マトロイド制約下での劣モジュラ関数最大化に対する高速なアルゴリズム

寺尾 樹哉, 小林 佑輔

京都大学数理解析研究所

最適化の理論とアルゴリズム:未来を担う若手研究者の集い 2024@筑波大学
5月19日(土)

Yusuke Kobayashi, Tatsuya Terao : Subquadratic Submodular Maximization with a General Matroid Constraint,
To appear in **ICALP 2024** (arXiv: 2405.00359)

劣モジュラ関数

<u>定義</u>

$f: 2^V \to \mathbb{R}$ で次の式を満たすもの

$$\boldsymbol{S \subseteq T \subseteq V, v \in V \setminus T \Longrightarrow \underbrace{f(S \cup \{v\}) - f(S)}_{\text{小さい集合の増分}} \geq \underbrace{f(T \cup \{v\}) - f(T)}_{\text{大きい集合の増分}}}$$

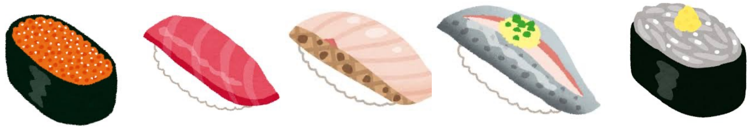
劣モジュラ関数

<u>定義</u>

$f: 2^V \to \mathbb{R}$ で次の式を満たすもの

$$S \subseteq T \subseteq V, v \in V \setminus T \Rightarrow \underbrace{\boxed{f(S \cup \{v\}) - f(S)}}_{\text{小さい集合の増分}} \geq \underbrace{\boxed{f(T \cup \{v\}) - f(T)}}_{\text{大きい集合の増分}}$$

経済学、ゲーム理論の観点から…

$V = \{$  $\}$, $f($  $) =$ うれしさ

劣モジュラ関数

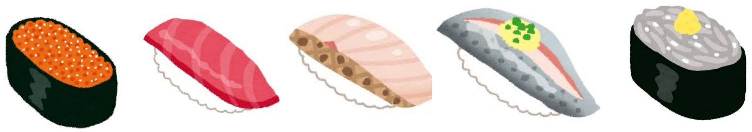
<u>定義</u>

$f: 2^V \to \mathbb{R}$ で次の式を満たすもの

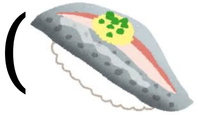
$\boldsymbol{S \subseteq T \subseteq V, v \in V \setminus T \Rightarrow} \boxed{\boldsymbol{f(S \cup \{v\}) - f(S)}} \geq \boxed{\boldsymbol{f(T \cup \{v\}) - f(T)}}$

小さい集合の増分　　大きい集合の増分

経済学、ゲーム理論の観点から…

$V = \{$  $\}$, $\boldsymbol{f}($  $) =$ うれしさ

<u>限界効用逓減性</u>

$f($  $) - f($  $) \geq f($  $) - f($  $)$

劣モジュラ関数

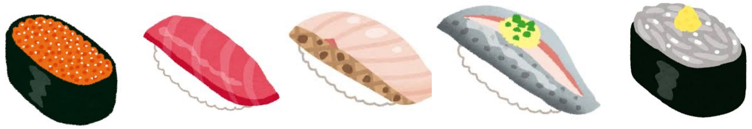
<u>定義</u>

$f: 2^V \to \mathbb{R}$ で次の式を満たすもの

$$\boldsymbol{S \subseteq T \subseteq V, v \in V \setminus T \Rightarrow \boxed{f(S \cup \{v\}) - f(S)} \geq \boxed{f(T \cup \{v\}) - f(T)}}$$

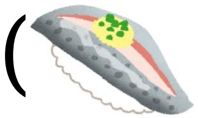
小さい集合の増分　　大きい集合の増分

経済学、ゲーム理論の観点から…

$V = \{$  $\}$, $f($  $) =$ うれしさ

<u>限界効用逓減性</u>

不可分財の効用を表現するのに適している！

$f($  $) - f($  $) \geq f($  $) - f($  $)$

単調劣モジュラ関数最大化

$f(S) \leq f(T) \ (S \subseteq T)$

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ, $f(\emptyset) = 0$

$$\max \quad f(S) \quad \text{s.t.} \quad S \in \mathcal{C}$$

$|S| \leq r$ など

単調劣モジュラ関数最大化

$f(S) \le f(T) \quad (S \subseteq T)$

$f: 2^V \to \mathbb{R}_{\geq 0}$: **非負単調劣モジュラ**, $f(\emptyset) = 0$

$$\max \quad f(S) \quad \text{s.t.} \quad S \in \mathcal{C}$$

$|S| \le r$ など

- 多くの問題の一般化
  例）最大被覆問題、施設配置問題
- 非常に多くの分野に実応用
  例）機械学習、コンピュータビジョン、経済学

サイズ制約下での単調劣モ最大化

[Nemhauser-Wolsey-Fisher 1978]

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ

$$\max \quad f(S) \qquad \text{s.t.} \qquad |S| \leq r$$

☞ 簡単な貪欲アルゴリズムで **(1 − 1/e)近似**

サイズ制約下での単調劣モ最大化

[Nemhauser-Wolsey-Fisher 1978]

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ

$$\max \quad f(S) \qquad \text{s.t.} \quad |S| \leq r$$

☞ 簡単な貪欲アルゴリズムで **(1 − 1/e)近似**

$f(S) \geq (1 - 1/e)\, f(OPT)$を出力

サイズ制約下での単調劣モ最大化

[Nemhauser-Wolsey-Fisher 1978]

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ

$$\max \quad f(S) \qquad \text{s.t.} \quad |S| \leq r$$

☞ 簡単な貪欲アルゴリズムで **(1 – 1/e)近似**

$f(S) \geq (1 - 1/e)\, f(OPT)$を出力

☞ **近似比$1 - 1/e$は多項式時間アルゴリズムの中で最良**

[Nemhauser-Wolsey 1978]

サイズ制約下での単調劣モ最大化

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ

$$\max \quad f(S) \quad \text{s.t.} \quad \boxed{|\boldsymbol{S}| \leq \boldsymbol{r}}$$

☞ 簡単な貪欲アルゴリズムで $\mathbf{(1 - 1/e)}$**近似**

サイズ制約 : $|\boldsymbol{S}| \leq \boldsymbol{r}$

サイズ制約下での単調劣モ最大化

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ

$$\max \quad f(S) \quad \text{s.t.} \quad \boxed{|\boldsymbol{S}| \leq \boldsymbol{r}}$$

☞ 簡単な貪欲アルゴリズムで **(1 − 1/e)近似**

ナップサック制約 : $\sum_{v \in S} \boldsymbol{w_v} \leq \mathbf{1}$

サイズ制約 : $|\boldsymbol{S}| \leq \boldsymbol{r}$ ⇒ **拡張！**

マトロイド制約 : $S \in \boldsymbol{\mathcal{I}}$

マトロイド制約下での単調劣モ最大化

$f: 2^V \to \mathbb{R}_{\geq 0}$: 非負単調劣モジュラ

$$\max \quad f(S) \qquad \text{s.t.} \quad \boldsymbol{S \in \mathcal{I}}$$

☞ **(1 − 1/e)近似**アルゴリズム？

ナップサック制約 : $\sum_{v \in S} \boldsymbol{w_v} \leq \mathbf{1}$

サイズ制約 : $|\boldsymbol{S}| \leq \boldsymbol{r}$

拡張！

マトロイド制約 : $\boldsymbol{S \in \mathcal{I}}$

マトロイド制約下での単調劣モ最大化

$f: 2^V \to \mathbb{R}_{\geq 0}$：非負単調劣モジュラ

$$\max \quad f(S) \qquad \text{s.t.} \quad \boldsymbol{S \in \mathcal{I}}$$

☞ **連続貪欲法**で**(1 − 1/e)近似**アルゴリズム [Calinescu et al. 2007]

ナップサック制約：$\sum_{v \in S} w_v \leq 1$

サイズ制約：$|S| \leq r$

拡張！

マトロイド制約：$S \in \mathcal{I}$

マトロイド $\mathcal{M} = (V, \mathcal{I})$: 線形独立性の一般化

$\mathcal{I}$の要素を**独立**集合と呼ぶ

<u>定義</u>

有限集合 V 上の空でない部分集合族 $\mathcal{I} \subseteq 2^V$ で次のよい性質を持つもの

- $S' \subseteq S \in \mathcal{I} \implies S' \in \mathcal{I}$
- $S, T \in \mathcal{I}, |S| > |T| \implies \exists\, v \in S - T$ s.t. $T \cup \{v\} \in \mathcal{I}$

マトロイド $\mathcal{M} = (V, \mathcal{I})$: 線形独立性の一般化

$\mathcal{I}$の要素を**独立**集合と呼ぶ

<u>定義</u>

有限集合 V 上の空でない部分集合族 $\mathcal{I} \subseteq 2^V$ で次のよい性質を持つもの

- $S' \subseteq S \in \mathcal{I} \implies S' \in \mathcal{I}$
- $S, T \in \mathcal{I}$, $|S| > |T| \implies \exists\, v \in S - T$ s.t. $T \cup \{v\} \in \mathcal{I}$

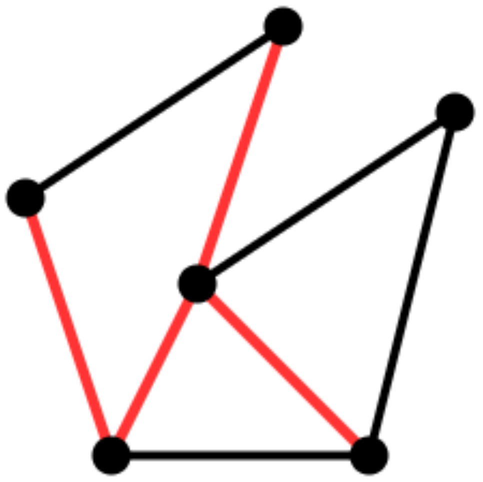
例）

- **線形マトロイド**

$$\begin{bmatrix} 0 & 1 & 2 & 0 \\ 3 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 1 & 2 & 3 & 0 \end{bmatrix}$$

V = 行ベクトル
$\mathcal{I}$ = 線形独立

- **グラフ的マトロイド**



V = 辺集合
$\mathcal{I}$ = 森全体

例) 組合せオークション

[Lehmann-Lehmann-Nisan 2006]

単調劣モ関数

入力: 財の集合 V,

iさんの効用関数 $f_i : 2^V \to \mathbb{R}_{\geq 0}$ $(i = 1, \ldots, m)$

f_1

f_2

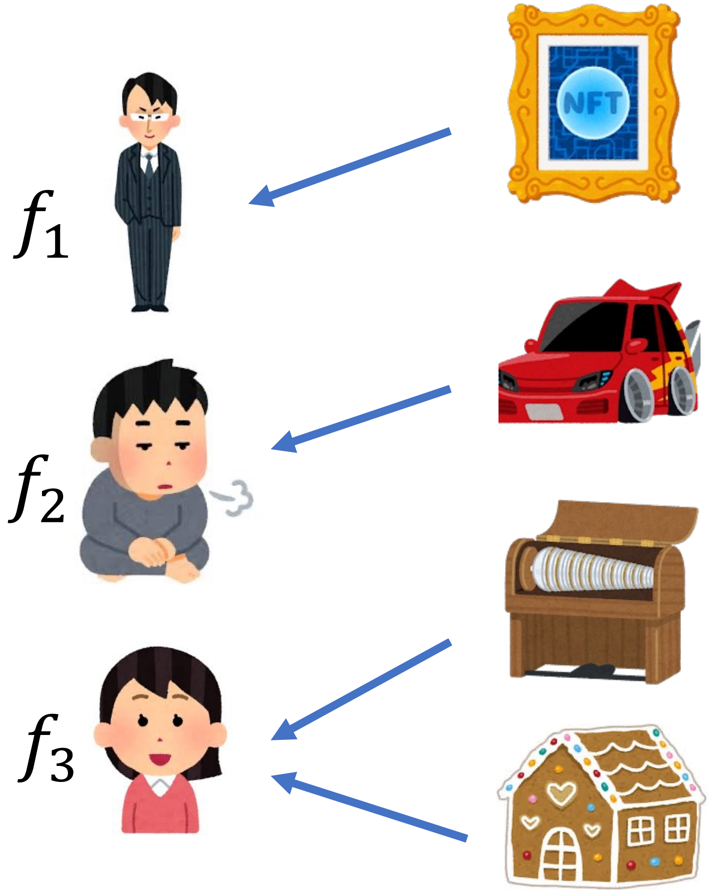
f_3

例）組合せオークション

[Lehmann-Lehmann-Nisan 2006]

単調劣モ関数

入力: 財の集合 V,
　　iさんの効用関数 $f_i: 2^V \to \mathbb{R}_{\geq 0}$ $(i = 1, \dots, m)$
出力: 財の分配 $(V_1, \dots, V_m)$ s.t. $\sum_i f(V_i)$が最大



例）組合せオークション

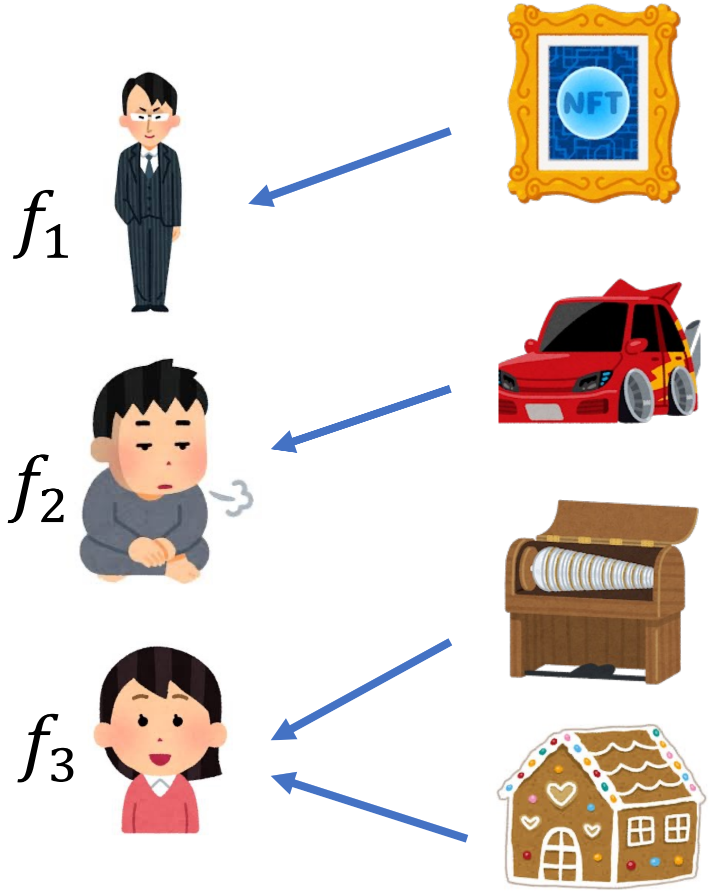
[Lehmann-Lehmann-Nisan 2006]

単調劣モ関数

入力: 財の集合 V,

iさんの効用関数 $f_i: 2^V \to \mathbb{R}_{\geq 0}$ $(i = 1, \dots, m)$

出力: 財の分配 $(V_1, \dots, V_m)$ s.t. $\sum_i f(V_i)$が最大



☞ **分割マトロイド制約の単調劣モ最大化**

で解ける！

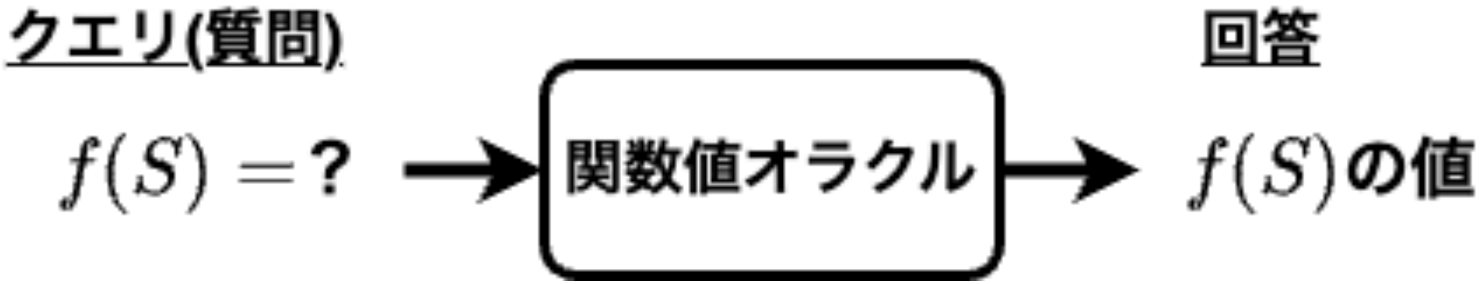
計算量の評価

Q. 劣モジュラ関数やマトロイドをどう扱うのか？

計算量の評価

Q. 劣モジュラ関数やマトロイドをどう扱うのか？

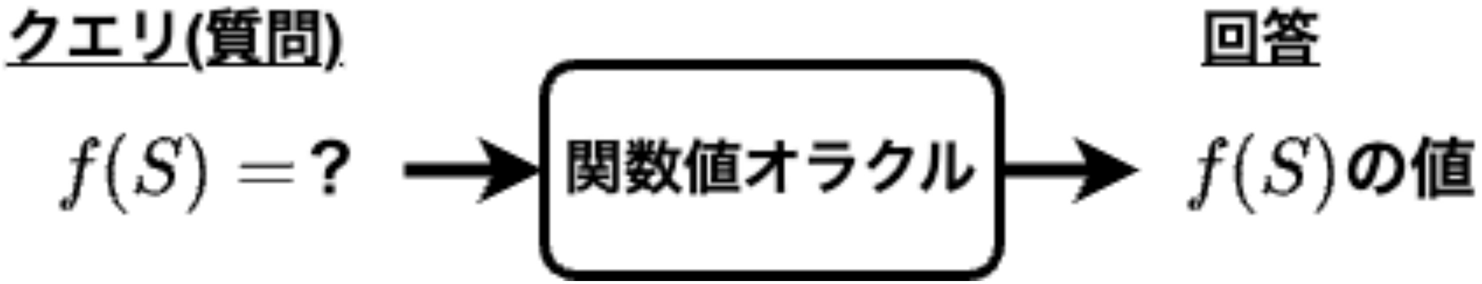
劣モジュラ関数へのアクセス



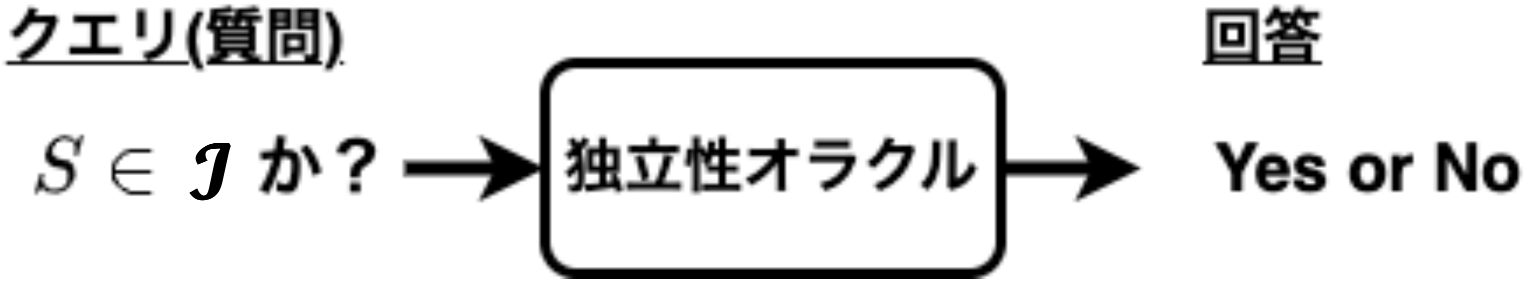
計算量の評価

Q. 劣モジュラ関数やマトロイドをどう扱うのか？

劣モジュラ関数へのアクセス



マトロイドへのアクセス



マトロイド制約下での単調劣モ最大化の $(1-1/e)$近似アルゴリズムの計算量

		クエリ回数
2007	Calinescu-Chekuri-Pál-Vondrák	$\tilde{O}(n^8)$

$n = |V|$, マトロイドのランク $\boldsymbol{r}(\leq \boldsymbol{n})$

マトロイド制約下での単調劣モ最大化の $(1-1/e-\epsilon)$近似アルゴリズムの計算量

		クエリ回数
2007	Calinescu-Chekuri-Pál-Vondrák	$\tilde{O}(n^8)$
2012	Filmus-Ward	$\tilde{O}_\epsilon(rn^4)$
2014	Badanidiyuru-Vondrák	$\tilde{O}_\epsilon(rn)$
2015	Buchbinder-Feldman-Schwartz	$\tilde{O}_\epsilon(r^2+\sqrt{r}n)$

$n = |V|$, マトロイドのランク $\boldsymbol{r}(\leq \boldsymbol{n})$

マトロイド制約下での単調劣モ最大化の $(1-1/e-\epsilon)$ 近似アルゴリズムの計算量

		クエリ回数
2007	Calinescu-Chekuri-Pál-Vondrák	$\tilde{O}(n^8)$
2012	Filmus-Ward	$\tilde{O}_\epsilon(rn^4)$
2014	Badanidiyuru-Vondrák	$\tilde{O}_\epsilon(rn)$
2015	Buchbinder-Feldman-Schwartz	$\tilde{O}_\epsilon(r^2+\sqrt{r}n)$
2024	**本研究**	$\boldsymbol{\tilde{O}_\epsilon(\sqrt{r}n)}$

$n=|V|$, マトロイドのランク $\boldsymbol{r}(\leq \boldsymbol{n})$

連続貪欲法 [Calinescu-Chekuri-Pál-Vondrák 2007]

元問題

max $\boldsymbol{f(S)}$ s.t. $\boldsymbol{S} \in \boldsymbol{\mathcal{I}}$

連続貪欲法 [Calinescu-Chekuri-Pál-Vondrák 2007]

元問題

max $f(S)$ s.t. $S \in \mathcal{I}$

緩和問題

max $F(x)$ s.t. $x \in \mathcal{B}(\mathcal{M})$

①**連続緩和**

fの多重線形拡張$F: [0,1]^n \to \mathbb{R}$

連続貪欲法 [Calinescu-Chekuri-Pál-Vondrák 2007]

元問題

max $\boldsymbol{f}(\boldsymbol{S})$ s.t. $\boldsymbol{S} \in \boldsymbol{\mathcal{I}}$

①連続緩和

緩和問題

max $\boldsymbol{F}(\boldsymbol{x})$ s.t. $\boldsymbol{x} \in \boldsymbol{\mathcal{B}}(\boldsymbol{\mathcal{M}})$

②連続貪欲法

$\boldsymbol{x} \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[\boldsymbol{F}(\boldsymbol{x})] \geq (\mathbf{1} - \mathbf{1}/\boldsymbol{e})\boldsymbol{f}(\boldsymbol{OPT})$

連続貪欲法 [Calinescu-Chekuri-Pál-Vondrák 2007]

元問題

$\max f(S)$ s.t. $S \in \mathcal{I}$

緩和問題

$\max F(x)$ s.t. $x \in \mathcal{B}(\mathcal{M})$

①連続緩和

②連続貪欲法

$x \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[F(x)] \geq (1 - 1/e) f(OPT)$

③丸め

$S \in \mathcal{I}$

s.t.

$\mathbb{E}[f(S)] \geq (1 - 1/e) f(OPT)$

連続貪欲法の高速化

元問題

$\max f(S)$ s.t. $S \in \mathcal{I}$

緩和問題

$\max F(x)$ s.t. $x \in \mathcal{B}(\mathcal{M})$

①連続緩和

②連続貪欲法

$x \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[F(x)] \geq (1 - 1/e - \epsilon) f(OPT)$

③丸め

$S \in \mathcal{I}$

s.t.

$\mathbb{E}[f(S)] \geq (1 - 1/e - \epsilon) f(OPT)$

連続貪欲法の高速化

元問題

max $\boldsymbol{f}(\boldsymbol{S})$ s.t. $\boldsymbol{S} \in \boldsymbol{\mathcal{I}}$

①連続緩和

緩和問題

max $\boldsymbol{F}(\boldsymbol{x})$ s.t. $\boldsymbol{x} \in \boldsymbol{\mathcal{B}}(\boldsymbol{\mathcal{M}})$

②連続貪欲法

$\boldsymbol{\widetilde{O}_{\epsilon}(rn)}$**回クエリ**

[Badanidiyuru-Vondrák 2014]

$\boldsymbol{x} \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[\boldsymbol{F}(\boldsymbol{x})] \geq (\boldsymbol{1} - \boldsymbol{1}/\boldsymbol{e} - \boldsymbol{\epsilon})\boldsymbol{f}(\boldsymbol{OPT})$

③丸め

$\boldsymbol{S} \in \mathcal{I}$

s.t.

$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq (\boldsymbol{1} - \boldsymbol{1}/\boldsymbol{e} - \boldsymbol{\epsilon})\boldsymbol{f}(\boldsymbol{OPT})$

連続貪欲法の高速化

元問題

max $f(S)$ s.t. $S \in \mathcal{I}$

①連続緩和

緩和問題

max $F(x)$ s.t. $x \in \mathcal{B}(\mathcal{M})$

②連続貪欲法

$\tilde{O}_\epsilon(rn)$**回クエリ**

[Badanidiyuru-Vondrák 2014]

$x \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[F(x)] \geq (1 - 1/e - \epsilon) f(OPT)$

③丸め

$S \in \mathcal{I}$

s.t.

$\mathbb{E}[f(S)] \geq (1 - 1/e - \epsilon) f(OPT)$

$O_\epsilon(r^2)$**回クエリ** [Chekuri-Vondrák-Zenklusen 2010]

連続貪欲法の高速化

元問題

$\max f(S)$ s.t. $S \in \mathcal{I}$

①連続緩和

緩和問題

$\max F(x)$ s.t. $x \in \mathcal{B}(\mathcal{M})$

②連続貪欲法

$\tilde{O}_\epsilon(\sqrt{r}n)$**回クエリ**

[Buchbinder et al. 2015]

$x \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[F(x)] \geq (1 - 1/e - \epsilon) f(OPT)$

③丸め

$O_\epsilon(r^2)$**回クエリ** [Chekuri-Vondrák-Zenklusen 2010]

$S \in \mathcal{I}$

s.t.

$\mathbb{E}[f(S)] \geq (1 - 1/e - \epsilon) f(OPT)$

連続貪欲法の高速化

元問題

$\max f(S)$ s.t. $S \in \mathcal{I}$

①連続緩和

緩和問題

$\max F(x)$ s.t. $x \in \mathcal{B}(\mathcal{M})$

②連続貪欲法

$\tilde{O}_\epsilon(\sqrt{r}n)$**回クエリ**

[Buchbinder et al. 2015]

$x \in \mathcal{B}(\mathcal{M})$

s.t.

$\mathbb{E}[F(x)] \geq (1 - 1/e - \epsilon) f(OPT)$

③丸め

$S \in \mathcal{I}$

s.t.

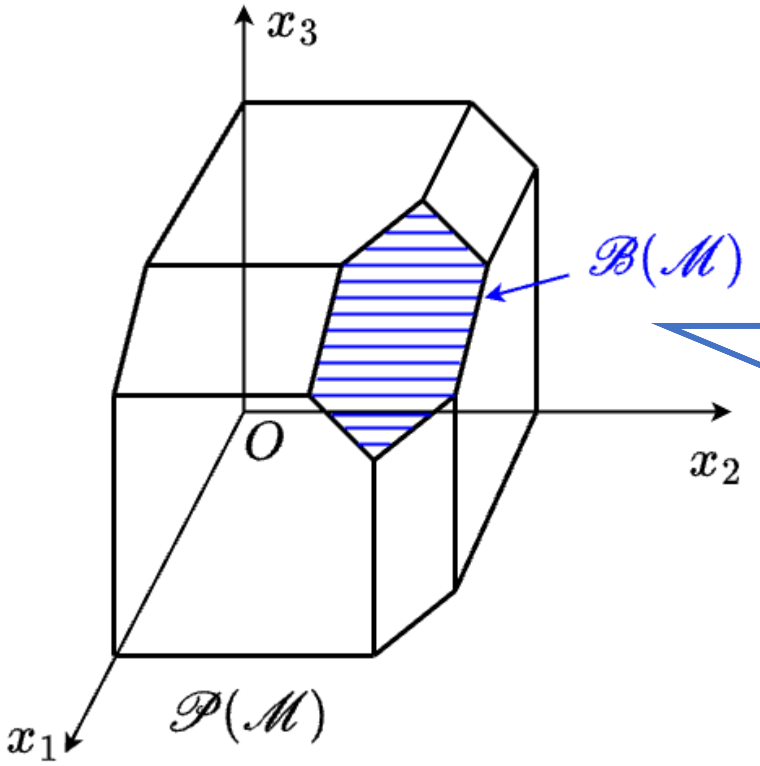
$\mathbb{E}[f(S)] \geq (1 - 1/e - \epsilon) f(OPT)$

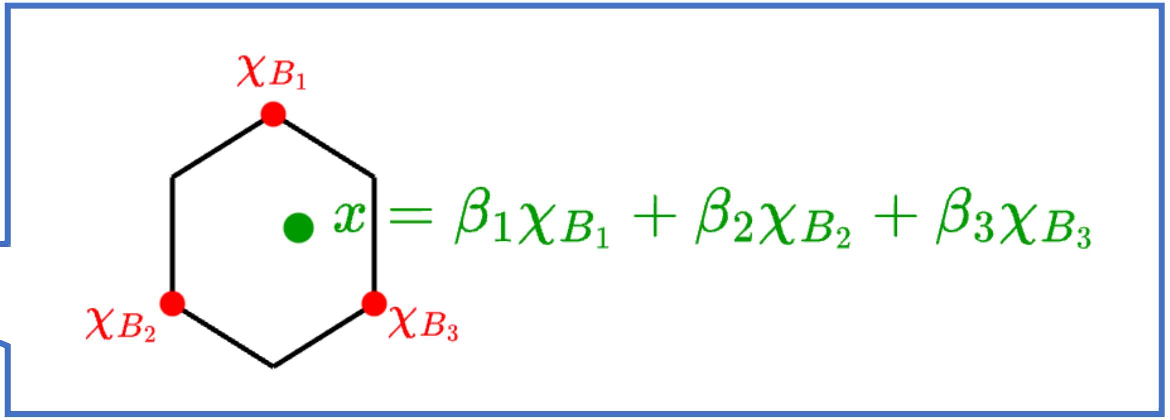
$\tilde{O}_\epsilon(r^{3/2})$**回クエリ** [本研究]

丸めアルゴリズム [Chekuri-Vondrák-Zenklusen 2010]

$\boldsymbol{x = \beta_1 \chi_{B_1} + \cdots + \beta_t \chi_{B_t}}$

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $x \in \mathcal{B}(\mathcal{M})$

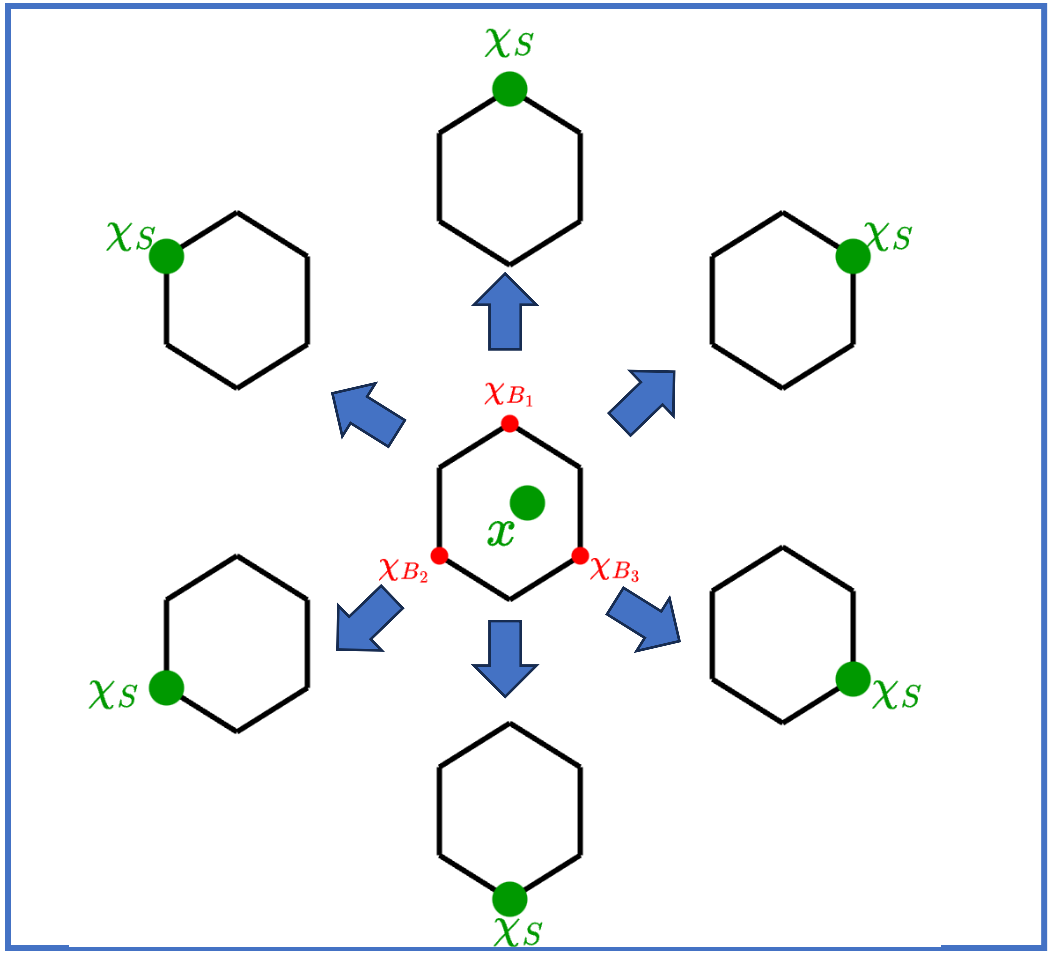




丸めアルゴリズム [Chekuri-Vondrák-Zenklusen 2010]

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $x \in \mathcal{B}(\mathcal{M})$

出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{F}(\boldsymbol{\chi_S})] \geq \boldsymbol{F}(\boldsymbol{x})$



$\boldsymbol{F}(\boldsymbol{\chi_S}) = \boldsymbol{f}(\boldsymbol{S})$

丸めアルゴリズム [Chekuri-Vondrák-Zenklusen 2010]

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $x \in \mathcal{B}(\mathcal{M})$
出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq \boldsymbol{F}(\boldsymbol{x})$

<u>定理</u> [Chekuri-Vondrák-Zenklusen 2010]
$\boldsymbol{O}(\boldsymbol{r}^2\boldsymbol{t})$回の独立性オラクルの使用

高速な丸めアルゴリズム

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $x \in \mathcal{B}(\mathcal{M})$
出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq (\mathbf{1} - \boldsymbol{\epsilon})\boldsymbol{F}(\boldsymbol{x})$

定理 [Chekuri-Vondrák-Zenklusen 2010]
$\boldsymbol{O}(\boldsymbol{r}^2\boldsymbol{t})$回の独立性オラクルの使用

定理 [本研究]
$\tilde{\boldsymbol{O}}_{\boldsymbol{\epsilon}}(\boldsymbol{r}^{3/2}\boldsymbol{t})$回の独立性オラクルの使用

[Chekuri-Vondrák-Zenklusen 2010]の丸めアルゴリズムの概要

$\boldsymbol{x} = \boldsymbol{\beta}_1 \boldsymbol{\chi}_{\boldsymbol{B}_1} + \cdots + \boldsymbol{\beta}_t \boldsymbol{\chi}_{\boldsymbol{B}_t}$

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $\boldsymbol{x} \in \mathcal{B}(\mathcal{M})$
出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq \boldsymbol{F}(\boldsymbol{x})$

(イメージ)

各B_iを β_iの比で混ぜて得られる基を出力

[Chekuri-Vondrák-Zenklusen 2010]の丸めアルゴリズムの概要

$\boldsymbol{x} = \boldsymbol{\beta_1 \chi_{B_1}} + \cdots + \boldsymbol{\beta_t \chi_{B_t}}$

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $\boldsymbol{x} \in \mathcal{B}(\mathcal{M})$

出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq \boldsymbol{F}(\boldsymbol{x})$

(イメージ)

各B_iを β_iの比で混ぜて得られる基を出力

B_1と B_2をβ_1:β_2で混ぜるとすると…

Repeat:

(Step1) $B_1 + v - u \in \mathcal{I}$かつ $B_2 + u - v \in \mathcal{I}$ なる要素 v, u をみつける

(Step2) 確率 $\beta_2/(\beta_1 + \beta_2)$で $B_1 \leftarrow B_1 + v - u$
確率 $\beta_1/(\beta_1 + \beta_2)$で $B_2 \leftarrow B_2 + u - v$

[Chekuri-Vondrák-Zenklusen 2010]の丸めアルゴリズムの概要

$\boldsymbol{x} = \boldsymbol{\beta_1 \chi_{B_1}} + \cdots + \boldsymbol{\beta_t \chi_{B_t}}$

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $\boldsymbol{x} \in \mathcal{B}(\mathcal{M})$
出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq \boldsymbol{F}(\boldsymbol{x})$

(イメージ)
各B_iを β_iの比で混ぜて得られる基を出力

結局…

Step: $B_1 + v - u \in \mathcal{I}$かつ $B_2 + u - v \in \mathcal{I}$ なる要素 v, u をみつける

を$\boldsymbol{O(rt)}$回繰り返せば良い！

[Chekuri-Vondrák-Zenklusen 2010]の丸めアルゴリズムの概要

$\boldsymbol{x} = \boldsymbol{\beta_1 \chi_{B_1}} + \cdots + \boldsymbol{\beta_t \chi_{B_t}}$

入力: $\mathcal{M} = (V, \mathcal{I})$, t個の基の凸結合で与えられる点 $\boldsymbol{x} \in \mathcal{B}(\mathcal{M})$
出力: $\mathcal{M}$の基 S s.t. 任意の劣モ関数fに対して$\mathbb{E}[\boldsymbol{f}(\boldsymbol{S})] \geq \boldsymbol{F}(\boldsymbol{x})$

(イメージ)
各B_iを β_iの比で混ぜて得られる基を出力

結局…

1step $\boldsymbol{O}(\boldsymbol{r})$**回クエリ**

Step: $B_1 + v - u \in \mathcal{I}$かつ $B_2 + u - v \in \mathcal{I}$ なる要素 v, u をみつける

を$\boldsymbol{O}(\boldsymbol{rt})$回繰り返せば良い！

技術的貢献①: 任意の長さの閉路でも良い

- [Chekuri-Vondrák-Zenklusen 2010]

  $\boxed{\text{Step: } B_1 + v - u \in \mathcal{I} \text{かつ } B_2 + u - v \in \mathcal{I} \text{ なる要素 } v, u \text{ をみつける}}$

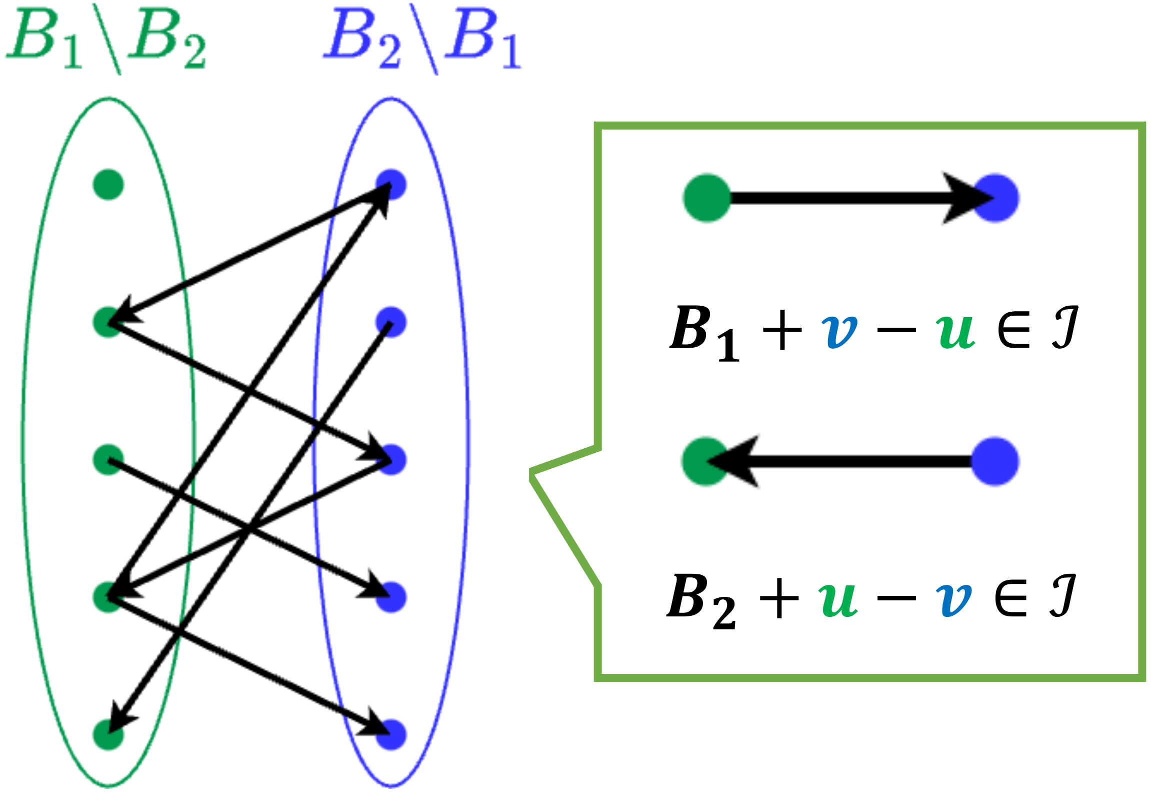
技術的貢献①: 任意の長さの閉路でも良い

- [Chekuri-Vondrák-Zenklusen 2010]

Step: $B_1 + v - u \in \mathcal{I}$かつ $B_2 + u - v \in \mathcal{I}$ なる要素 v, u をみつける

拡張！

- [本研究]



技術的貢献①: 任意の長さの閉路でも良い

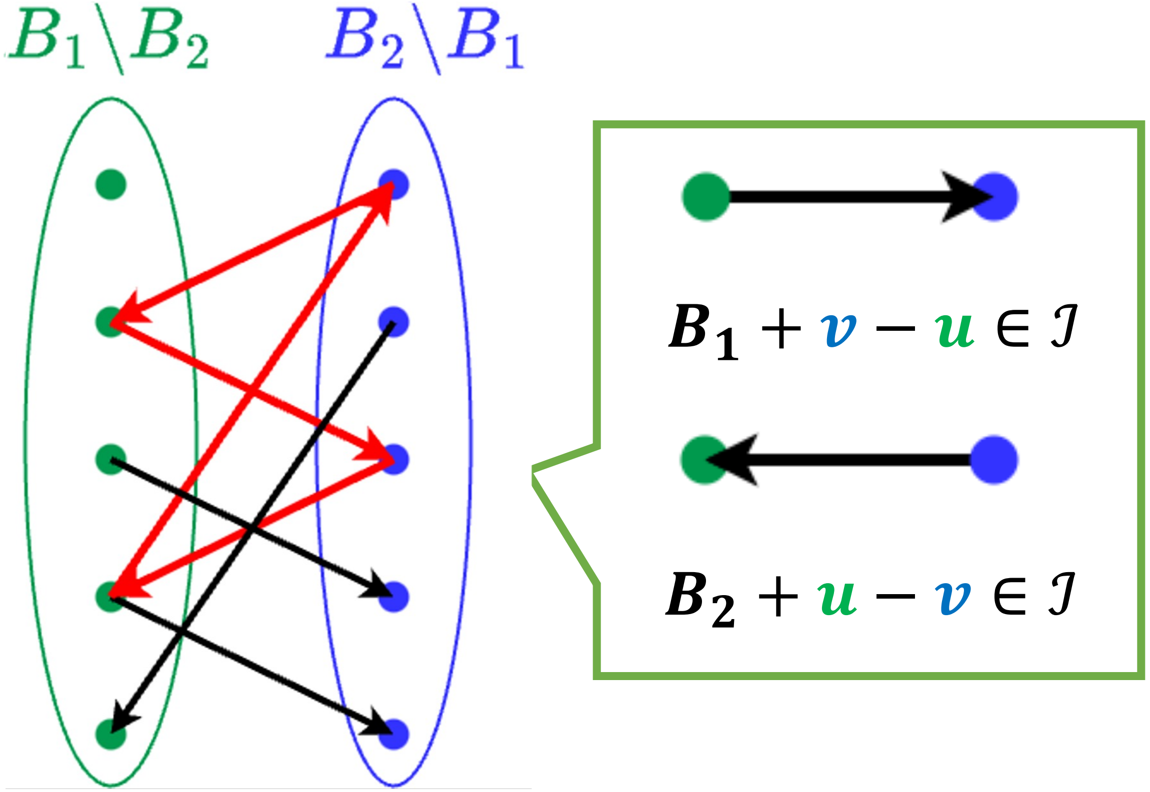
- [Chekuri-Vondrák-Zenklusen 2010]

Step: $B_1 + v - u \in \mathcal{I}$かつ $B_2 + u - v \in \mathcal{I}$ なる要素 v, u をみつける

- [本研究]

拡張！

Step: 補助グラフ上で**閉路**をみつける



技術的貢献①: 任意の長さの閉路でも良い

- [Chekuri-Vondrák-Zenklusen 2010]

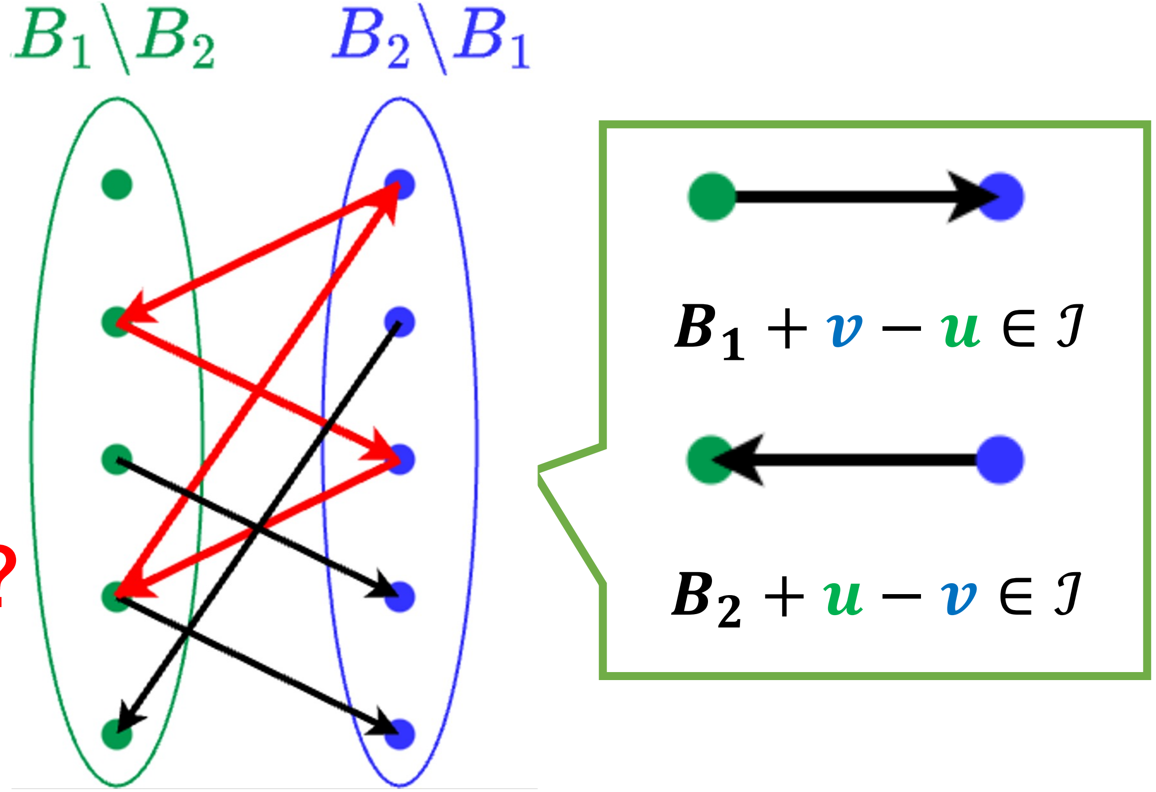
Step: $B_1 + v - u \in \mathcal{I}$かつ $B_2 + u - v \in \mathcal{I}$ なる要素 v, u をみつける

- [本研究]

拡張！

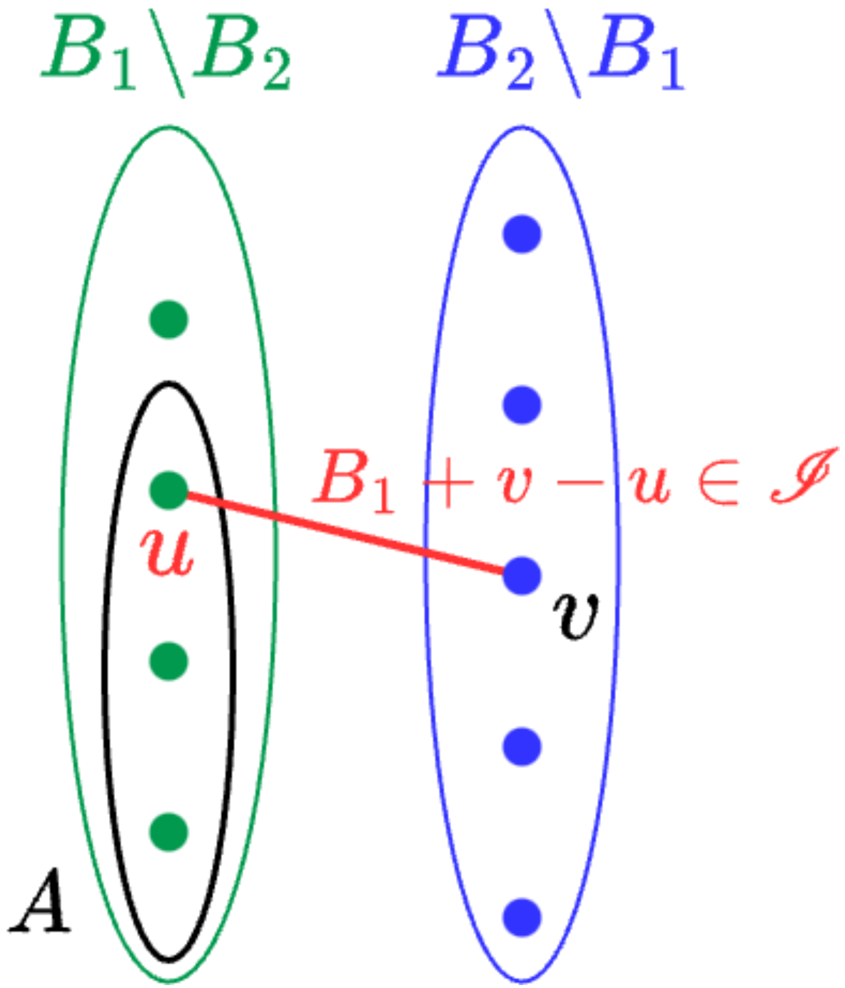
Step: 補助グラフ上で**閉路**をみつける

Q. 閉路を高速にみつけられるのか？



マトロイド交叉の高速化の道具

[Nguyễn 2019, Chakrabarty et al. 2019]



入力：$\mathcal{M} = (V, \mathcal{I})$, $B \in \mathcal{I}$, $v \in V \setminus B$, $A \subseteq B$
出力：$B - u + v \in \mathcal{I}$ なる $u \in A$ を一つ

二分探索を用いることで、
$\boldsymbol{O(\log |A|)}$ 回の独立性オラクルへのクエリでできる

技術的貢献②: 高速に閉路を見つけるアルゴリズム

結局…

<u>Step: 閉路をみつける</u>

を$\boldsymbol{O(rt)}$回繰り返せば良い！

技術的貢献②: 高速に閉路を見つけるアルゴリズム

補題: 1step十分高い確率で $\tilde{O}(\sqrt{r})$**回クエリ**でできる

結局…

Step: 閉路をみつける

を$O(rt)$回繰り返せば良い！

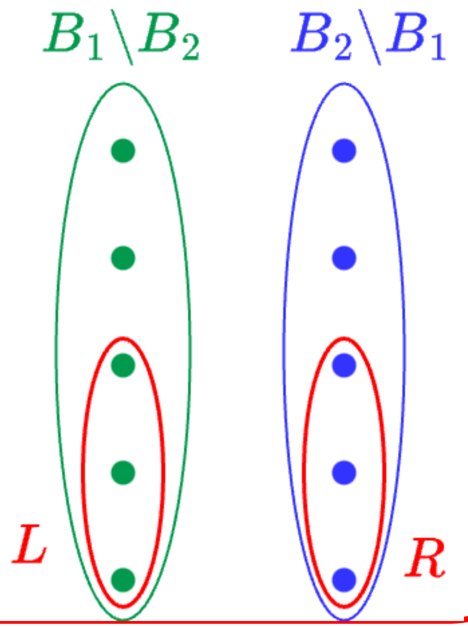
技術的貢献②: 高速に閉路を見つけるアルゴリズム

結局…

補題: 1step十分高い確率で $\tilde{O}(\sqrt{r})$**回クエリ**でできる

Step: 閉路をみつける

① 左右から$\tilde{O}(\sqrt{r})$個ずつ頂点集合 L, Rをサンプル
② $G[L \cup R]$での各頂点の次数を調べる
③ if (全ての頂点の次数)≥ 1 then 閉路が存在
else 次数0の頂点を通る長さ2の閉路を探索



を$O(rt)$回繰り返せば良い！

結論

- **マトロイド制約下での単調劣モジュラ最大化**問題に対して
  オラクルの使用回数の少ないアルゴリズムを設計
- [Chekuri-Vondrák-Zenklusen 2010]よりも**高速な丸めアルゴリズム**

- **任意の長さの閉路に拡張**
- **閉路を高速に見つけるアルゴリズムの開発**
  ☞ マトロイド交叉の高速化のテクニック

結論

- **マトロイド制約下での単調劣モジュラ最大化**問題に対して
  オラクルの使用回数の少ないアルゴリズムを設計
- [Chekuri-Vondrák-Zenklusen 2010]よりも**高速な丸めアルゴリズム**

> - **任意の長さの閉路に拡張**
> - **閉路を高速に見つけるアルゴリズムの開発**
>   ☞ **マトロイド交叉の高速化のテクニック**

Q. 本結果はさらに改善できるか？

(参考)

- マトロイドが**ランクオラクル**で与えられている場合 : $\tilde{O}_\epsilon(n + r^{3/2})$回クエリ [本研究]
- 特殊なマトロイドの場合 : 線形時間 [Ene-Nguyễn 2019, Henzinger et al. 2023]