整数格子点上の劣モジュラ被覆に対する高速アルゴリズム

(NIPS 2015 採択論文)

相馬 輔 (東京大学)
共同研究者: 吉田 悠一 (NII & PFI)

ERATO 感謝祭 Season III

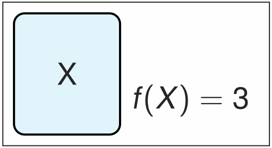
河原林巨大グラフプロジェクト
Kawarabayashi Large Graph Project

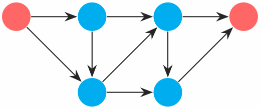
概要

- 劣モジュラ被覆と呼ばれる機械学習で
  よく使われる問題を，**整数格子点上に拡張**
- 拡張モデルは既存モデルより**複雑な状況を扱える**
- **効率的な近似アルゴリズムを設計**

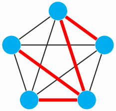
劣モジュラ最適化

劣モジュラ関数: “効率的” に解ける離散構造に現れる関数





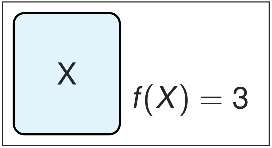
ネットワークフロー

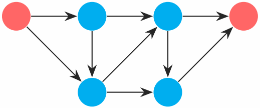


全域木

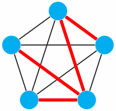
劣モジュラ最適化

劣モジュラ関数: “効率的” に解ける離散構造に現れる関数





ネットワークフロー



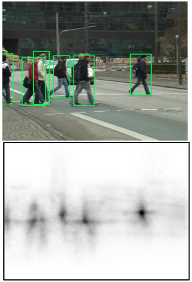
全域木

$f : 2^S \to \mathbf{R}$ が **劣モジュラ関数**:

$$f(X \cup s) - f(X) \geq f(Y \cup s) - f(Y) \quad (X \subseteq Y, s \notin Y)$$

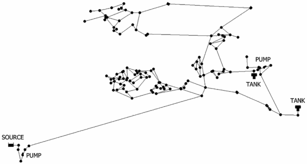
“限界効用逓減 (diminishing return)”

機械学習での劣モジュラ最適化



オブジェクト検知

[Song et al. ’14]



センサー配置問題

[Krause-Guestrin ’05]

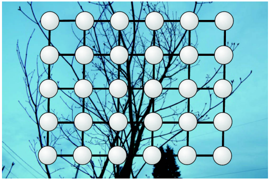


Image Segmentation

[Jegelka-Bilmes ’11]

劣モジュラ被覆 [Wolsey '82]

$c, f : 2^S \to \mathbf{R}_+$ 単調劣モジュラ関数, $\alpha > 0$

$$\min \quad c(X) \qquad \text{sub. to} \quad f(X) \geq \alpha$$

劣モジュラ被覆 [Wolsey '82]

$c, f : 2^S \to \mathbf{R}_+$ 単調劣モジュラ関数, $\alpha > 0$

$$\min \quad c(X) \qquad \text{sub. to} \quad f(X) \geq \alpha$$

コスト

解の品質保証

劣モジュラ被覆 [Wolsey ’82]

$c, f : 2^S \to \mathbf{R}_+$ 単調劣モジュラ関数, $\alpha > 0$

$$\min \quad \underbrace{c(X)}_{\text{コスト}} \qquad \text{sub. to} \quad \underbrace{f(X)}_{\text{解の品質保証}} \geq \alpha$$

応用:

- センサー配置 [Krause-Guestrin ’05, Krause-Leskovec ’08]
- 文書要約 [Lin-Bilmes ’10]
- オブジェクト検知 [Song et al. ’14, Chen et al. ’14]

劣モジュラ被覆 [Wolsey '82]

$c, f : 2^S \to \mathbf{R}_+$ 単調劣モジュラ関数, $\alpha > 0$

$$\min \quad \underbrace{c(X)}_{\text{コスト}} \qquad \text{sub. to} \quad \underbrace{f(X) \geq \alpha}_{\text{解の品質保証}}$$

既存研究:

- NP 困難
- $c(X) = |X|$ のとき, $O(\log \frac{d}{\beta})$-近似 [Wolsey '82]
- f, c が整数値のとき, $O(\rho \log d)$-近似 [Wan et al. '09]

$d = \max_s f(s)$, ρ: c の曲率,

$\beta := \min\{f(s \mid X) : s \in S, X \subseteq S, f(s \mid X) > 0\}$

整数格子点上の劣モジュラ被覆

$c, f : \mathbf{Z}_+^S \to \mathbf{R}_+$ **整数格子点上の**単調 DR 劣モジュラ関数
$\alpha > 0$, $r \in \mathbf{Z}_+$

$$\min \quad c(\boldsymbol{x}) \qquad \text{sub. to} \quad f(\boldsymbol{x}) \geq \alpha, \mathbf{0} \leq \boldsymbol{x} \leq r\mathbf{1}$$

注意: $r = 1 \implies$ (集合版) 劣モジュラ被覆

結果

$\mathbf{Z}_+^S$ 上の劣モジュラ被覆に対する近似アルゴリズム

- 近似比: $O(\rho \log d/\beta)$, 計算量: $O\left(n \log nr \log r\right)$
- $r = 1$ の場合，既存の最良近似比と一致
- 数値実験により解の品質・効率性を確認

$$d := \max_s f(\boldsymbol{e}_s),\ \rho := c\ \text{の曲率},$$
$$\beta := \min\{f(\boldsymbol{e}_s \mid \boldsymbol{x}) : s \in S, \boldsymbol{x} \in \mathbf{Z}_+^S, f(\boldsymbol{e}_s \mid \boldsymbol{x}) > 0\}$$

整数格子点上の劣モジュラ関数

$f : \mathbf{Z}_+^S \to \mathbf{R}$

DR(Diminishing Return) 劣モジュラ:

$$f(\boldsymbol{x} + \mathbf{e}_i) - f(\boldsymbol{x}) \geq f(\boldsymbol{y} + \mathbf{e}_i) - f(\boldsymbol{y}) \qquad (\boldsymbol{x} \leq \boldsymbol{y} \in \mathbf{Z}^S, i \in S)$$

整数格子点上の劣モジュラ関数

$f : \mathbf{Z}_+^S \to \mathbf{R}$

DR(Diminishing Return) 劣モジュラ:

$$f(\boldsymbol{x} + \mathbf{e}_i) - f(\boldsymbol{x}) \geq f(\boldsymbol{y} + \mathbf{e}_i) - f(\boldsymbol{y}) \qquad (\boldsymbol{x} \leq \boldsymbol{y} \in \mathbf{Z}^S, i \in S)$$

cf. 限界効用逓減性:

$$f(X \cup s) - f(X) \geq f(Y \cup s) - f(Y)$$

整数格子点上の劣モジュラ関数

$f : \mathbf{Z}_+^S \to \mathbf{R}$

DR(Diminishing Return) 劣モジュラ:

$$f(\boldsymbol{x} + \mathbf{e}_i) - f(\boldsymbol{x}) \geq f(\boldsymbol{y} + \mathbf{e}_i) - f(\boldsymbol{y})$$

$$(\boldsymbol{x} \leq \boldsymbol{y} \in \mathbf{Z}^S, i \in S)$$

cf. 限界効用逓減性:

$$f(X \cup s) - f(X) \geq f(Y \cup s) - f(Y)$$

注意: 以下の不等式より**強い性質**

成分ごと max

成分ごと min

$$f(\boldsymbol{x}) + f(\boldsymbol{y}) \geq f(\boldsymbol{x} \vee \boldsymbol{y}) + f(\boldsymbol{x} \wedge \boldsymbol{y})$$

$$(\boldsymbol{x}, \boldsymbol{y} \in \mathbf{Z}^S)$$

cf.

$$f(X) + f(Y) \geq f(X \cup Y) + f(X \cap Y)$$

整数格子点上の劣モジュラ関数

$f : \mathbf{Z}_+^S \to \mathbf{R}$

DR(Diminishing Return) 劣モジュラ:

$$f(\boldsymbol{x} + \mathbf{e}_i) - f(\boldsymbol{x}) \geq f(\boldsymbol{y} + \mathbf{e}_i) - f(\boldsymbol{y}) \quad (\boldsymbol{x} \leq \boldsymbol{y} \in \mathbf{Z}^S, i \in S)$$

cf. 限界効用逓減性:

$$f(X \cup s) - f(X) \geq f(Y \cup s) - f(Y)$$

⇕ 集合関数だと同値

注意: 以下の不等式より**強い性質**

成分ごと max　成分ごと min

$$f(\boldsymbol{x}) + f(\boldsymbol{y}) \geq f(\boldsymbol{x} \vee \boldsymbol{y}) + f(\boldsymbol{x} \wedge \boldsymbol{y}) \quad (\boldsymbol{x}, \boldsymbol{y} \in \mathbf{Z}^S)$$

cf.

$$f(X) + f(Y) \geq f(X \cup Y) + f(X \cap Y)$$

整数格子点上の関数を考える意味

集合関数モデルの制限: **2値変数しか扱えない**

集合関数では捉えきれない問題

- 最適予算配分問題 [Alon et al.'13, S.-Kakimura-Inaba-Kawarabayashi '14]
- センサー配置問題

整数格子点上の劣モジュラ関数なら，既存モデルの利点を保ったまま多値変数を扱える

アイデア

Greedy
(集合版) 劣モジュラ
被覆へ帰着

- **擬多項式時間**
- 近似比は良い

アイデア

Greedy
(集合版) 劣モジュラ
被覆へ帰着

- **擬多項式時間**
- 近似比は良い

しきい値による高速化手法
[Badanidiyuru-Vondrák '14]

Decreasing Threshold
しきい値・二分探索 で
高速化した Greedy

- 多項式時間
- (定数倍を除き)
同じ近似比

Greedy

$\boldsymbol{x} \leftarrow \mathbf{0}$

while $f(\boldsymbol{x}) < \alpha$:

$\quad s \leftarrow \mathrm{argmax}_{s \in S} \left\{ \frac{f(\boldsymbol{e}_s | \boldsymbol{x})}{c(\boldsymbol{e}_s | \boldsymbol{x})} : x(s) < r \right\}$

$\quad \boldsymbol{x} \leftarrow \boldsymbol{x} + \boldsymbol{e}_s$

return $\boldsymbol{x}$

$$f(\boldsymbol{e}_s \mid \boldsymbol{x}) = f(\boldsymbol{x}_s + \boldsymbol{x}) - f(\boldsymbol{x})$$

$$\frac{f(\boldsymbol{e}_s \mid \boldsymbol{x})}{c(\boldsymbol{e}_s \mid \boldsymbol{x})} = \frac{\text{ゲイン増分}}{\text{コスト増分}} = \text{コスト・ゲイン比}$$

Greedy の問題点

✕ 1 歩ずつしか進めない ... **擬多項式時間**

Greedy の問題点

× 1 歩ずつしか進めない ... **擬多項式時間**

解決策

- **しきい値による高速化手法** [Badanidiyuru-Vondrák '14]
- **二分探索によるステップサイズの決定** [S.-Yoshida '15]

しきい値 θ よりコスト・ゲイン比が良い領域を
できるだけ大きく進む

提案アルゴリズム

$\boldsymbol{x} \leftarrow \boldsymbol{0}$

for ($\theta =$ 十分大きな値; $\theta \geq$ 十分小さな値; $\theta \leftarrow \theta(1-\epsilon)$) :

  for $s \in S$:

    $\dfrac{f(k\boldsymbol{e}_s \mid \boldsymbol{x})}{kc(\boldsymbol{e}_s)} \geq \theta$ となる**最大の** $0 < k \leq r - x(s)$ を二分探索

    if k が存在: $\boldsymbol{x} \leftarrow \boldsymbol{x} + k\boldsymbol{e}_s$

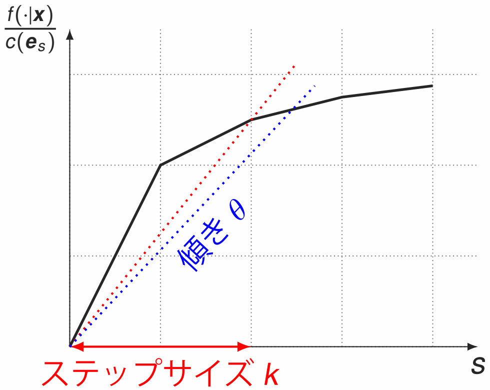
      if $f(\boldsymbol{x}) \geq \alpha$: **return** $\boldsymbol{x}$

return $\boldsymbol{x}$

提案アルゴリズム

$\dfrac{f(k\boldsymbol{e}_s \mid \boldsymbol{x})}{kc(\boldsymbol{e}_s)}$ = コスト・ゲイン比 $\dfrac{f(k\boldsymbol{e}_s \mid \boldsymbol{x})}{c(k\boldsymbol{e}_s \mid \boldsymbol{x})}$ の下界

k に関して**単調減少**



結果

定理

提案アルゴリズムは $(1+3\epsilon)\rho\left(1+\log\frac{d}{\beta}\right)$-近似の (ほぼ) 実行可能解を $O\left(\frac{n}{\epsilon}\log\frac{nrc_{\max}}{\delta c_{\min}}\log r\right)$ 時間で出力する.

注意:

- (ほぼ) 実行可能は $f(\boldsymbol{x}) \geq (1-\delta)\alpha$ の意味
- f:整数値 なら常に実行可能解を出力する

数値実験

$f(\boldsymbol{x})$: データセットから生成, $c(\boldsymbol{x}) := \|\boldsymbol{x}\|_1$

データセット:

- BWSN ... 現実の貯水池のネットワーク
- Synthetic ... ランダムデータ

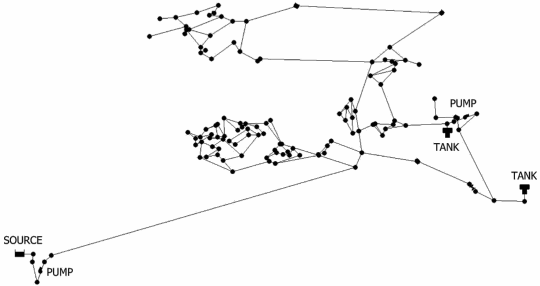
比較対象:

- Greedy ... (集合版) 劣モジュラ被覆への帰着 ※**擬多項式時間**
- Degree ... $f(\boldsymbol{e}_s)$ が大きい s から選択
- Uniform ... $f(k\mathbf{1}) \geq \alpha$ となる最小の $k\mathbf{1}$ を出力

計算機: Xeon E5-2690 2.9GHz CPU, 256GB RAM (4GB で十分)

BWSN (Battle of Water Sensor Network) [Ostfeld et al. ’08]

- 126 頂点, 168 枝
- 96 時間までの汚染シミュレーション 3000 通り



S: センサー集合, $\boldsymbol{x}$: センサーの電力分布,
$\Pr(s$ が汚染を検知する$) = 1 - (1 - p)^{\boldsymbol{x}(s)}$ $(p = 0.001)$

$$f(\boldsymbol{x}) = \mathbf{E}[96 \times 60 \times 60 - (\text{汚染検知時刻})]$$

Synthetic

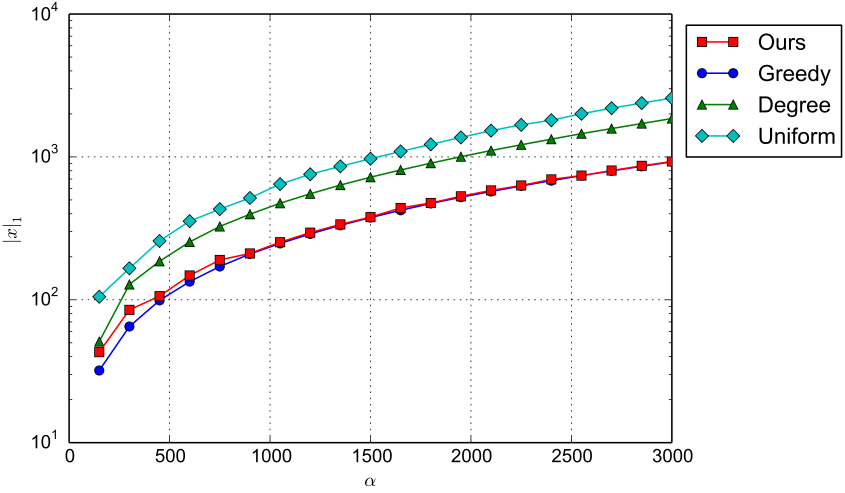
ランダム二部グラフ $G := (S, T; E)$

- $|S| = |T| = 100000$
- S 上の次数分布: $\Pr(d(s) = k) \propto k^{-3.0} \quad (s \in S)$
- 各 s はランダムな T の点に接続

パラメータ: $p = 0.001$

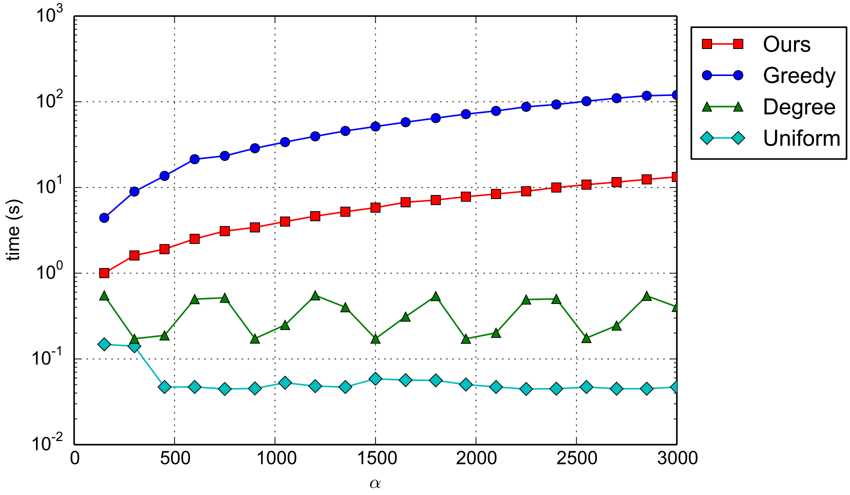
$$f(\boldsymbol{x}) := \sum_{t \in T} \left(1 - \prod_{s \in \Gamma(t)} (1 - p)^{\boldsymbol{x}(s)} \right)$$

BWSN: 目的関数値



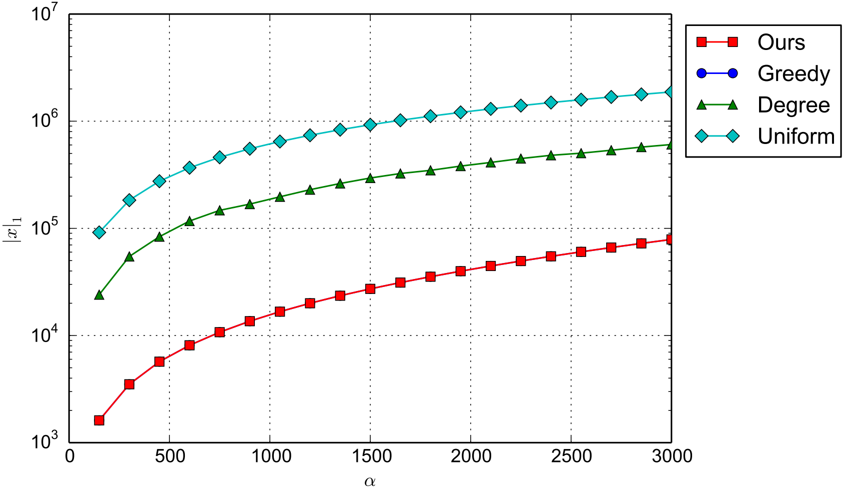
- Greedy に匹敵する性能

BWSN: 実行時間



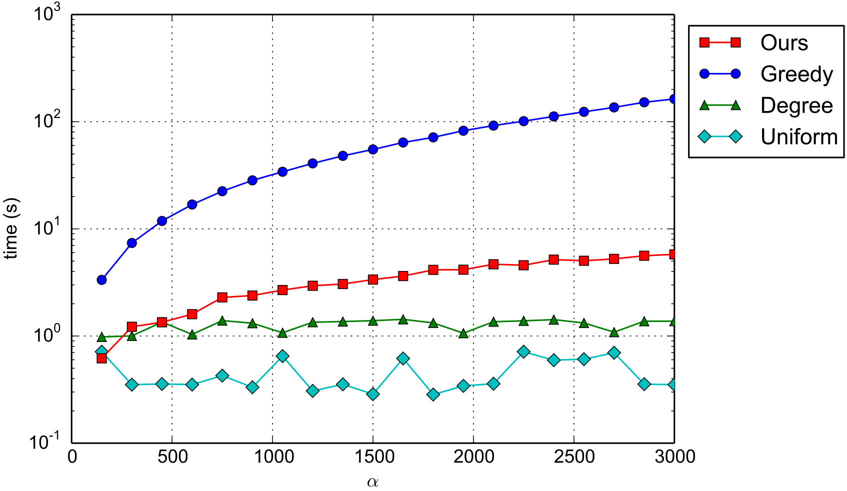
- Greedy より **10 倍**高速

Synthetic: 目的関数値



- Greedy に匹敵する性能

Synthetic: 実行時間



- Greedy より **10 倍**高速

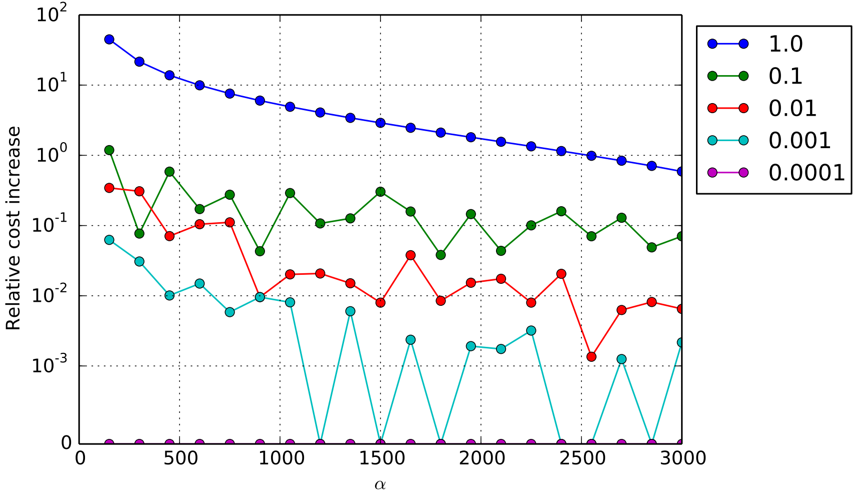
結果

$\mathbf{Z}_+^S$ 上の劣モジュラ被覆に対する近似アルゴリズム

- 近似比: $O(\rho \log d/\beta)$, 計算量: $O\left(n \log nr \log r\right)$
- $r = 1$ の場合，既存の最良近似比と一致
- 数値実験により解の品質・効率性を確認

$d := \max_s f(\boldsymbol{e}_s)$, $\rho := c$ の曲率,
$\beta := \min\{f(\boldsymbol{e}_s \mid \boldsymbol{x}) : s \in S, \boldsymbol{x} \in \mathbf{Z}_+^S, f(\boldsymbol{e}_s \mid \boldsymbol{x}) > 0\}$

ϵ とコスト (BWSN)



ϵ と実行時間 (BWSN)