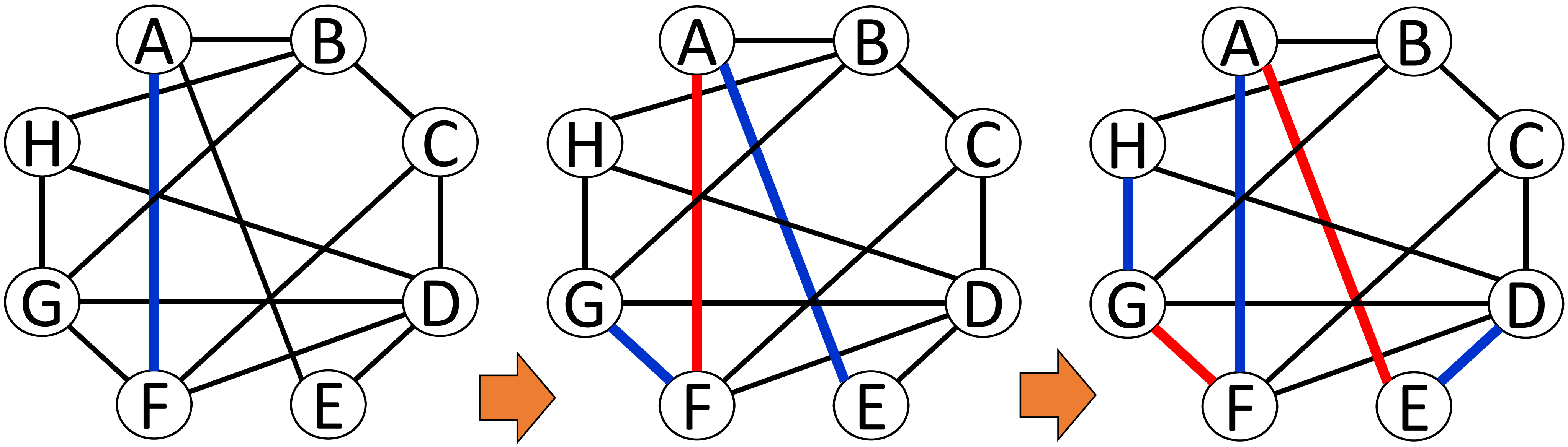
最大マッチングの計算

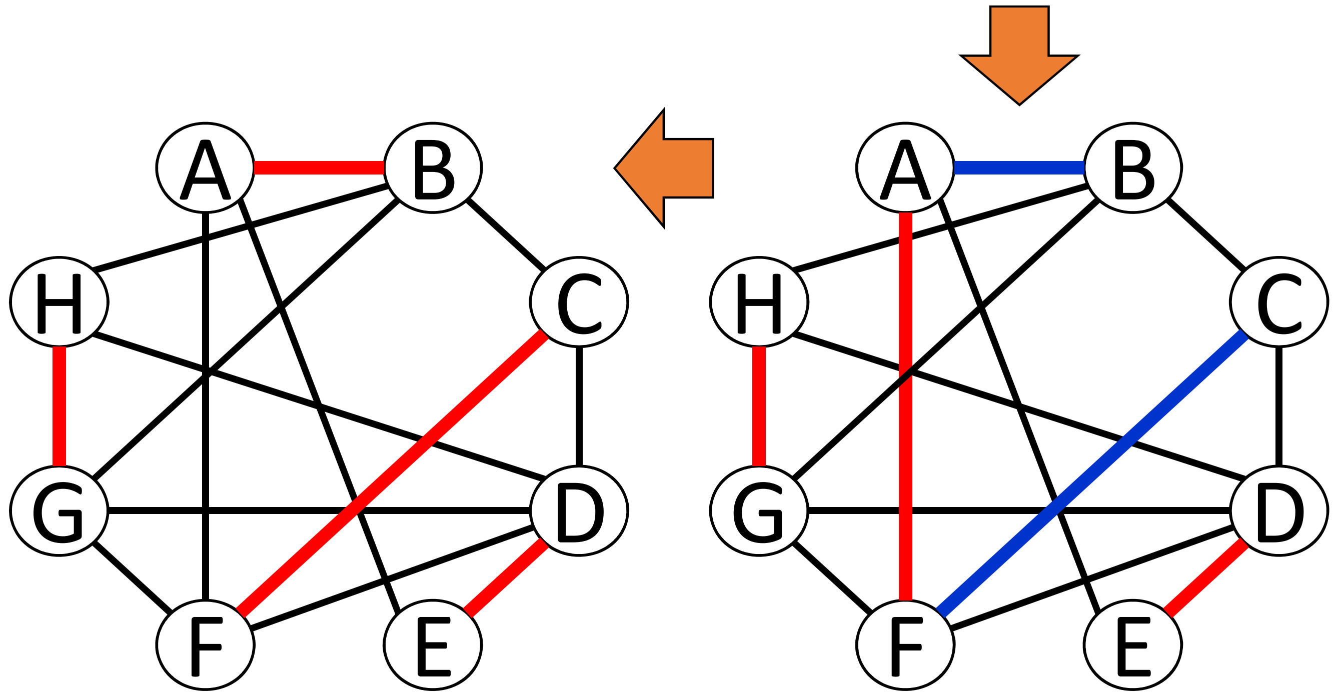
最大マッチング問題のアルゴリズム

「Mは最大マッチング←→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→現在のMは最大マッチング（終了）
存在する→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



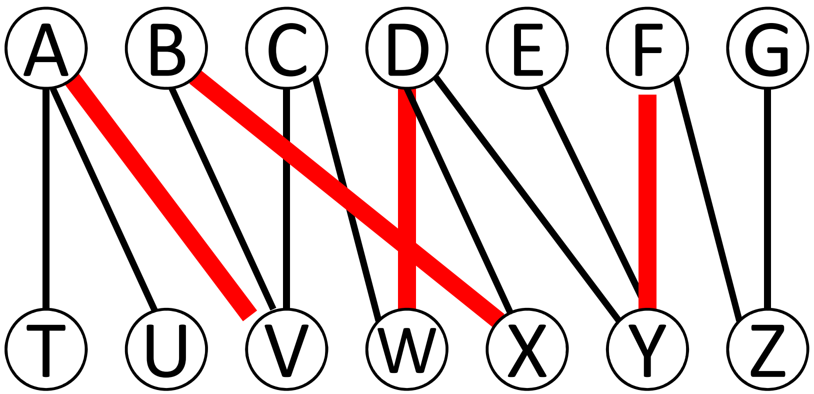


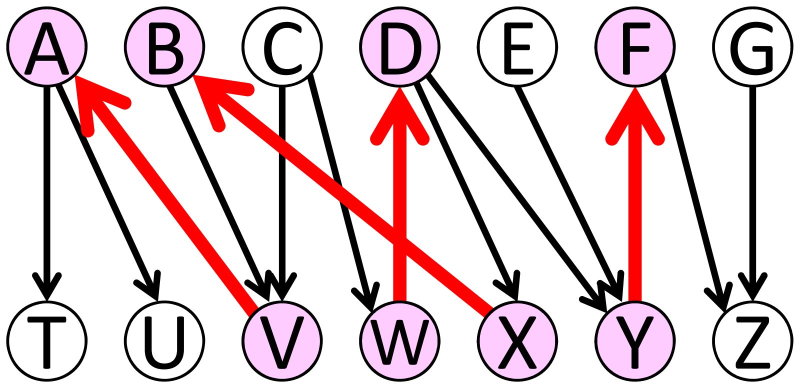
増加路なし
➔
最大マッチング

増加路の見つけ方：二部グラフの場合

- 増加路を見つける問題は，有向路を見つける問題に帰着できる
- 二部グラフの頂点集合が V_1, V_2 に分かれているとき，
  - マッチングMの枝は，V_2 の頂点から V_1 の頂点に向き付け
  - それ以外の枝は，V_1 の頂点から V_2 の頂点に向き付け
- V_1 の頂点のうち，
  - マッチングの枝が接続している頂点の集合→U_1
  - 残りの頂点の集合→T_1
- U_2, T_2 も同様に定義

ピンクの頂点：U_1とU_2
白い頂点：T_1とT_2



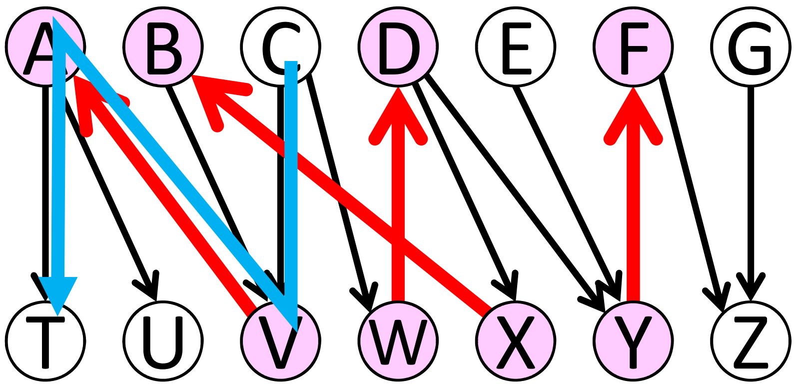


増加路の見つけ方：二部グラフの場合

- 新しい有向グラフにおける T_1 の頂点から T_2の頂点への有向路
  ←→元の二部グラフの（Mに関する）増加路

∴有向グラフにおいて，上記のような有向路を求めればよい
（有向路を見つける効率的なアルゴリズムは幾つか存在）

ピンクの頂点：U_1とU_2
白い頂点：T_1とT_2



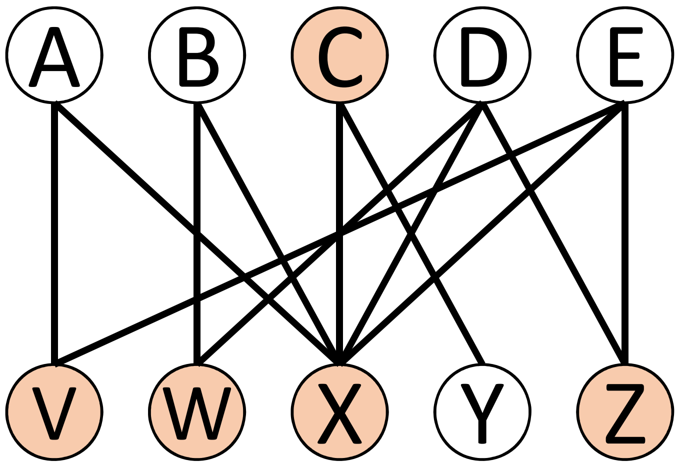
一般のグラフでも，
増加路の計算は
「効率的」に
できる，
でも計算方法は
かなり複雑

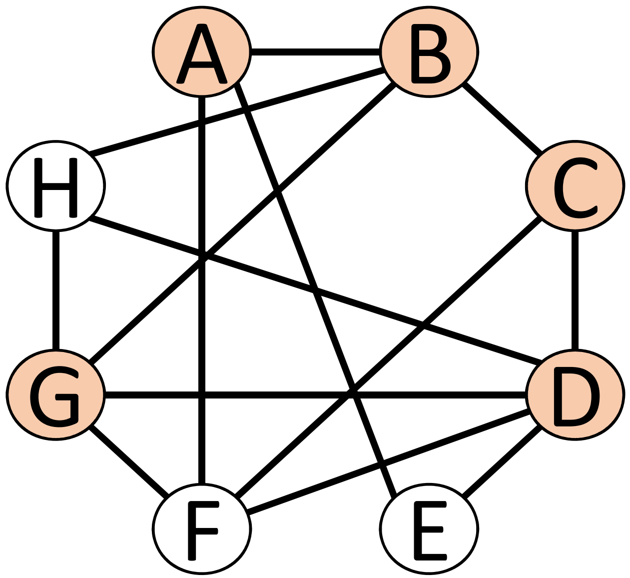
最大マッチング最小被覆定理

グラフの頂点被覆

- マッチングが最大でない証拠は？→増加路
- マッチングが最大である証拠は？

定義: グラフの頂点集合Sは頂点被覆
⇔グラフの各枝(u,v)に対し $u \in S$ または $v \in S$
（$u, v \in V \setminus S$ となる枝(u,v)が存在しない）

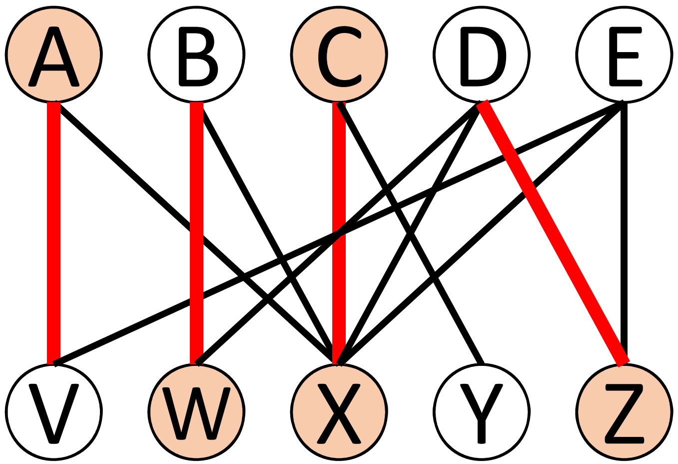


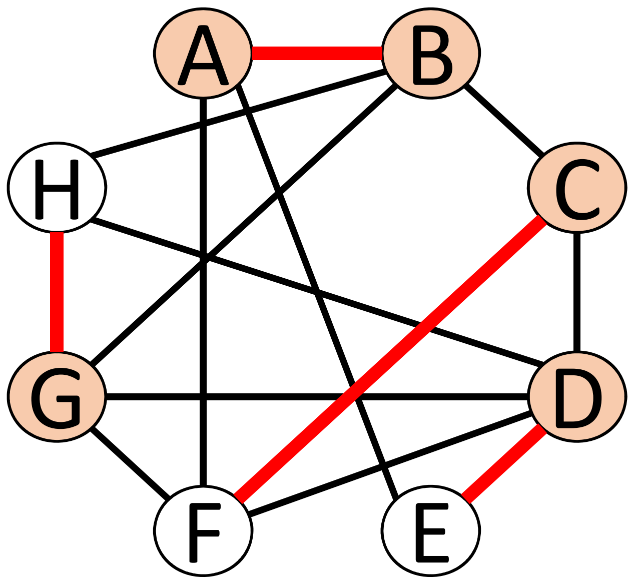


頂点被覆とマッチング

- マッチングM と頂点被覆 S の要素数の関係

命題　任意のマッチングMと頂点被覆 Sに対し，　|M|≦|S|





定義: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

頂点被覆とマッチング（証明）

命題　任意のマッチングMと頂点被覆 Sに対し，　$|M| \leqq |S|$

（証明）マッチングと頂点被覆の定義を使う.

$k=|M|$, Mの枝を$e_1, \ldots, e_k$ とおく.

頂点被覆の定義により，各枝 e_i は S の中のある頂点 v_i に接続

ここで，$v_1, \ldots, v_k$ は全て異なる.

もし $v_i = v_j$ $(i \neq j)$とすると，v_i には枝 e_i, e_j の2本の枝が接続する.

これはマッチングの定義に反する.

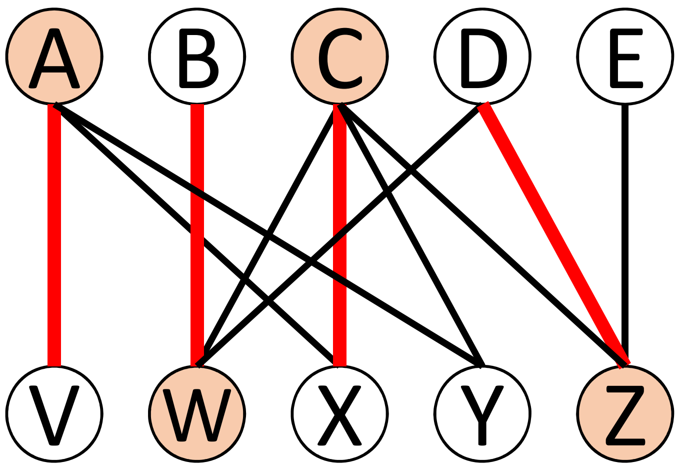
$\{v_1, \ldots, v_k\} \subseteq S$ なので，$|S| \geqq k=|M|$　■

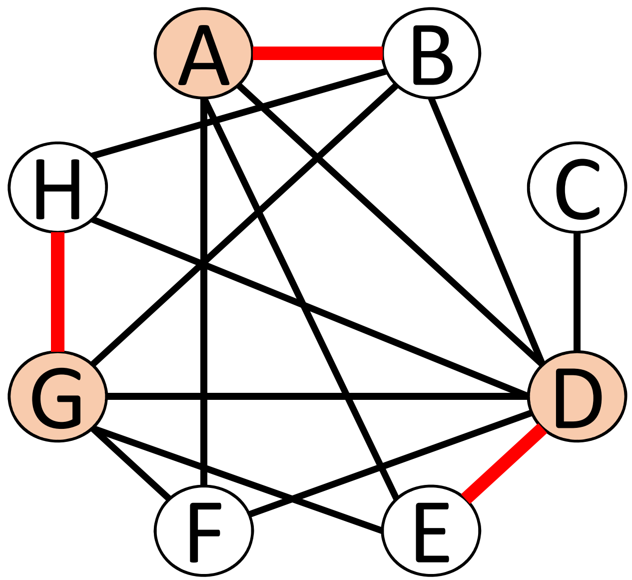
頂点被覆とマッチング

命題　任意のマッチングMと頂点被覆 Sに対し,　|M|≦|S|

系　任意のマッチングMと頂点被覆 Sに対し,
|M|＝|S|　➔ M は最大マッチング（S は頂点数最小の被覆）

証明は演習問題





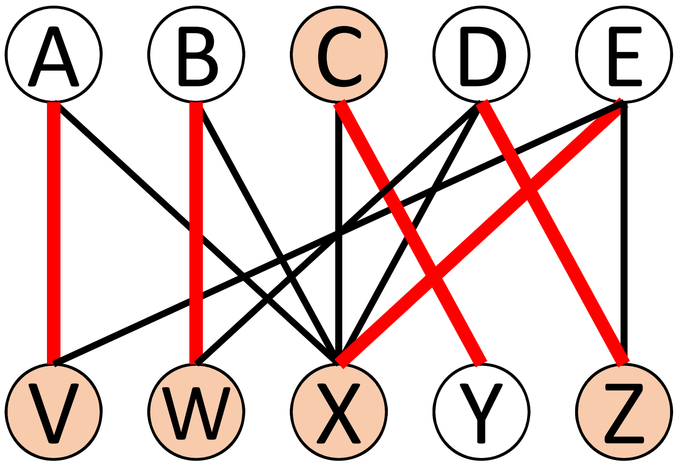
最大マッチング最小被覆定理

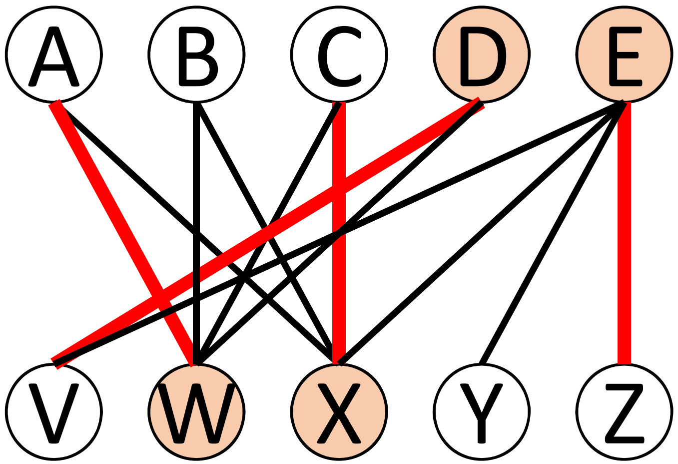
一般のグラフでは
成り立たない

最小(頂点)被覆＝頂点数最小の頂点被覆

定理 (最大マッチング最小被覆定理, Kőnig (ケーニグ) の定理)
　二部グラフにおいて,
　　　最大マッチングの枝数＝最小頂点被覆の頂点数

証明はやや難しい



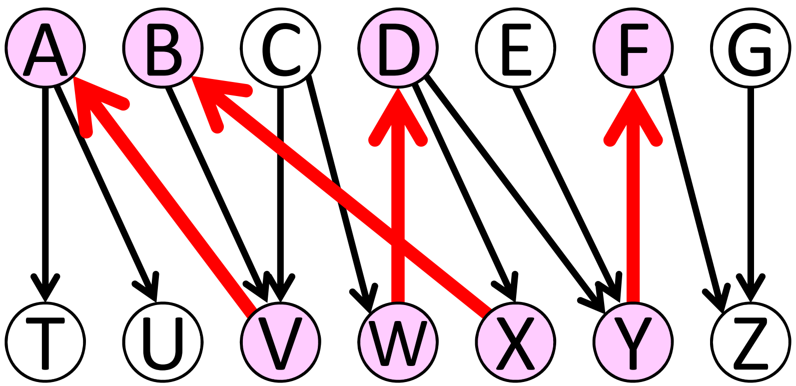


最大マッチング最小被覆定理の証明(0)

用語の定義:

- 頂点uから頂点v へ到達可能←→ u から v への路が存在
- 頂点集合 S から 頂点集合 T へ到達可能

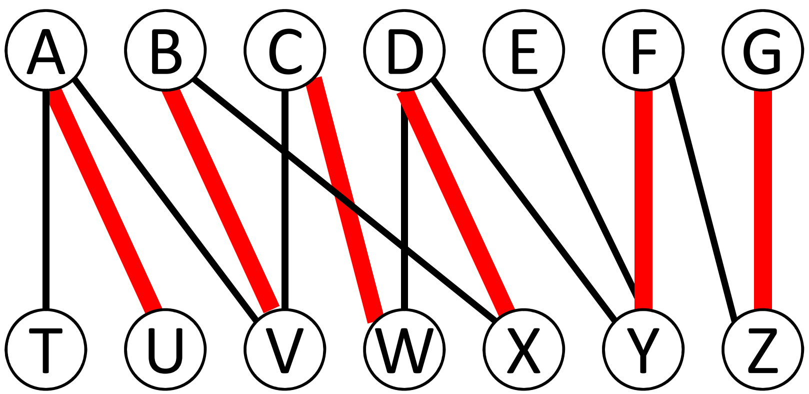
←→ ある u∈S からある v∈T へ到達可能

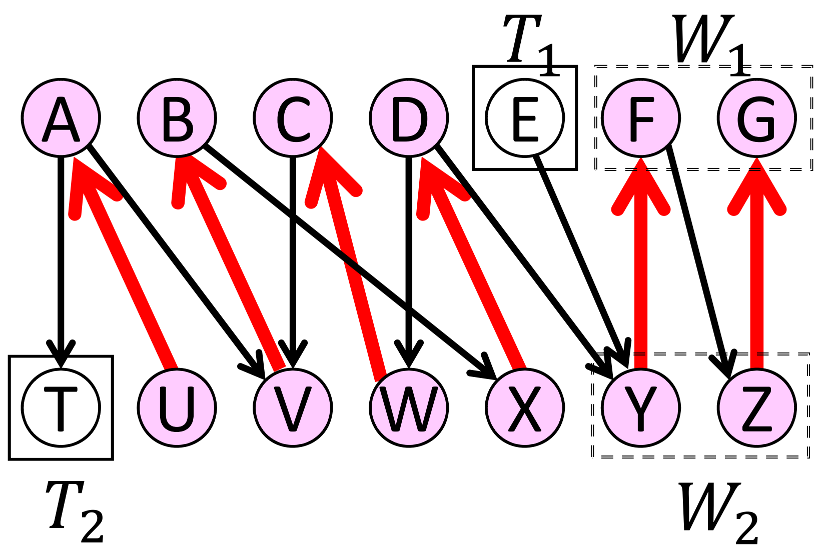


- 頂点Cから頂点Zに到達可能
- 頂点集合{E,G}から頂点集合{T,U}に到達不可能

最大マッチング最小被覆定理の証明(１)

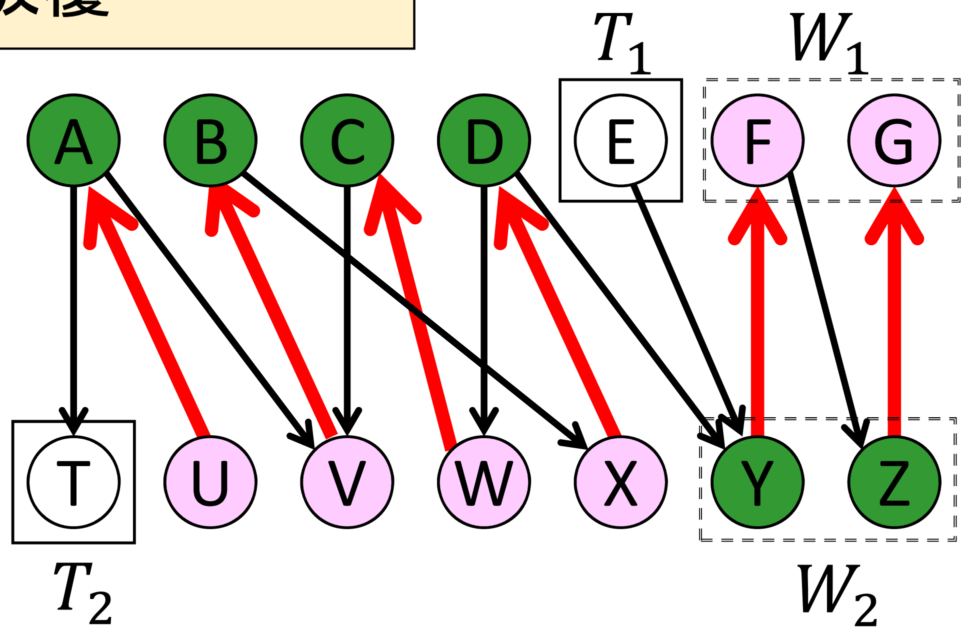
- 最大マッチング M* を利用（存在は自明）
- 増加路の計算のときと同様に，有向グラフおよび U_1, T_1, U_2, T_2 を定義
  - 増加路は存在しない➔T_1からT_2へ到達不可能 ①
- W_2＝有向グラフにおいて，T_1から到達可能なV_2の頂点の集合
  - $T_2 \cap W_2 = \emptyset$（∵①）　➔　$W_2 \subseteq V_2 \setminus W_2 = U_2$
- W_1＝マッチングM*において，W_2の相手となる頂点の集合
  - $W_1 \subseteq U_1$（∵ W_1, U_1 の定義）





最大マッチング最小被覆定理の証明(2)

命題　$S = (U_1 \setminus W_1) \cup W_2$ は頂点被覆



- |W$_1$|=|W$_2$|, |M*|=|U$_1$|なので,
  |S|=|M*|.

よって, S は最小頂点被覆

命題の証明:
すべての枝が S に接続していることを示せば良い.

(1) マッチングの各枝: S の定義より, S のいずれかの頂点に接続

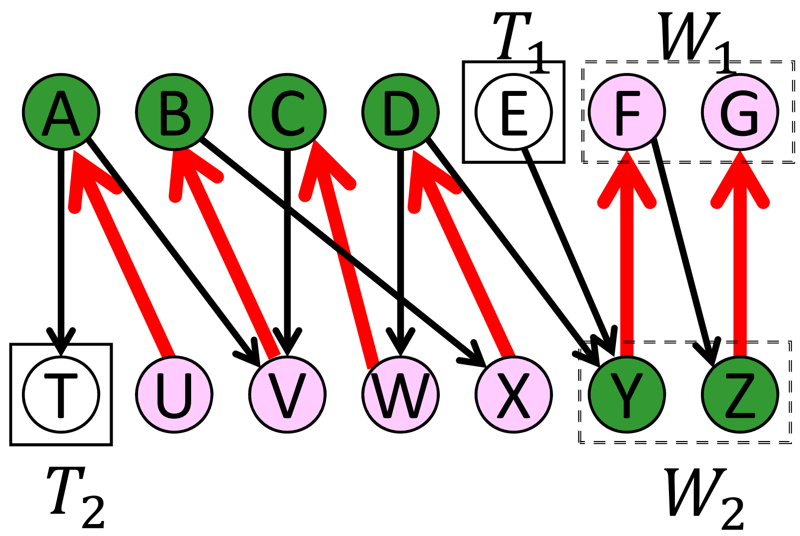
(2) マッチング以外の枝 (u,v), $u \in V_1, v \in V_2$:

(i) $v \in W_2$ の場合: W$_2$⊆S なので, S に接続

(ii) $v \notin W_2$ の場合: W$_2$ の定義より, v はT$_1$ から到達不可能
W$_1$, W$_2$ の定義より, W$_1$ はT$_1$ からW$_2$を経由してのみ到達可能
∴ u はT$_1$∪W$_1$ に含まれない ➔ $u \in U_1 \setminus W_1$

最大マッチング最小被覆定理の証明(3)

命題　$S = (U_1 \setminus W_1) \cup W_2$ は頂点被覆



命題の背理法による証明:

ある枝(u,v)が S に接続していないと仮定

➔ $u \in V_1 \setminus (U_1 \setminus W_1) = T_1 \cup W_1,\ v \in V_2 \setminus W_2 = T_2 \cup (U_2 \setminus W_2)$

(1) $u \in T_1$ のとき: T_1 の定義より, (u,v) は M* の枝ではない.

　　また W_2 の定義より, $v \in W_2$.　これは $v \in V_2 \setminus W_2$ に矛盾.

(2) $u \in W_1$ のとき:

W_1, W_2 の定義より, $u \in W_1$ はT_1 からW_2を経由して到達可能.

したがって, T_1 から $v \in T_2$ に到達可能.

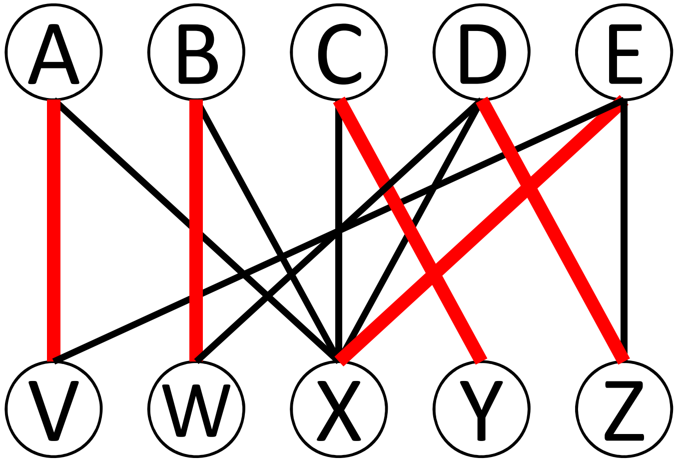
一方, W_2 の定義より, $v \in V_2 \setminus W_2$ はT_1 から $v \in T_2$ に到達不可能（矛盾）.

ホールの定理

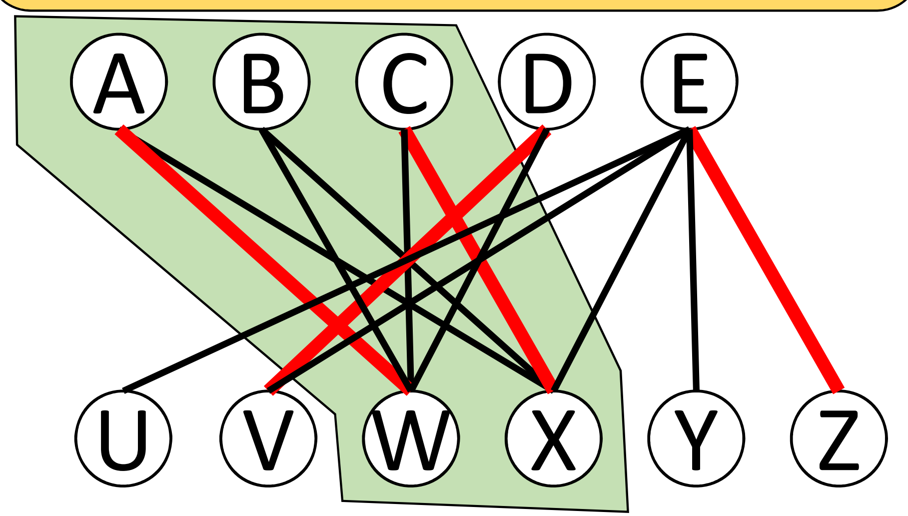
全員への割当可能性の判定

労働者全員に仕事を割り当てることができるか？

{A,...,E}全員に割り当て可能



Bには割り当てなし
➔割当方法を修正して，
全員に割り当てられるか？



命題

ある労働者の集合に対し，
「その労働者に割当可能な
仕事の数＜労働者の数」
➔ 全員への割当は不可能

A,B,Cに割当可能な仕事 = {W,X}
∴A,B,C全員に仕事を
割り当てることは不可能

ホールの定理

マッチングの言葉での言い換え:
頂点集合$V_1 \cup V_2$の二部グラフにおいて,
　V_1の頂点全てに接続するマッチングが存在するか？

命題　あるX $\subseteq V_1$に対し,「|X| > Xに隣接するV_2の頂点数」
➔ V_1の頂点全てに接続するマッチングは存在しない
（対偶: V_1の頂点全てに接続するマッチングが存在
➔ 任意のX $\subseteq V_1$に対し,「|X| ≦ Xに隣接するV_2の頂点数」）

逆も成り立つ

定理（Hall（ホール）の定理）
　二部グラフにおいて,
V_1の頂点全てに接続するマッチングが存在
⇔ 任意のX $\subseteq V_1$に対し,　|X| ≦ Xに隣接するV_2の頂点数

ホールの定理:証明

[➔] V_1の頂点全てに接続するマッチングMが存在

$V_1 = \{u_1, \dots, u_n\}$, $M = \{(u_1, v_1), \dots, (u_n, v_n)\}$ とおく

任意の$X = \{u_{i_1}, u_{i_2}, \dots, u_{i_k}\} \subseteq V_1$に対し,

$\{v_{i_1}, v_{i_2}, \dots, v_{i_k}\}$はXに隣接　∴ |X|≦ Xに隣接する$V_2$の頂点数

[🡐] 最大マッチング最小被覆定理をつかう

「V_1が最小被覆」と仮定➔最大マッチングの枝数= $|V_1|$

➔最大マッチングはV_1の頂点全てに接続する

以下,V_1が最小被覆であることを証明すればよい

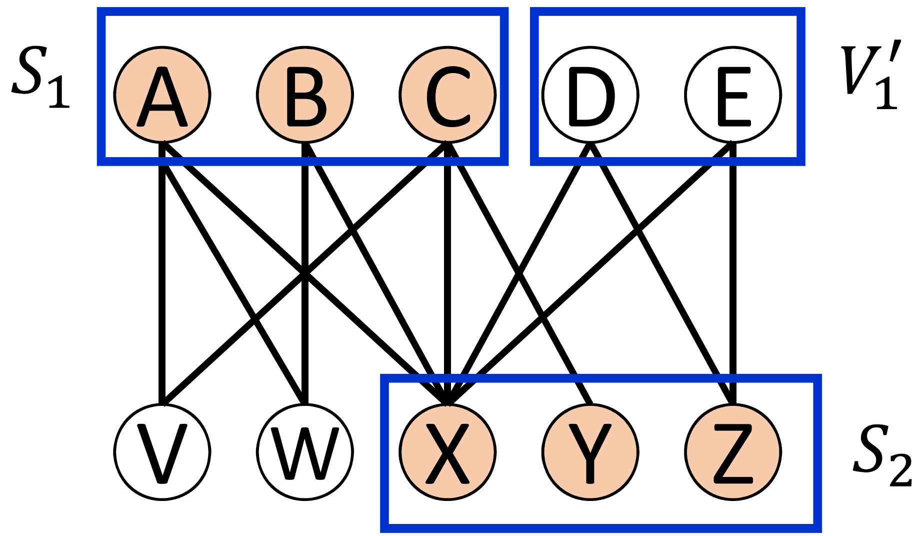
ホールの定理：証明（続き）

任意の頂点被覆 S に対し，|S|≧ $|V_1|$ を示す

$S_1 = S \cap V_1, S_2 = S \cap V_2,\ V_1' = V_1 \setminus S_1$ とおく

$u \in V_1'$ に接続する枝(u,v)は S_1 で被覆されない➔$v \in S_2$

∴V_1'に隣接するV_2の頂点⊆ S_2



仮定より，

$|V_1'| \leq V_1'$に隣接するV_2の頂点数

$\leq |S_2|$

∴$|S| = |S_1| + |S_2| \geq |S_1| + |V_1'| = |V_1|$ ■