

準線形回の独立性オラクルへのクエリを
使用するマトロイド交叉問題の
決定的な $(\frac{2}{3} - \epsilon)$ 近似アルゴリズム

寺尾 樹哉 (京都大学数理解析研究所 D2)

コンピュータ・科学研究会 (COMP)

① 愛媛大学

2025年12月5日

(Appeared at WADS'25)

マトロイド"とは?

Def.

有限集合 V とその空でない部分集合族 $\mathcal{I} \subseteq 2^V$ s.t.

① $S' \subseteq S \in \mathcal{I} \Rightarrow S' \in \mathcal{I}$

② $S, T \in \mathcal{I}, |S| > |T| \Rightarrow \exists e \in S \setminus T$ s.t. $T \cup \{e\} \in \mathcal{I}$

マトロイド" とは? - 線形独立性の一般化

Def.

有限集合 V とその空でない部分集合族 $\mathcal{I} \subseteq 2^V$ s.t.

$$\textcircled{1} \quad S' \subseteq S \in \mathcal{I} \Rightarrow S' \in \mathcal{I}$$

$$\textcircled{2} \quad S, T \in \mathcal{I}, |S| > |T| \Rightarrow \exists e \in S \setminus T \text{ s.t. } T \cup \{e\} \in \mathcal{I}$$

e.g. 線形マトロイド

$$\begin{pmatrix} \boxed{1} & \boxed{1} & \boxed{2} \\ \boxed{2} & \boxed{1} & \boxed{2} \end{pmatrix}$$

$$V = \{\textcircled{1}, \textcircled{2}, \textcircled{3}\}$$

$$\mathcal{I} = \{\emptyset, \{\textcircled{1}\}, \{\textcircled{2}\}, \{\textcircled{3}\}, \{\textcircled{1}, \textcircled{2}\}, \{\textcircled{1}, \textcircled{3}\}\}$$

マトロイドとは？ - 線形独立性の一般化

Def.

有限集合 V とその空でない部分集合族 $\mathcal{I} \subseteq 2^V$ s.t.

$$\textcircled{1} \quad S' \subseteq S \in \mathcal{I} \Rightarrow S' \in \mathcal{I}$$

$$\textcircled{2} \quad S, T \in \mathcal{I}, |S| > |T| \Rightarrow \exists e \in S \setminus T \text{ s.t. } T \cup \{e\} \in \mathcal{I}$$

e.g. 線形マトロイド

$S \in \mathcal{I} : S$ は独立

$$\begin{pmatrix} \boxed{1} & \boxed{1} & \boxed{2} \\ \boxed{2} & \boxed{1} & \boxed{2} \end{pmatrix}$$

$$V = \{\textcircled{1}, \textcircled{2}, \textcircled{3}\}$$

$$\mathcal{I} = \{\emptyset, \{\textcircled{1}\}, \{\textcircled{2}\}, \{\textcircled{3}\}, \{\textcircled{1}, \textcircled{2}\}, \{\textcircled{1}, \textcircled{3}\}\}$$

マトロイドとは？ - 線形独立性の一般化

Def.

有限集合 V とその空でない部分集合族 $\mathcal{I} \subseteq 2^V$ s.t.

$$\textcircled{1} \quad S' \subseteq S \in \mathcal{I} \Rightarrow S' \in \mathcal{I}$$

$$\textcircled{2} \quad S, T \in \mathcal{I}, |S| > |T| \Rightarrow \exists e \in S \setminus T \text{ s.t. } T \cup \{e\} \in \mathcal{I}$$

e.g. • 線形マトロイド

$$\begin{pmatrix} \boxed{1} & \boxed{1} & \boxed{2} \\ \boxed{2} & \boxed{1} & \boxed{2} \end{pmatrix}$$

$$V = \{\textcircled{1}, \textcircled{2}, \textcircled{3}\}$$

$$\mathcal{I} = \{\emptyset, \{\textcircled{1}\}, \{\textcircled{2}\}, \{\textcircled{3}\}, \{\textcircled{1}, \textcircled{2}\}, \{\textcircled{1}, \textcircled{3}\}\}$$

- (可割)マトロイド
- ラミナ-マトロイド
- 閉包的マトロイド
- sparse paving マトロイド
etc...

マトロイド交叉問題とは？

入力: 2つのマトロイド $M_1 = (V, I_1)$, $M_2 = (V, I_2)$

出力: 最大サイズの **共通独立集合** $S \in I_1 \cap I_2$

マトロイド交叉問題とは？

— 二部グラフの最大マッチング問題の一般化

入力: 2つのマトロイド $M_1 = (V, I_1)$, $M_2 = (V, I_2)$

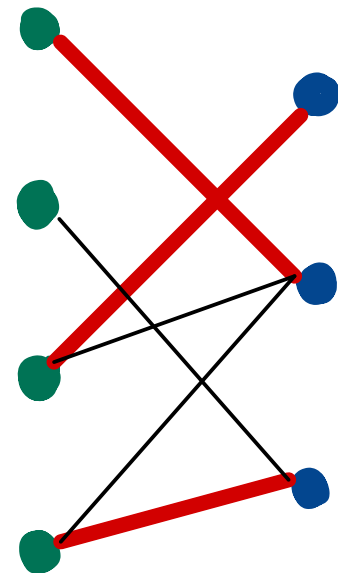
出力: 最大サイズの共通独立集合 $S \in I_1 \cap I_2$

eg. 二部グラフの最大マッチング

V = 辺集合

I_1 = 左側の各頂点に接する辺は高々1本

I_2 = 右側の各頂点に接する辺は高々1本



マトロイド交叉問題とは？

— 二部グラフの最大マッチング問題の一般化

入力: 2つのマトロイド $M_1 = (V, I_1)$, $M_2 = (V, I_2)$

出力: 最大サイズの共通独立集合 $S \in I_1 \cap I_2$

なぜ、本問題は重要か？

マトロイド交差問題とは？

— 二部グラフの最大マッチング問題の一般化

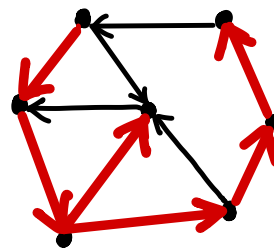
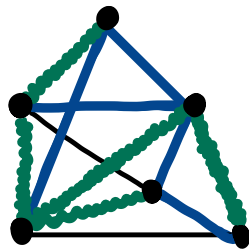
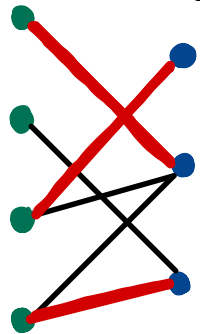
入力: 2つのマトロイド $M_1 = (V, I_1)$, $M_2 = (V, I_2)$

出力: 最大サイズの共通独立集合 $S \in I_1 \cap I_2$

なぜ、本問題は重要か？

① 多くの問題の拡張

二部グラフのマッチング, 全域木の読み込み, 有向全域木



マトロイド交差問題とは？

— 二部グラフの最大マッチング問題の一般化

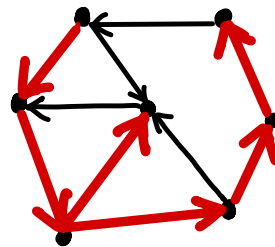
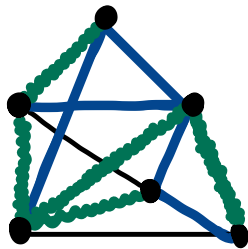
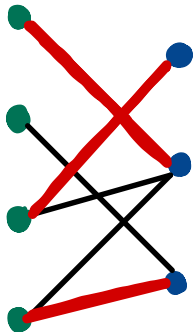
入力: 2つのマトロイド $M_1 = (V, I_1)$, $M_2 = (V, I_2)$

出力: 最大サイズの共通独立集合 $S \in I_1 \cap I_2$

なぜ、本問題は重要か？

① 多くの問題の拡張

二部グラフのマッチング, 全域木の誘因, 有向全域木



② 組合せ最適化への応用

- ・ 電気回路解析
- ・ ネットワーク符号化

$$\lambda() : (V, I_1), (V, I_2)$$
$$\max |S| \text{ s.t. } S \in I_1 \cap I_2$$

ストロイド交叉問題は

多項式時間で解ける

[Edmonds'70, Azhmer-Dowling'71, Lawler'75]

$$\lambda(I) : (V, I_1), (V, I_2)$$

$$\max |S| \text{ s.t. } S \in I_1 \cap I_2$$

マトロイド交叉問題は

多項式時間で解ける

[Edmonds'70, Azhmer-Dowling'72, Lawler'75]

マトロイドは独立性オクルールと与えられる

クエリ(質問)

回答

$S \in I$ か?

独立性
オクルール

Yes or No

$$\lambda \cap: (V, I_1), (V, I_2)$$

$$\max |S| \text{ s.t. } S \in I_1 \cap I_2$$

マトロイド交叉問題は

多項式回のクエリで解ける

[Edmonds'70, Agrver-Dowling'72, Lawler'75]

マトロイドは独立性オラクルと与えられる

クエリ(質問)

回答

$S \in I$ か? $\rightarrow$ 独立性
オラクル $\rightarrow$ Yes or No

計算時間 = オラクルへのクエリ回数

ストロイド交叉問題の 時間計算量

$\lambda(): (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$

独立性を
の **ワイルドカード**

ストロイド交叉問題の 時間計算量

$\lambda(): (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$

独立性の
への **ワイルド**回数

1970s
1986

Edmonds, Lawler, Aigner-Dowling
Cunningham

$O(nr^2)$
 $O(nr^{3/2})$

ストロブ交叉問題の 時間計算量

$\lambda(): (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$

独立性の
 への **ワイルドカード**

1970s	Edmonds, Lawler, Aigner-Dowling	$O(nr^2)$
1986	Cunningham	$O(nr^{3/2})$
2015	Lee-Sidford-Wong	$\tilde{O}(n^2)$
2019	Nguyen, Chakraborty-Lee-Sidford-Singla-Wong	$\tilde{O}(nr)$
2021	Blikstad - v.d. Brand - Mukhopadhyay - Namngkai	$\tilde{O}(n^{9/5})$
2021	Blikstad *	$\tilde{O}(nr^{3/4})$

* [Blikstad21] は乱択を許せば $\tilde{O}(nr^{3/4})$, 決定的時間は $\tilde{O}(nr^{5/6})$

今日のお話: 準線形時間 アルゴリズム

入力のサイズ n の ほぼ線形時間 $O(n \cdot \text{polylog}(n))$ の アルゴリズム

今日のお話: 準線形時間 アルゴリズム

入力のサイズ n の ほぼ線形時間 $O(n \cdot \text{polylog}(n))$ の アルゴリズム

★ 厳密解ではなく、近似解を目指す

α 近似解 $S \in \mathcal{I}_1 \cap \mathcal{I}_2$

s.t. $|S| \geq \alpha \cdot \text{OPT}$

今日のお話: 準線形時間 アルゴリズム

入力のサイズ n の ほぼ線形時間 $O(n \cdot \text{polylog}(n))$ の アルゴリズム

★ 厳密解ではなく、近似解を目指す

α 近似解 $S \in \mathcal{I}_1 \cap \mathcal{I}_2$
s.t. $|S| \geq \alpha \cdot \text{OPT}$

e.g.

sparse cut : Khandekar-Rao-Vazirani'09, Madry'10

k -cut : Quanrud'19

k -ECSS : Chalermsook-Huang-Narasimha-Saranurak-Sukprasert-Yingchareonthawornchai'22

Weighted matching : Preis'99, Vinkemeier-Hougardy'03, Duan-Pettie'14, Zhang-Henzinger'23

ストロイド交叉問題の

準線形時間アルゴリズム

$\lambda(): (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 α -approx.: find $S \in I_1 \cap I_2$
s.t. $|S| \geq \alpha \cdot r$

Folklore

$\frac{1}{2}$ 近似

$O(n) \text{ TI'}$

ストロイド交叉問題の

準線形時間アルゴリズム

$\lambda(): (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1, I_2} |S|$
 α -approx.: find $S \in I_1, I_2$
st. $|S| \geq \alpha \cdot r$

Folklore

$\frac{1}{2}$ 近似

$O(n)$ 時間

Guruganesh-Singla '17

$\frac{1}{2} + 10^{-4}$ 近似

$O(n)$ 時間

2-ロード交叉問題の

準線形時間アルゴリズム

$\lambda(): (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1, I_2} |S|$
 α -approx.: find $S \in I_1, I_2$
s.t. $|S| \geq \alpha \cdot r$

Folklore	$\frac{1}{2}$ 近似的	$O(n) \text{ 時間}$
Guruganesh-Singla '17	$\frac{1}{2} + 10^{-4}$ 近似的	$O(n) \text{ 時間}$
Quarmrud '24	$1 - \epsilon$ 近似的	$\tilde{O}_\epsilon(n + r^{3/2}) \text{ 時間}$
Blikstad-Tu '25 *	$1 - \epsilon$ 近似的	$\tilde{O}_\epsilon(n) \text{ 時間}$

* arXiv に投稿したのが本研究とほぼ同時

2-ロード交叉問題の

準線形時間アルゴリズム

$\lambda \cap: (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 α -approx.: find $S \in I_1 \cap I_2$
s.t. $|S| \geq \alpha \cdot r$

Folklore

$\frac{1}{2}$ 近似

$O(n)$ 時間

最近の研究は全て乱択アルゴリズム!

Blikstad - Tu'25*

$1 - \epsilon$ 近似

$\tilde{O}_\epsilon(n)$ 時間

Q. 決定的には？

乱数  を使わないアルゴリズム

Q. 決定的には？

$\lambda(): (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 α -approx.: find $S \in I_1 \cap I_2$
s.t. $|S| \geq \alpha \cdot r$

乱数  を使わないアルゴリズム

自明な $O(n)$ ワイ) の $\frac{1}{2}$ 近似アルゴリズム しか知られていない。

Q. 決定的には？

$\lambda(): (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 α -approx.: find $S \in I_1 \cap I_2$
s.t. $|S| \geq \alpha \cdot r$

乱数  を使わないアルゴリズム

自明な $O(n) \text{ TIME}$ の $\frac{1}{2}$ 近似アルゴリズム しか知られていない。

貪欲アルゴリズム

```
S ← ∅  
for v ∈ V do  
  | if S ∪ {v} ∈ I1 ∩ I2 then  
  |   L S ← S ∪ {v}  
return S
```

Q. 決定的には？

$\lambda(): (V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 α -approx.: find $S \in I_1 \cap I_2$
s.t. $|S| \geq \alpha \cdot r$

乱数  を使わないアルゴリズム

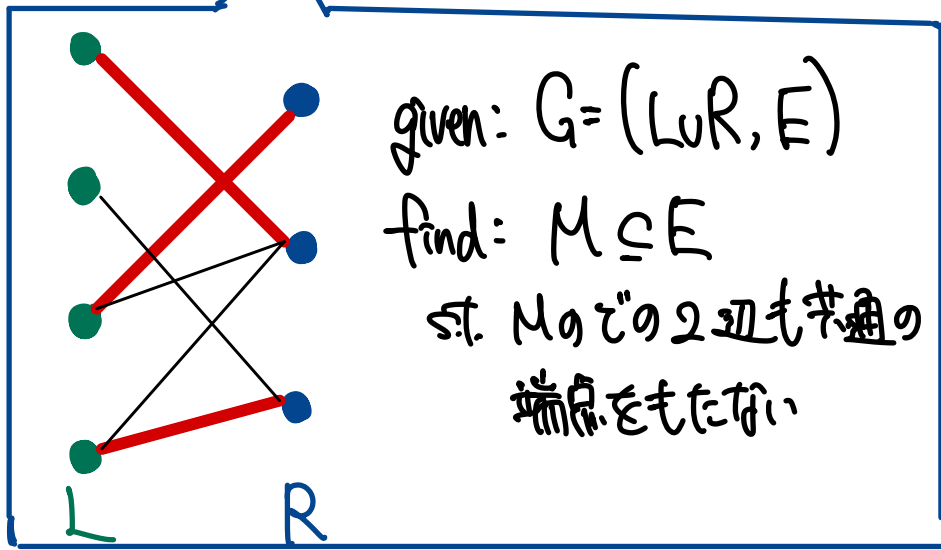
自明な $O(n) \text{ TI}$ の $\frac{1}{2}$ 近似アルゴリズム は知られていない。

主結果

$O(\frac{n}{\epsilon} + r \log r) \text{ TI}$ の $(\frac{2}{3} - \epsilon)$ 近似アルゴリズム

アイティ: (二部) マッチングからの類推

(二部) マッチング

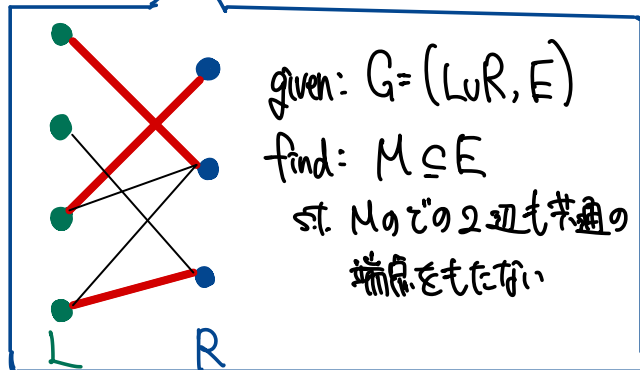


ショートカット

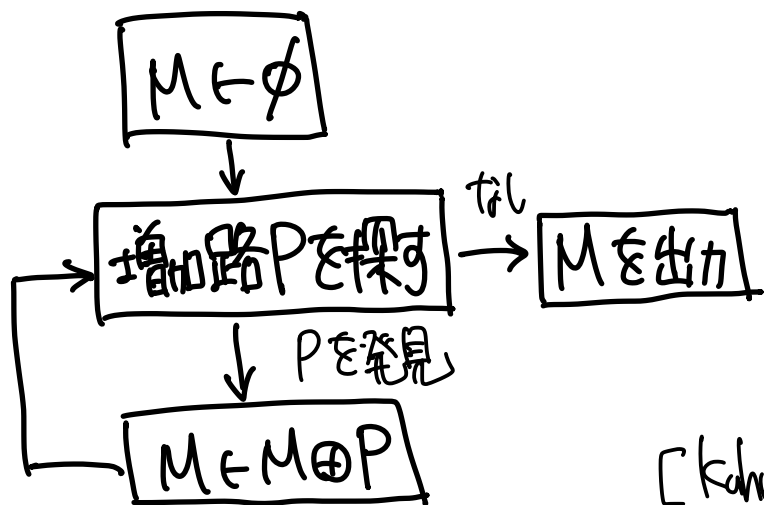
input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

アイティ: (二部)マッチングの類推

(二部) マッチング



アルゴリズム



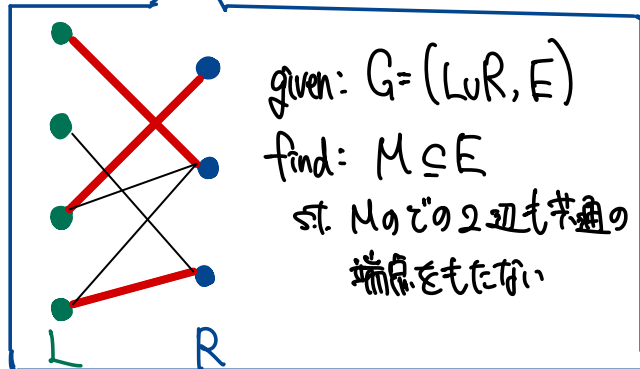
[Kahn '55]
 [Edmonds '65]

ショートカット

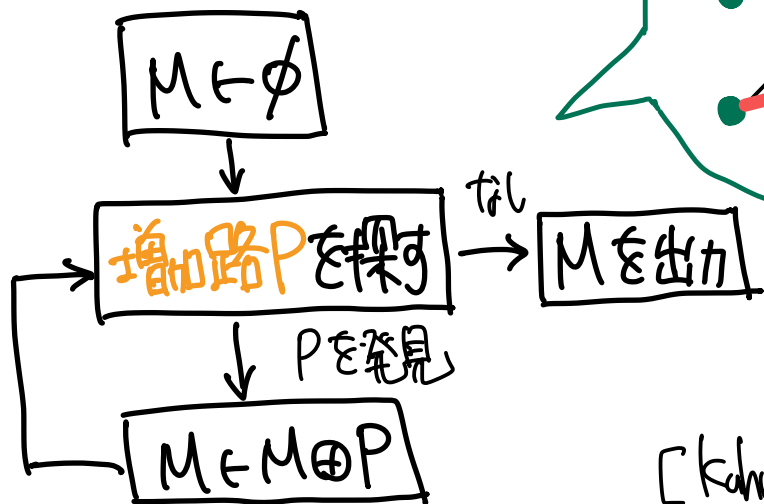
input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

アイティ: (二部)マッチングからの類推

(二部) マッチング



アルゴリズム



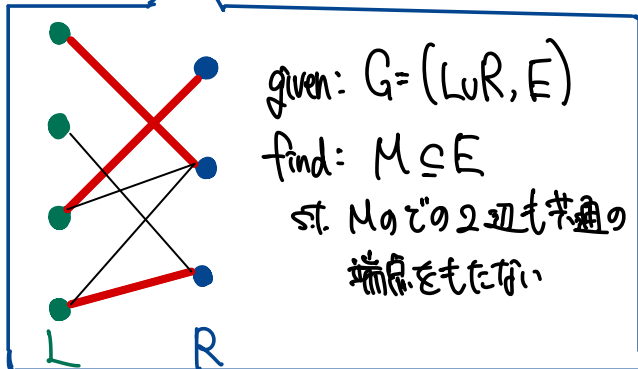
[Kuhn '55]
 [Edmonds '65]

ショートカット

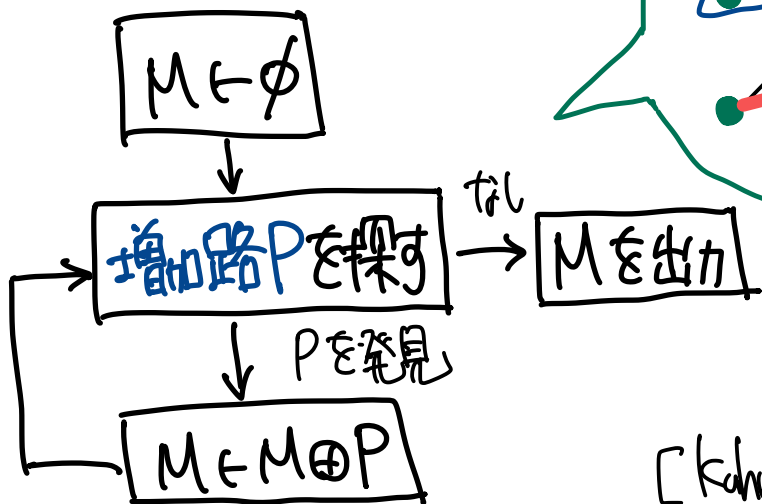
input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

アイティ: (二部) マッチングからの類推

(二部) マッチング



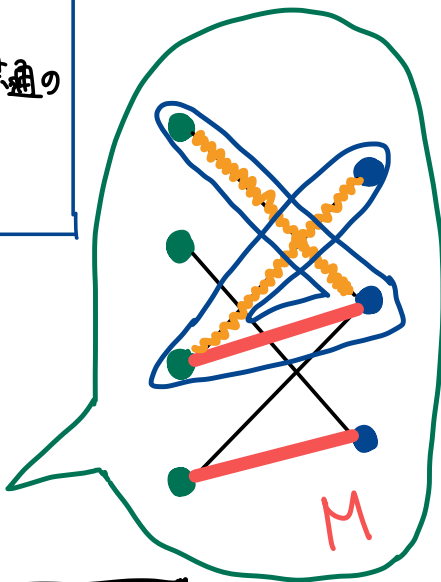
アルゴリズム



[Kuhn '55]
 [Edmonds '65]

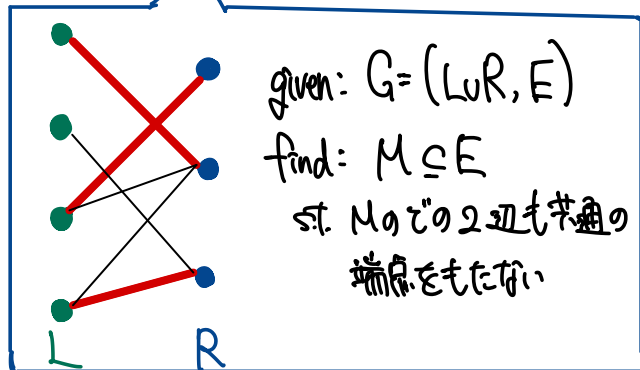
ショートカット

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

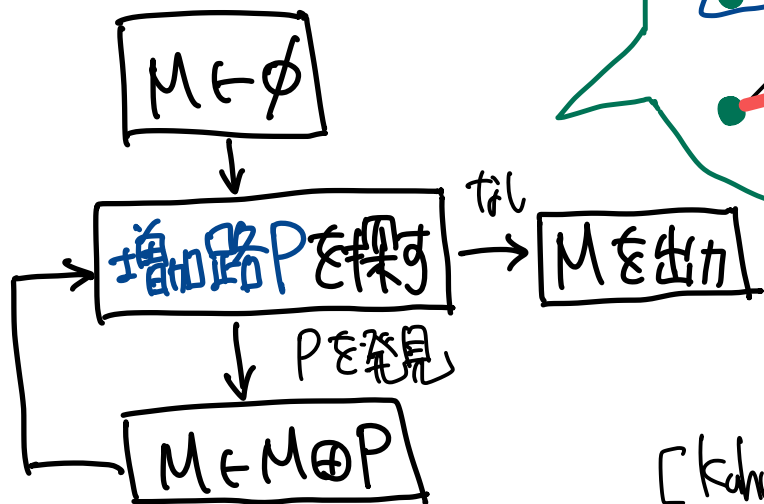


アイティ: (二部) マッチングの類推

(二部) マッチング



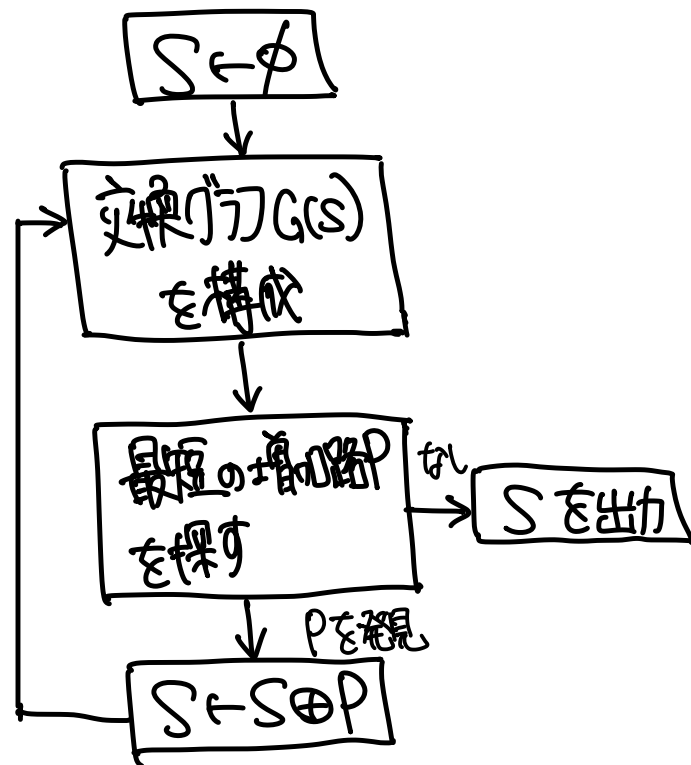
アルゴリズム



[Kuhn '55]
 [Edmonds '65]

ショートカット

アルゴリズム

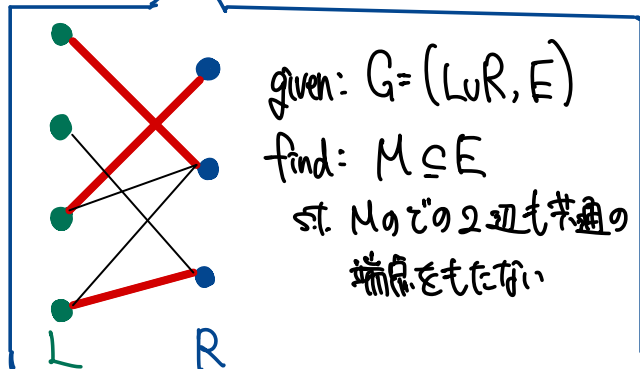


[Edmonds '70]

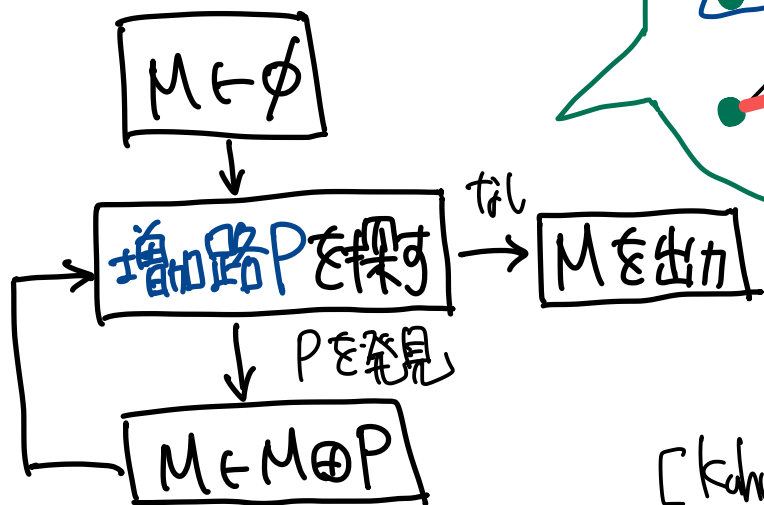
input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

アイティ: (二部)マッチングからの類推

(二部) マッチング



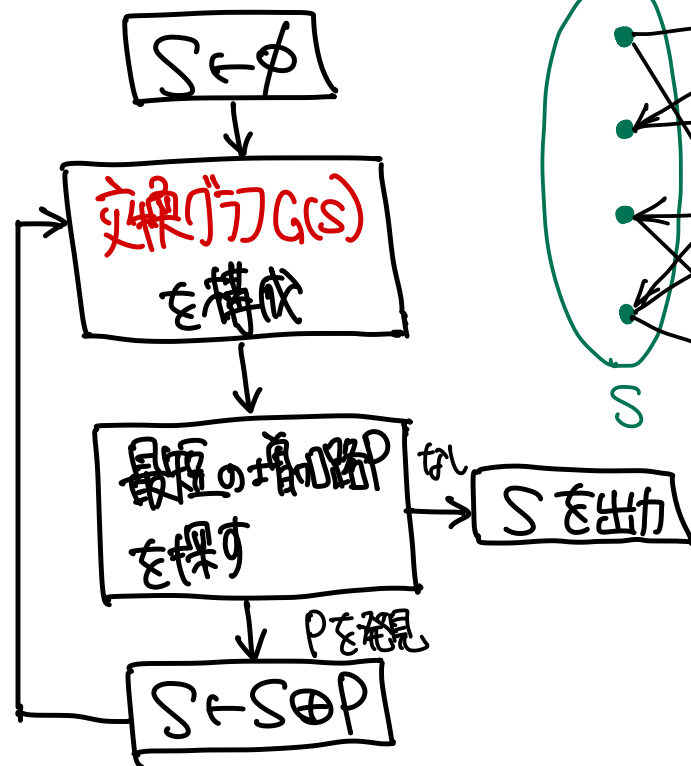
アルゴリズム



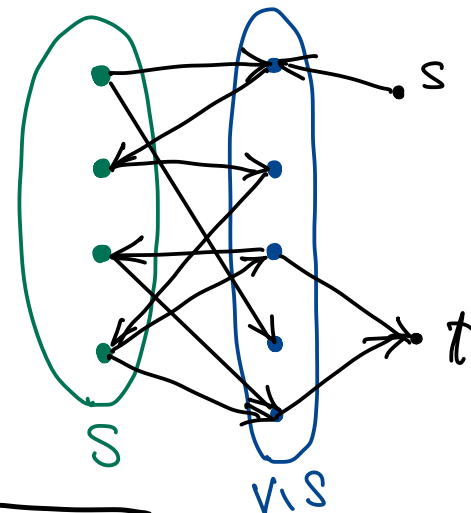
[Kuhn '55]
 [Edmonds '65]

ショートカット

アルゴリズム



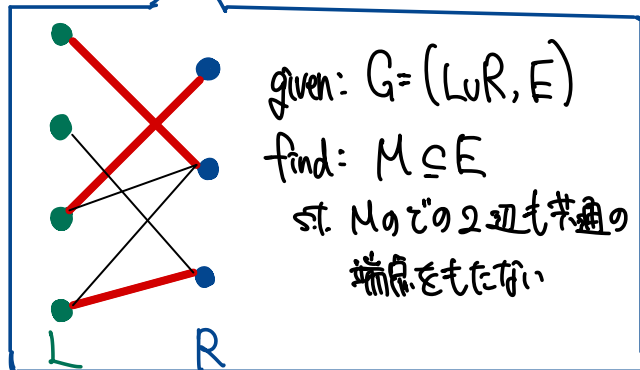
input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries



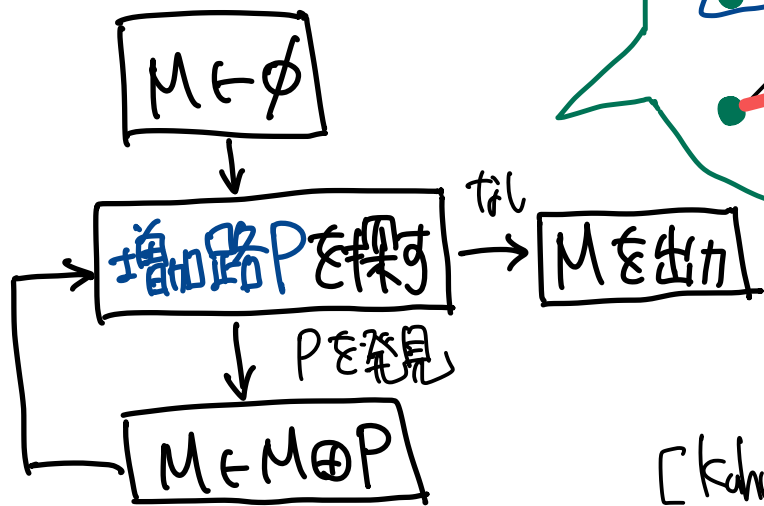
[Edmonds '70]

アイティ: (二部)マッチングの類推

(二部) マッチング



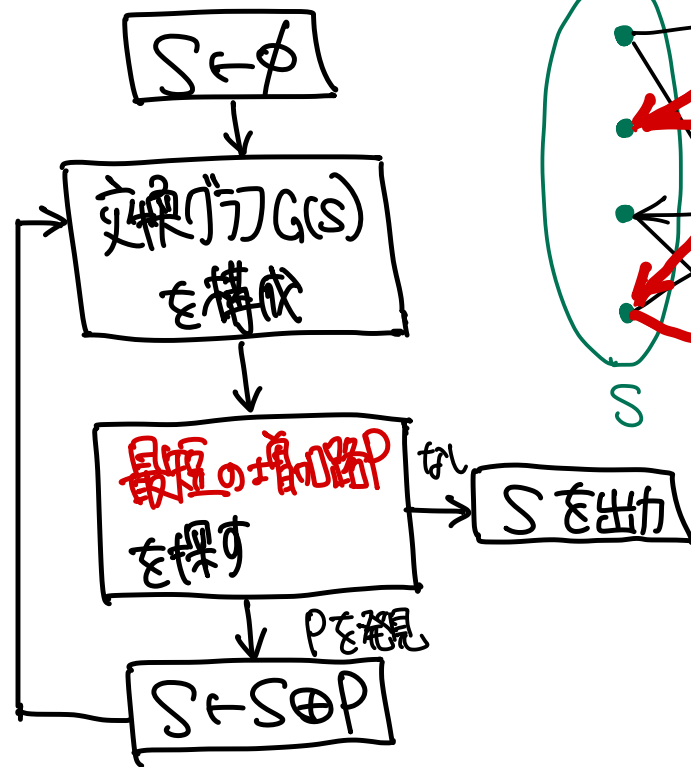
アルゴリズム



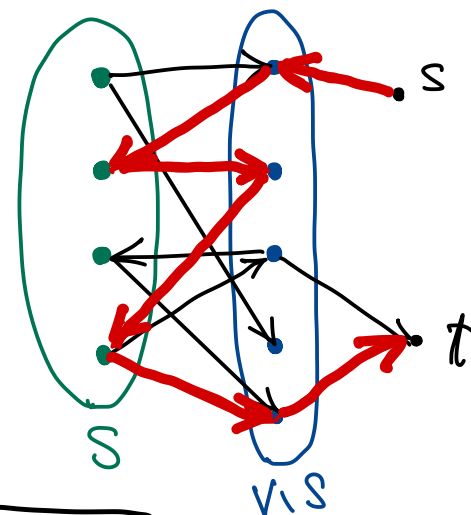
[Kuhn '55]
 [Edmonds '65]

ショートカット

アルゴリズム



input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries



[Edmonds '70]

アイティ: (二部) スパチ=グ からの 類推

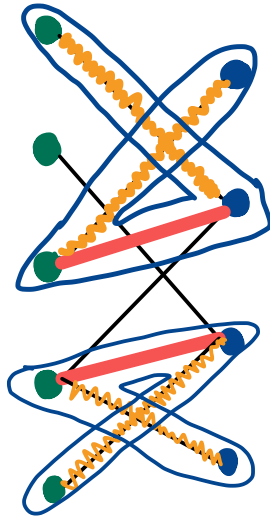
(二部) スパチ=グ

高速なアルゴリズム

★ 最短の増幅路をきとめてみる

- ジョッキ=グジョー

[Hopcroft-Karp '73]



ショートカット

input: $(V, I_1), (V, I_2)$

$n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$

find $S \in I_1 \cap I_2$

s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$

using $O(n/\epsilon + r \log r)$ queries

アイティ: (二部) スパチ=グ からの類推

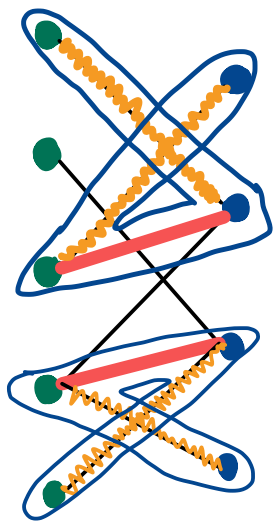
(二部) スパチ=グ

高速なアルゴリズム

★ 最短の増大路をきとめてみる

- ジョッキ=グジョー

[Hopcroft-Karp '73]



- 点素, 極大な増大路の集合をとればOK
- BFS+DFS

ショートカット

input: $(V, I_1), (V, I_2)$

$n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$

find $S \in I_1 \cap I_2$

s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$

using $O(n/\epsilon + r \log r)$ queries

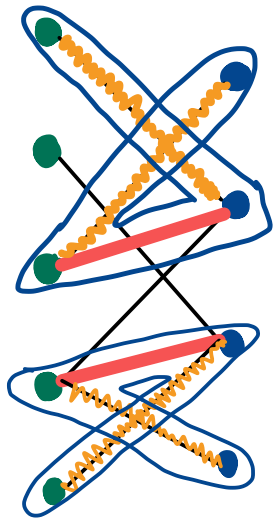
アイティ: (二部) スパチ=グ からの 類推

(二部) スパチ=グ

高速なアルゴリズム

- ★ 最短の増道路をきとめてみる
- ジョッキ=グジョー

[Hopcroft-Karp '73]



- 点素々、極大な増道路の集合をとればOK
- BFS+DFS

ショートカット

高速なアルゴリズム

- ★ 最短の増道路をきとめてみる
- ジョッキ=グジョー

[Cunningham '86]

- 点素々、極大な増道路の集合をとればOK?

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
find $S \in I_1 \cap I_2$
s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
using $O(n/\epsilon + r \log r)$ queries

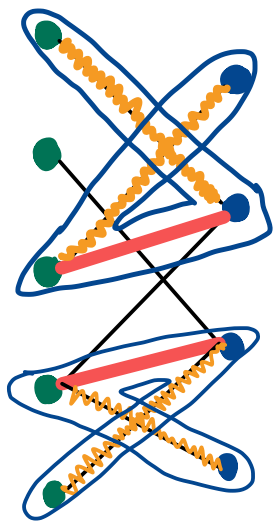
アイT": (二部) スパチ=グ"からの類推

(二部) スパチ=グ"

高速なアルゴリズム

- ★ 最短の増幅路をきとめてみる
- ジョット=グ"ジョー

[Hopcroft-Karp '73]



- 点素か、極大な増幅路の集合をきとればOK
- BFS+DFS

ショート交又

高速なアルゴリズム

- ★ 最短の増幅路をきとめてみる
- ジョット=グ"ジョー

[Cunningham '86]

- 点素か、~~極大な増幅路の集合をきとればOK~~
- BFS+DFS

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
find $S \in I_1 \cap I_2$
s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot n$
using $O(n/\epsilon + r \log r)$ queries

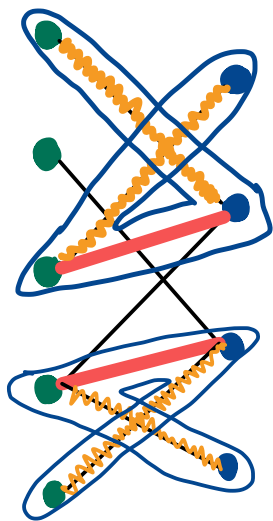
アイT": (二部) スパチ=グからの類推

(二部) スパチ=グ

高速なアルゴリズム

- ★ 最短の増道路をきとめてみる
- ジョッキ=グジョー

[Hopcroft-Karp '73]



- 点素々、極大な増道路の集をえればOK
- BFS + DFS

ショート交又

高速なアルゴリズム

- ★ 最短の増道路をきとめてみる
- ジョッキ=グジョー

[Cunningham '86]

~~- 点素々、極大な増道路の集をえればOK~~



増路集合をみつければOK [CLSW '19]

$\Pi_0 := (B_1, A_1, B_2, \dots, B_{k+1})$

- | | |
|--|-------------------------------|
| ① $A_k \subseteq D_{2k}, B_k \subseteq D_{2k-1}$ | ④ $S + B_{k+1} \in I_2$ |
| ② $ B_i = A_i = \dots = B_{k+1} $ | ⑤ $S - A_k + B_{k+1} \in I_1$ |
| ③ $S + B_1 \in I_1$ | ⑥ $S - A_k + B_k \in I_2$ |

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

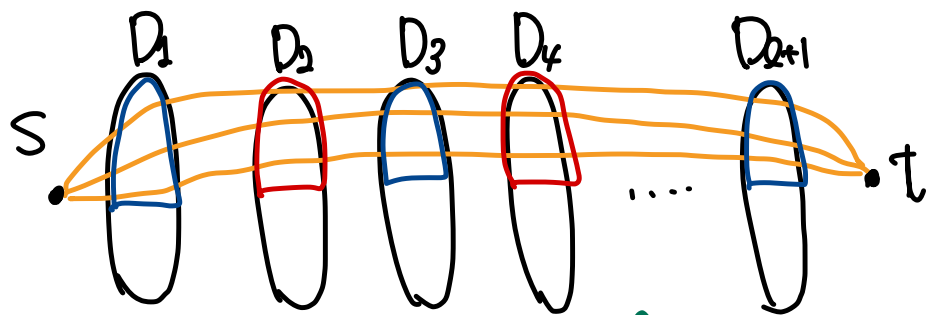
増加集合 とは？

2トロイド交差

★ フロット・ジョー [Cunningham'86]

- 最短の増加路を まよめ (みよめ)

複数本の増加路



点素な増加路を2つだけでは ~~ダメ~~

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
find $S \in I_1 \cap I_2$
s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
using $O(n/\epsilon + r \log r)$ queries

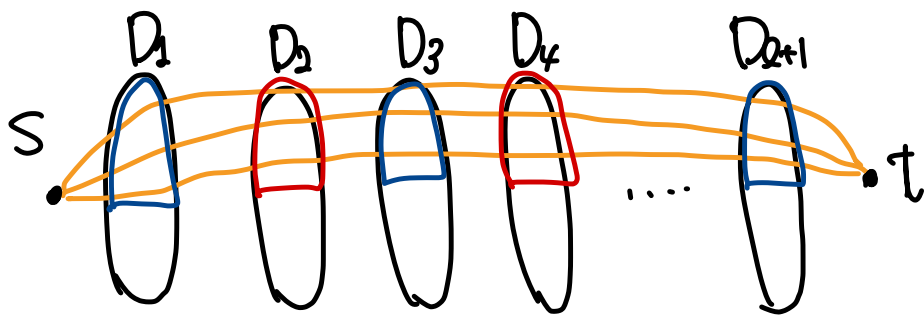
増加集合 とは？

2トroid交叉

★ フロット・ジョー [Cunningham'86]

- 最短の増加路を まよめ (みつけ)

複数本の増加路



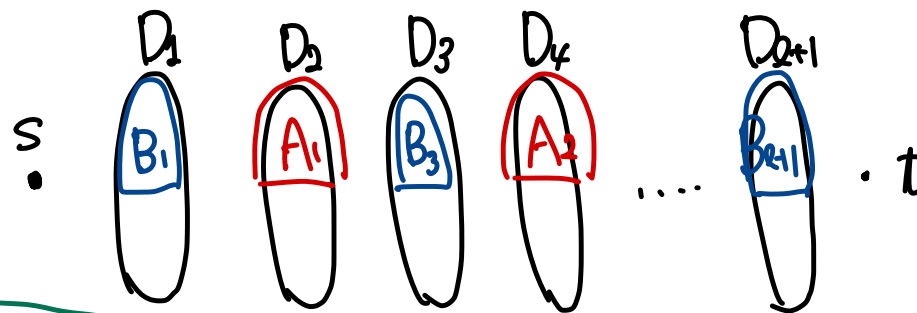
点素な増加路を2つだけではダメ

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

★ 増加集合 [CLSSW'19]

$\Pi_k := (B_1, A_1, B_2, \dots, B_{k+1})$

- ① $A_k \subseteq D_{k+1}, B_k \subseteq D_{k+1}$
- ② $|B_1| = |A_1| = \dots = |B_{k+1}|$
- ③ $S + B_1 \in I_1$
- ④ $S + B_{k+1} \in I_2$
- ⑤ $S - A_k + B_{k+1} \in I_1$
- ⑥ $S - A_k + B_k \in I_2$



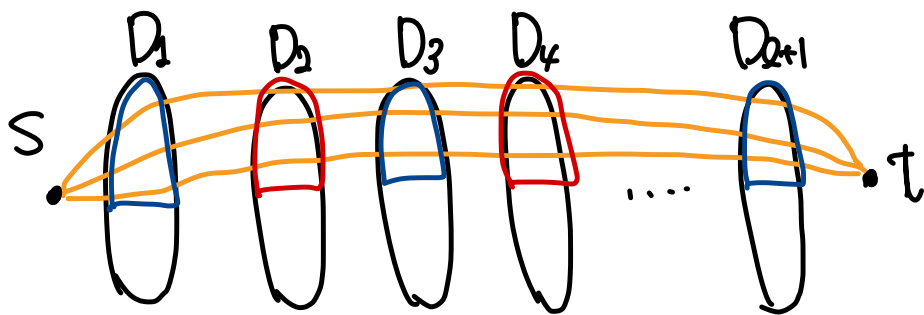
増加集合 とは？

2トroid交叉

★ フロット・ジョー [Cunningham'86]

- 最短の増加路を まよめ (みつけ)

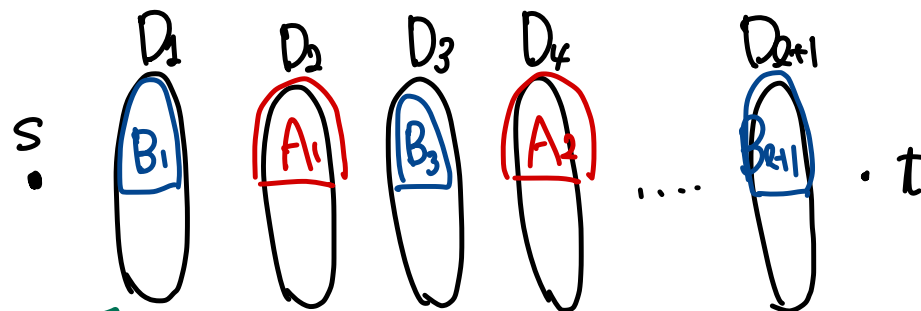
複数本の増加路



★ 増加集合 [CLSSW'19]

$\Pi_k := (B_1, A_1, B_2, \dots, B_{k+1})$

- ① $A_k \subseteq D_{k+1}, B_k \subseteq D_{k+1}$
- ② $|B_1| = |A_1| = \dots = |B_{k+1}|$
- ③ $S + B_1 \in I_1$
- ④ $S + B_{k+1} \in I_2$
- ⑤ $S - A_k + B_{k+1} \in I_1$
- ⑥ $S - A_k + B_k \in I_2$



各集合には対応がある！

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

アイT": (二部) スキチ=V の 類推

(二部) スキチ=V

M に長さ 3 の増加路が存在するか

$\Rightarrow M$ は $\frac{2}{3}$ 近似

ショートカット

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
find $S \in I_1 \cap I_2$
s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
using $O(n/\epsilon + r \log r)$ queries

アイティ: (二部)マッチングの類推

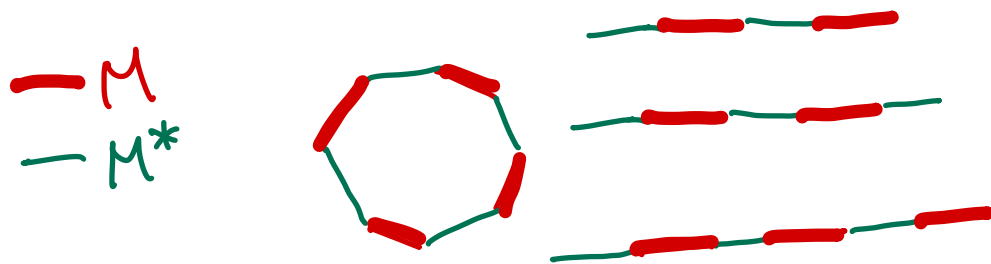
(二部) マッチング

ショートカット

M に長さ 3 の増道路が存在しない

$\Rightarrow M$ は $\frac{2}{3}$ 近似

⊙ 最大マッチング M^* をもてきたとき



$M \Delta M^*$ は長さ 4 以上の交互道 or 交互サイクルに分解できる

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

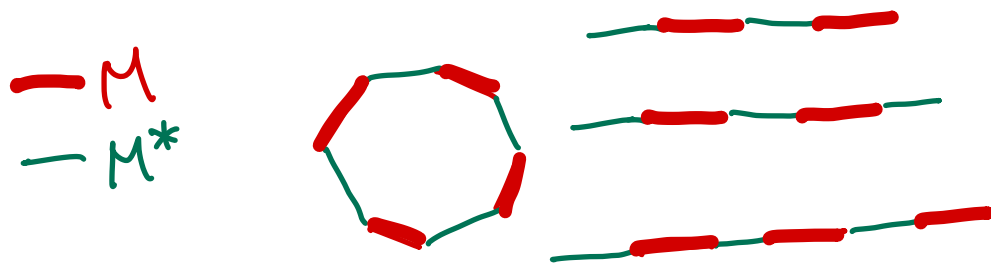
アイティ: (二部)マッチングの類推

(二部) マッチング

M に長さ 3 の増加路が存在しない

$\Rightarrow M$ は $\frac{2}{3}$ 近似

⊙ 最大マッチング M^* をもてきたとき



$M \Delta M^*$ は長さ 4 以上の交互道 or 交互サイクル
に分解できる

ショートカット

$G(S)$ に長さ 4 の増加路が存在しない

$\Rightarrow S$ は $\frac{2}{3}$ 近似

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

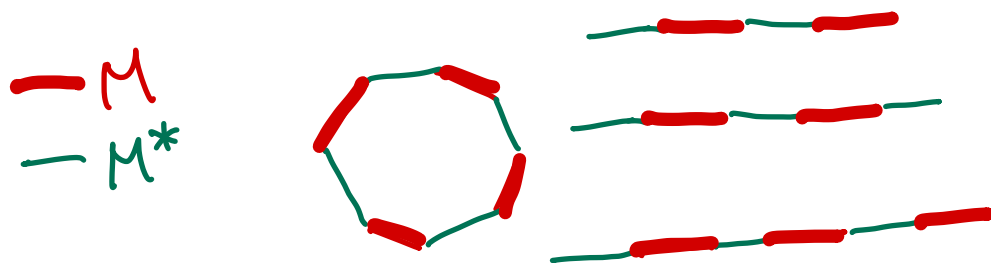
アイティ: (二部) スパッチング の 類推

(二部) スパッチング

M に長さ 3 の増加路が存在しない

$\Rightarrow M$ は $\frac{2}{3}$ 近似

⊙ 最大スパッチング M^* をもてきたとき



$M \Delta M^*$ は長さ 4 以上の交互道 or 交互サイクル
に分解できる

ショートカット

$G(S)$ に長さ 4 の増加路が存在しない

$\Rightarrow S$ は $\frac{2}{3}$ 近似

本研究における反例

[Bliksrud'21] で、長さ 4 の増加路に対する
増加分集合をみつけるアルゴリズムを途中で止める

[Bliksrud'21] では $O(\sqrt{n})$ 回



本研究では $O(1/\epsilon)$ 回

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

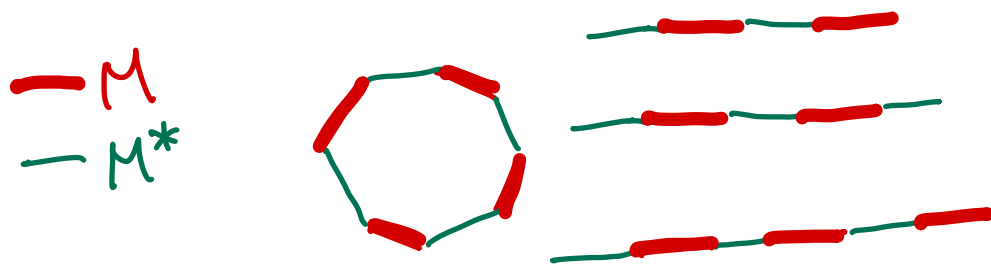
アイティ: (二部) スパッチング からの類推

(二部) スパッチング

M に長さ 3 の増加路が存在しない

$\Rightarrow M$ は $\frac{2}{3}$ 近似

⊙ 最大スパッチング M^* をもてきたとき



$M \Delta M^*$ は長さ 4 以上の交互道 or 交互サイクルに分解できる

ショートカット

$G(S)$ に長さ 4 の増加路が存在しない

$\Rightarrow S$ は $\frac{2}{3}$ 近似

本研究における反例

[Blkstad'21] で、長さ 4 の増加路に対する増加集合をみつけるアルゴリズムを途中で止める

ほぼ全ての長さ 4 の増加路をみつけられる

残りの本数 $\leq \epsilon r$

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

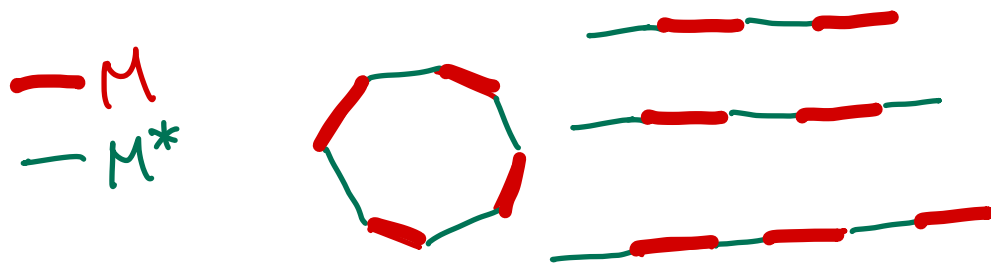
アイティ: (二部) スパチ=グ からの類推

(二部) スパチ=グ

M に長さ 3 の増加路が存在しない

$\Rightarrow M$ は $\frac{2}{3}$ 近似

⊙ 最大スパチ=グ M^* をもてきたとき



$M \Delta M^*$ は長さ 4 以上の交互道 or 交互サイクル
に分解できる

ショート交又

$G(S)$ に長さ 4 の増加路が存在しない

$\Rightarrow S$ は $\frac{2}{3}$ 近似

本研究における反例

[Blkstad'21] で、長さ 4 の増加路に対応する
増加集合をみつけるアルゴリズムを途中で止める

ほぼ全ての長さ 4 の増加路をみつけられる

$(\frac{2}{3} - \epsilon)$ -近似

input: $(V, I_1), (V, I_2)$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 find $S \in I_1 \cap I_2$
 s.t. $|S| \geq (\frac{2}{3} - \epsilon) \cdot r$
 using $O(n/\epsilon + r \log r)$ queries

本結果の予想外に面白いポイント

本アルゴリズムは ストリーミングアルゴリズム
に拡張できる！

本結果の予想外に面白いポイント

本アルゴリズムは ストリーミングアルゴリズム
に拡張できる!

メモリよりも小さい空間計算量で問題を解く
(メモリ)

本結果の予想外に面白いポイント

本アルゴリズムは ストリーミングアルゴリズム
に拡張できる!

メモリよりも小さい空間計算量で問題を解く
(メモリ)

$O(n)$

計算機

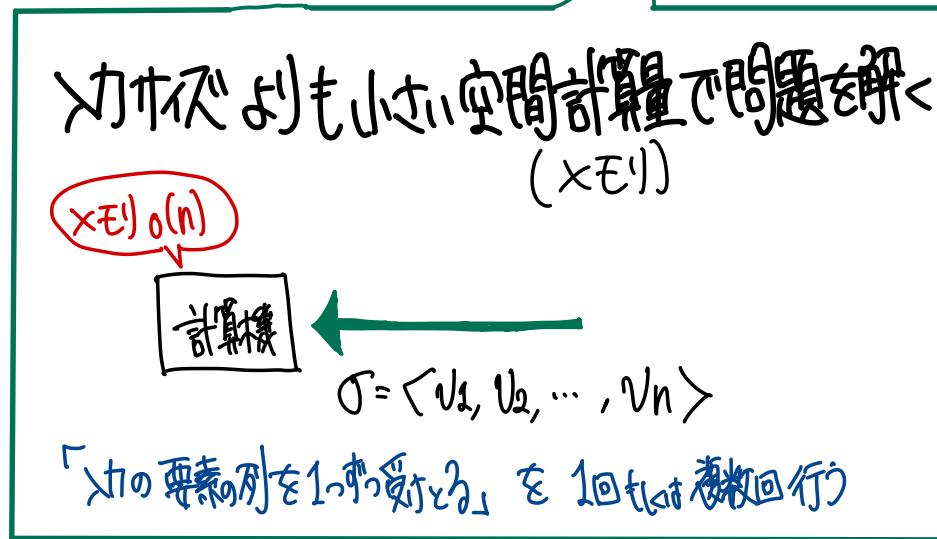


$\sigma = \langle v_1, v_2, \dots, v_n \rangle$

「 σ の要素列を1つずつ受ける」を1回も複数回行う

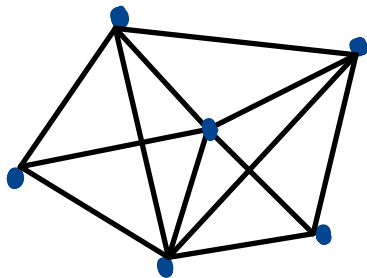
本結果の予想外に面白いポイント

本アルゴリズムは **ストリーミングアルゴリズム**
に拡張できる!



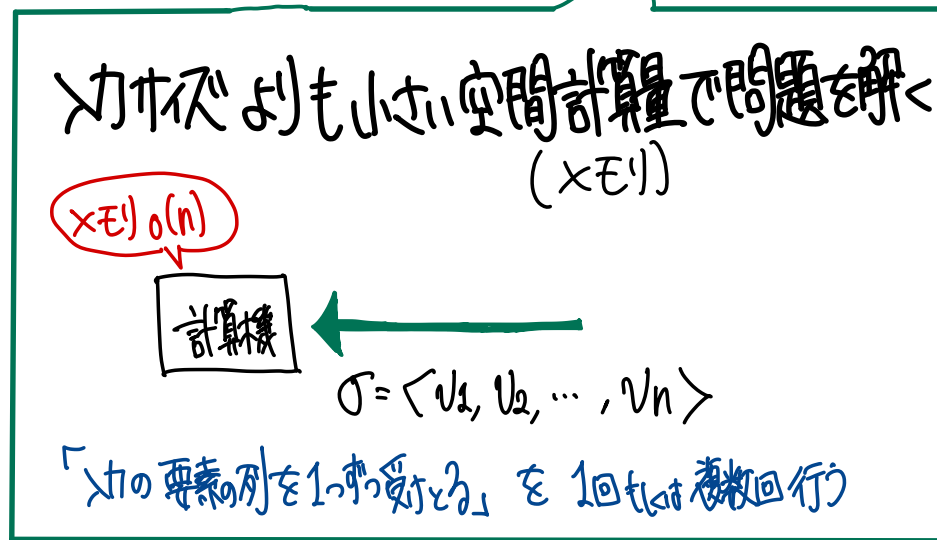
★ 特に、グラフの問題では、**頂点数のオーダーのメモリ**で解く

セミストリーミングアルゴリズムがよく研究



本結果の予想外に面白いポイント

本アルゴリズムは **ストリーミングアルゴリズム**
に拡張できる!



辺数より小

★ 特に、グラフの問題では、**頂点数のオーダーのメモリ**で解く

e.g. **マッチング問題**

セミストリーミングアルゴリズムがよく研究

ストロイド交叉問題

の ストリーミングアルゴリズム

$\lambda(): (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$

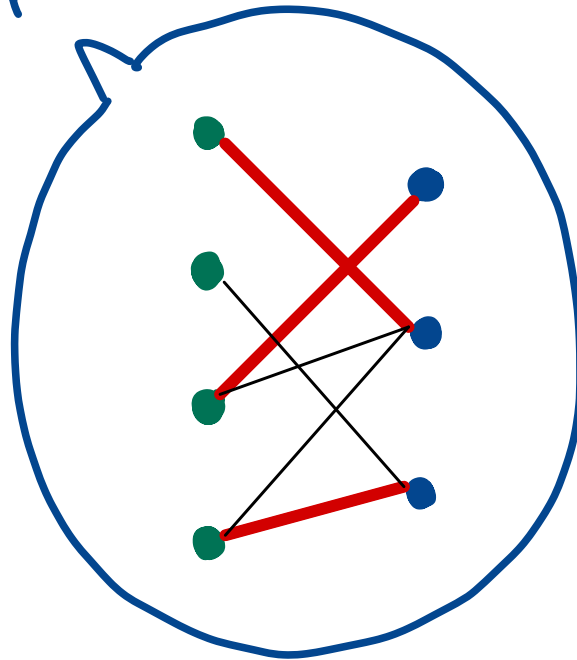
マトロイド交叉問題

の ストリーミング アルゴリズム

マッチング問題の
セミストリーミングアルゴリズム

Paz-Schwartzman'17

Bernstein'20



$\lambda() : (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 $r_i := \max_{S \in I_i} |S| \ (i=1,2)$

マトロイド交叉問題

の ストリーミングアルゴリズム

マッチング問題の
セミストリーミングアルゴリズム

Paz-Schwartzman'17

拡張



Garg-Jordan-Svensson'21

1回のパス, 重みあり, $(1/2 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(n + r_2)$

Bernstein'20



Huang-Sellier'24

1回のパス, $\mathcal{U} = \mathcal{U}' \cup \mathcal{U}''$, $(2/3 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r)$

$\lambda() : (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 $r_i := \max_{S \in I_i} |S| \ (i=1,2)$

マトロイド交叉問題

の ストリーミング アルゴリズム

マトロイド交叉問題の
ストリーミング アルゴリズム
Paz-Schwartzman'17

拡張



Garg-Jordan-Svensson'21

1回のパス, 重みあり, $(1/2 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r_1 + r_2)$

Bernstein'20



Huang-Sellier'24

1回のパス, ラベル付き, $(2/3 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r)$

本研究

$O(1/\epsilon)$ 回のパス, $(2/3 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r_1 + r_2)$

$\lambda(): (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 $r_i := \max_{S \in I_i} |S| \ (i=1,2)$

マトロイド交叉問題

の ストリ-ミング アルゴリズム

マトロイド交叉問題の
セ-ストリ-ミング アルゴリズム

Paz-Schwartzman '17

拡張



Garg-Jordan-Svensson '21

1回のパス, 重みあり, $(1/2 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r_1 + r_2)$

Bernstein '20



Huang-Sellier '24

1回のパス, $\mathcal{L} = \mathcal{L}' \cup \mathcal{L}''$, $(2/3 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r)$

Feigenbaum-Kannan-McGregor
- Suri-Zhang '04



本研究

$O(1/\epsilon)$ 回のパス, $(2/3 - \epsilon)$ 近似, 空間 $\tilde{O}_\epsilon(r_1 + r_2)$

グラフのストリ-ミングの最初の論文

$\lambda \cap := (V, I_1), (V, I_2)$
 $\max |S| \text{ s.t. } S \in I_1 \cap I_2$
 $n := |V|, r := \max_{S \in I_1 \cap I_2} |S|$
 $r_i := \max_{S \in I_i} |S| \text{ (} i=1,2 \text{)}$

結論

- マトロイド交差問題に対し、
決定的な $(2/3 - \epsilon)$ -近似 $\tilde{O}_\epsilon(n)$ のアルゴリズム
を設計
- これを $O(1/\epsilon)$ 回繰り返す $(2/3 - \epsilon)$ -近似 セミストリミナル アルゴリズム
に応用 : FkMSZ'04 の拡張

結論

- マトロイド交差問題に対し、

決定的な $(2/3 - \epsilon)$ -近似 $\tilde{O}_\epsilon(n)$ のアルゴリズム
を設計

- これを $O(1/\epsilon)$ 回繰り返す $(2/3 - \epsilon)$ -近似 セミストリ-ミ-グ アルゴリズム
に適用: FkMSZ'04 の拡張

open questions

- 決定的な $(1 - \epsilon)$ -近似 $\tilde{O}_\epsilon(n)$ のアルゴリズム
- $\text{poly}(1/\epsilon)$ 回繰り返す $(1 - \epsilon)$ -近似 セミストリ-ミ-グ アルゴリズム