最大リグレット最小化0-1整数計画問題に対するコア選択法

A core selection algorithm for the min–max regret binary integer programming problem

○長谷川 和樹 [1]
静岡大学

呉 偉 [2]
静岡大学

柳浦 睦憲 [3]
名古屋大学

[1]Kazuki Hasegawa
Shizuoka University
hasegawa.kazuki.19@shizuoka.ac.jp

[2]Wei Wu
Shizuoka University
goi@shizuoka.ac.jp

[3]Mutsunori Yagiura
Nagoya University
yagiura@nagoya-u.jp

Abstract In this paper, we propose a heuristic framework called the core selection (CS) method for the binary integer programming problem under the min–max regret criterion (MMR-BIP). This framework allows the integration of existing heuristic and exact algorithms as subroutines. In the first iteration, a heuristic algorithm is employed to quickly obtain a good solution, which is then used to construct a core set. The MMR-BIP is subsequently solved within this core set. In subsequent iterations, the core set is progressively enlarged, and the MMR-BIP is re-solved on the updated core set. Experimental results indicate that the CS method outperforms existing methods on certain instances.

1 はじめに

ナップサック問題や一般化割当問題をはじめとして，古典的な 0-1 整数計画問題 (binary integer programming problem, BIP) では各決定変数に対する入力は確定していた．しかしながら，現実世界の問題を考えると多くの入力データには不確定さが内在している．本研究で扱う最大リグレット最小化 BIP (min–max regret BIP, MMR-BIP) では，目的関数に関わる入力に対して不確定さを考慮する．

MMR-BIP の特殊ケースである最大リグレット最小化巡回セールスマン問題 (min–max regret traveling salesman problem, MMR-TSP) に対して Hasegawa と Wu [1] が発見的解法を提案し，MMR-TSP に対しては既存の解法を上回る結果を確認することができた．本研究では MMR-TSP に対する発見的解法を拡張し，改善手法であるコア選択 (core selection, CS) 法が一般化された MMR-BIP にも適用できること，および既存手法を上回る性能を持つことを確認する．

2 問題定義

2.1 0-1 整数計画問題

BIP は，数理計画問題のうち決定変数の取りうる値を 0 または 1 に限定した問題である．一般的な BIP では，ナップサック問題のように目的関数の最大化を目指す問題と，一般化割当問題のように最小化を目指す問題が存在する．本稿では，一般性を失わないため，目的関数の最小化を目指す問題について考える．決定要素の集合を $N = \{1, 2, \ldots n\}$, n 次元の 0-1 決定変数を $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ とし，目的関数の係数を $\boldsymbol{c} = (c_1, c_2, \ldots, c_n)$ とする場合に，線形目的関数を持つ BIP は

$$\min_{\boldsymbol{x} \in X} \sum_{i \in N} c_i x_i \tag{1}$$

のように表すことができる．ここで，$X \subseteq \{0, 1\}^n$ は実行可能領域である．最短路問題，割当問題，最小全域木問題などのように多項式時間アルゴリズムが存在する問題や，ナップサック問題，一般化割当問題などのような NP 困難な問題を含め，多くの組合せ最適化問題を問題 (1) のような形で表すことができる．

2.2 最大リグレット最小化 0-1 整数計画問題

MMR-BIP では，目的関数に関わる入力に対して不確定さを考慮する．2.1 節で用いた $\boldsymbol{c}$ の代わりに，各決定要素 $i \in N$ に対応するコストを区間 $[c_i^-, c_i^+]$ 内の任意の値を取りうる不確定なものであると定義し，不確定集合 U が

$$U = \{\boldsymbol{c} \mid c_i \in [c_i^-, c_i^+] \quad \forall i \in N\} \tag{2}$$

となる．U の各要素 $\boldsymbol{c}$ をシナリオと呼ぶ．本研究で扱う最大リグレット最小化基準において，あるシナリオ $\boldsymbol{c}$ における解 $\boldsymbol{x}$ に対するリグレット $r(\boldsymbol{x}, \boldsymbol{c})$ は，シナリオ $\boldsymbol{c}$ のときの最適値との評価値の差として

$$r(\boldsymbol{x}, \boldsymbol{c}) = \sum_{i \in N} c_i x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i y_i \tag{3}$$

のように定義される．MMR-BIP は，最大のリグレット $\max_{\boldsymbol{c} \in U} r(\boldsymbol{x}, \boldsymbol{c})$ を最小にするような BIP の実行可能解 $\boldsymbol{x}$ を求める問題であり，以下のように表すことができる:

$$\min_{\boldsymbol{x} \in X} \max_{\boldsymbol{c} \in U} \left(\sum_{i \in N} c_i x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i y_i \right). \tag{4}$$

ここで，任意の解 $\boldsymbol{x} \in X$ に対して最大のリグレットを与えるシナリオ（最悪シナリオ）$\boldsymbol{c}^{\sigma(\boldsymbol{x})}$ は以下のように定義することができる:

$$c_i^{\sigma(\boldsymbol{x})} = \begin{cases} c_i^+ & (x_i = 1) \\ c_i^- & (x_i = 0) \end{cases} \qquad \forall i \in N. \tag{5}$$

最悪シナリオ (5) を用いると，問題 (4) は

$$\min_{\boldsymbol{x} \in X} \left(\sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} y_i \right) \tag{6}$$

と表すことができる．

3 既存発見的解法

3.1 シナリオ固定法

BIP の実行可能解は，対応する MMR-BIP の実行可能解でもある．シナリオ固定法はある固定したシナリオの下で BIP を解くことにより，MMR-BIP の実行可能解を求める手法である．固定したシナリオが中間値シナリオ $\boldsymbol{c}^{\text{med}}$ $(c_i^{\text{med}} = (c_i^- + c_i^+)/2 \quad \forall i \in N)$ であるとき，シナリオ固定法で得られる解の目的関数値は最適値の 2 倍以内に抑えられることが知られている [2].

3.2 双対置換法

問題 (6) の $\boldsymbol{y}$ に関する部分問題をその線形緩和問題の双対問題として置換することによって，その部分問題が最小化問題から最大化問題に置換され，問題 (6) をひとつの最小化問題として扱うことができる．このような手法を双対置換（dual substitution, DS）法と呼ぶ [3].

3.3 反復双対置換法

DS 法は最大リグレット最小化一般化割当問題に対して近似保証がないことが証明されている [4]. そのため，ロバスト 0-1 整数計画問題に対して DS 法で得られる解を改善する反復双対置換法 (iterated dual substitution method, iDS 法) が提案された [3]．iDS 法ではこれまでの探索で得られた実行可能解を探索空間（iDS 法の次の反復で対象とする問題の実行可能領域）から排除する制約を追加することによって DS 法を反復的に利用する解法である．iDS 法においてこれまで得られた実行可能解集合を $\hat{X}$ とし，新たな反復で $\hat{X}$ に含まれる解を排除するために，DS モデルに追加する 2 種類の制約

- ハミング距離制約
- 最良シナリオ制約

を説明する．

ハミング距離制約

ハミング距離制約は，既知の解 $\hat{\boldsymbol{x}}$ からハミング距離 r 離れることを意味する制約である:

$$\sum_{\substack{i \in N: \\ \hat{x}_i = 0}} x_i + \sum_{\substack{i \in N: \\ \hat{x}_i = 1}} (1 - x_i) \geq r \qquad \forall \hat{\boldsymbol{x}} \in \hat{X}. \tag{7}$$

最良シナリオ制約

最良シナリオ制約は最良シナリオ補題を利用する制約である．最良シナリオとは，解 $\boldsymbol{x}$ にとって最も都合の良いシナリオを指し，その 1 つは

$$c_i^{\phi(\boldsymbol{x})} = \begin{cases} c_i^- & \text{if } x_i = 1 \\ c_i^+ & \text{otherwise} \end{cases} \qquad \forall i \in N, \tag{8}$$

すなわち

$$c_i^{\phi(\boldsymbol{x})} = (c_i^- - c_i^+) x_i + c_i^+ \qquad \forall i \in N \tag{9}$$

と定義できる．最良シナリオ $\boldsymbol{c}^{\phi(\boldsymbol{x})}$ には次のような性質があることが知られている．

補題 1 ([3]). 解 $\boldsymbol{x}$ と解 $\boldsymbol{y}$ において，

$$\sum_{i\in N} c_i^{\phi(\boldsymbol{x})} x_i \geq \sum_{i\in N} c_i^{\phi(\boldsymbol{x})} y_i \tag{10}$$

であるとき，$\max_{\boldsymbol{c}\in U} r(\boldsymbol{x},\boldsymbol{c}) \geq \max_{\boldsymbol{c}\in U} r(\boldsymbol{y},\boldsymbol{c})$ が成り立つ．

補題 1 は解 $\boldsymbol{x}$ にとって最も都合の良いシナリオの下でも自身よりも巡回路のコストが小さい解 $\boldsymbol{y}$ が存在する場合，$\boldsymbol{x}$ は MMR-BIP においても $\boldsymbol{y}$ に支配されることを意味する．補題 1 により，既知の解 $\hat{\boldsymbol{x}}$ と比較して最大リグレットが小さくなる見込みのない解を排除することを

$$\sum_{i\in N} c_i^{\phi(\boldsymbol{x})} x_i < \sum_{i\in N} c_i^{\phi(\boldsymbol{x})} \hat{x}_i \qquad \forall \hat{\boldsymbol{x}} \in \hat{X} \tag{11}$$

のように表現できる．不等式 (11) の $\boldsymbol{c}^{\phi(\hat{\boldsymbol{x}})}$ に (9) を代入してまとめると，最良シナリオ制約は次のようになる:

$$\sum_{\substack{i\in N:\\ \hat{x}_i=1}} c_i^{+} x_i + \sum_{\substack{i\in N:\\ \hat{x}_i=0}} c_i^{-} x_i < \sum_{\substack{i\in N:\\ \hat{x}_i=1}} c_i^{+} \quad \forall \hat{\boldsymbol{x}} \in \hat{X}. \tag{12}$$

4 コア選択法

MMR-BIP を解くことを考えると，その計算時間は決定要素の個数 $|N|$ の増加に伴って大幅に増加していく傾向がある．そこで，新たに N の部分集合（コア）$C \subseteq N$ を考え，C のみを対象として MMR-BIP を解く手法を提案する．提案する CS 法では，まず求解が速い発見的解法を用いて実行可能解を構築し，それらの解に含まれている要素（つまり $x_i = 1$ となるような i）全ての集合をコア C とする．その後，C 内の要素のみを対象に解の質が高い発見的解法（あるいは厳密解法）を実行して新たな実行可能解を求める．2 回目以降の反復では C 内の要素を増やし，拡張した C を対象に解くことを時間制限が来る，または $C = N$ となるまで繰り返す．CS 法は MMR-TSP に対して 3 種類の既存手法と比較しても優れた性能を示していた [1]．

0-1 整数計画問題に拡張するために，先行研究 [1] の手法から改良し，以下のようなアルゴリズム設計を行う．本研究では，コア C の拡張にシナリオ固定法の線形緩和解を用いる．具体的には，得られた解のうち $x_i \geq \epsilon$ となるような i をコアに追加する．なお，パラメータ ϵ は閾値である．そして，C 内の要素に対しては iDS 法を用いる．具体的なアルゴリズムを Algorithm 1 に示す．ここで，

Algorithm 1 コア選択法

コア C を空集合で定義: $C \leftarrow \emptyset$.
実行可能解集合 $\hat{X}$ を空集合で定義: $\hat{X} \leftarrow \emptyset$.
while $C \subsetneq N$ **do**
  MMR-BIP の線形緩和問題をシナリオ固定法を用いて解き，解 $\breve{\boldsymbol{x}}$ を構築する．
  for all $i \in N$ **do**
    if $\breve{x}_i \geq \epsilon$ **then**
      $C \leftarrow C \cup \{i\}$.
    end if
  end for
  C に限定した MMR-BIP を $\hat{X}$ の最良シナリオ制約の下で DS 法で解き，得られた解を $\boldsymbol{x}^*$ とする．
  $\hat{X} \leftarrow \hat{X} \cup \{\boldsymbol{x}^*\}$.
  if $\boldsymbol{x}_{\text{inc}}$ が存在しない or $r_{\max}(\boldsymbol{x}_{\text{inc}}) > r_{\max}(\boldsymbol{x}^*)$ **then**
    $\boldsymbol{x}_{\text{inc}} \leftarrow \boldsymbol{x}^*$.
  end if
end while
return $\boldsymbol{x}_{\text{inc}}$.

$\boldsymbol{x}_{\text{inc}}$ はある時点での暫定解を表し，$r_{\max}(\boldsymbol{x})$ は解 $\boldsymbol{x}$ の最大リグレットを表す．

5 計算実験

5.1 計算環境

本研究では，シナリオ固定法にて使用するシナリオはすべて中間値シナリオ ($c_i^{\text{med}} = (c_i^- + c_i^+)/2 \quad \forall i \in N$) とし，iDS 法を実装する際には先行研究 [3] の結果をもとに最良シナリオ制約を用いて反復を行った．また，CS 法を実装する際には，パラメータ ϵ を $\epsilon = 0.0001$ として実装した．

計算環境は CPU 2.40 GHz Intel Core i5-1135G7，メモリ 8.0 GB である．混合整数計画モデルを解く際に Nuorium Optimizer V26.1.1 を使用し，Python 3.12.7 を用いて実装した．

5.2 実験対象問題

先行研究 [3] で公開されているナップサック問題，多次元ナップサック問題，集合被覆問題，一

般化割当問題の最大リグレット最小化ベンチマーク問題例に対して計算実験を行った．

5.2.1　ナップサック問題

ナップサック問題（knapsack problem, KP）は，n 個のアイテムの集合 $J \in \{1, 2, \ldots, n\}$ と，各アイテム j の重さ a_j および価値 c_j が与えられた時に，与えられるナップサックの耐荷重 b を守りつつ価値が最大になるようにアイテムを選択する問題である．KP はアイテム j を選択する場合に $x_j = 1$ となるような 0-1 変数 x_j を用いることで，以下のように表現することができる:

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} c_j x_j & \\
\text{s.t.} \quad & \sum_{j=1}^{n} a_j x_j \le b & \\
& x_j \in \{0, 1\} & \forall j \in J.
\end{aligned}
$$

5.2.2　多次元ナップサック問題

多次元ナップサック問題（multidimentional knapsack problem, MKP）は，KP を一般化した問題である．n 個のアイテムの集合 $J \in \{1, 2, \ldots, n\}$ と，m 次元の制約の集合 $I \in \{1, 2, \ldots, m\}$，各アイテム j の次元 i に対応するコスト a_{ij} および価値 c_j が与えられた時に，与えられる m 個の容量制限 b_i を守りつつ価値が最大になるようにアイテムを選択する問題である．MKP は KP と同様にアイテム j を選択する場合に $x_j = 1$ となるような 0-1 変数 x_j を用いることで，以下のように表現することができる:

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} c_j x_j & \\
\text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i & \forall i \in I \\
& x_j \in \{0, 1\} & \forall j \in J.
\end{aligned}
$$

5.2.3　集合被覆問題

集合被覆問題（set covering problem, SCP）は，アイテムの集合 $I \in \{1, 2, \ldots n\}$ と I の部分集合 $J \in \{1, 2, \ldots n\}$ が与えられる．各部分集合 j にはコスト c_j が与えられている中で，すべてのアイテムを少なくとも 1 度被覆するような j のの選び方の中で総コストが最小となるものを求める問題である．a_{ij} を部分集合 j がアイテム i を被覆している場合に 1 となるような 0-1 変数であるとすると，SCP は部分集合 j を選択する時に $x_j = 1$ となるような 0-1 変数 x_j を用いることで，以下のように表現することができる:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} & \\
\text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i & \forall i \in I \\
& \sum_{i=1}^{m} x_{ij} = 1 & \forall j \in J \\
& x_{ij} \in \{0, 1\} & \forall j \in J.
\end{aligned}
$$

5.2.4　一般化割当問題

一般化割当問題（generalized assignment problem, GAP）は，タスクの集合 $J \in \{1, 2, \ldots n\}$ と資源の集合 $I \in \{1, 2, \ldots m\}$ が与えられる．タスク j を資源 i に割り付ける際にはコストが c_{ij} 発生し，同時に資源 i を a_{ij} 消費する．その中で，各資源の容量 b_i を守りながら全てのタスクを 1 度実行し，全体のコストを最小化するような割当て方法を見つけることが GAP の目的である．GAP はタスク j を資源 i に割り付ける場合に $x_{ij} = 1$ となるような 0-1 変数 x_{ij} を用いることで，以下のように表現することができる:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} & \\
\text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i & \forall i \in I \\
& \sum_{i=1}^{m} x_{ij} = 1 & \forall j \in J \\
& x_{ij} \in \{0, 1\} & \forall j \in J.
\end{aligned}
$$

5.3　計算実験結果

計算実験の結果を表 1 に示す．表 1 は，CS 法と既存手法である反復双対置換法（iDS）とシナリオ固定法（fix）を，同一の制限時間である 180 秒の下で実行した際に得られた目的関数値に関する勝敗数を示している．表の “–” の左側には CS 法が勝利した（目的関数値の良い解を得た）問題例

表 1: CS 法と既存手法との結果比較

問題	問題例数	iDS		fix	
KP	108	6–0	(97–5)	51–0	(4–53)
MKP	90	9–31	(25–25)	35–28	(4–23)
SCP	90	8–18	(10–54)	86–0	(0–4)
GAP	240	52–8	(96–84)	127–17	(0–96)
合計	528	74–57	(229–168)	299–45	(8–176)

の数を，右側にはCS法が敗北した（目的関数値の悪い解を得た）問題例の数を記載している．また，括弧の内部には，同じ目的関数値の解に辿り着いた問題例の中で該当の解に辿り着くまでの時間についての勝敗を示している．例えば，KPでは計108題の問題例に対して計算実験を行い，そのうちCS法は6題でiDSよりも優れた解を得ている．一方で，iDSよりも悪い目的関数値しか得られなかった問題例は一つも存在しなかった．また，同じ目的関数値の解に辿り着いた問題例でも97題でiDSよりも早く該当の解を得ることができ，5題は早さにおいてiDSに敗れた．

計算実験の結果から，KPおよびGAPに対してCS法はiDS法とシナリオ固定法よりも優越していることがわかり，特にGAPに対してはその傾向が顕著である．MKPに対してはiDS法と比較して劣っているが，一部の問題例には制限時間内にiDS法よりも優れた解を見つけることができている．

6　まとめ

本研究では最大リグレット最小化0-1整数計画問題に対してコア選択法を適用し，計算実験を通して性能を確認した．計算実験の結果から，特に一般化割当問題に対しては，既存手法である反復双対置換法やシナリオ固定法と比較して，より優れた性能を確認できた．

今後の展望として，コア選択法をより広範な問題に対しても適用しその性能を確認することと，パラメータの選択や内包する手法の検討などを通じて更なる性能向上の方法について考えていきたい．

参考文献

[1] Kazuki Hasegawa and Wei Wu. A heuristic approach for the robust traveling salesman problem. In *2022 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pp. 561–565, 2022.

[2] Adam Kasperski and Paweł Zieliński. An approximation algorithm for interval data min-max regret combinatorial optimization problems. *Information Processing Letters*, Vol. 97, No. 5, pp. 177–180, 2006.

[3] Wei Wu, Manuel Iori, Silvano Martello, and Mutsunori Yagiura. An iterated dual substitution approach for binary integer programming problems under the min-max regret criterion. *INFORMS Journal on Computing*, Vol. 34, No. 5, pp. 2523–2539, 2022.

[4] Wei Wu, Manuel Iori, Silvano Martello, and Mutsunori Yagiura. Exact and heuristic algorithms for the interval min–max regret generalized assignment problem. *Computers & Industrial Engineering*, Vol. 125, pp. 98–110, 2018.

謝辞

提供いただきましたNuorium Optimizerによって，非常に有意義な研究を行うことができました．株式会社NTTデータ数理システム様に心より感謝申し上げます．

A　付録

表2–表3にKPの，表4–表5にMKPの，表6–表8にSCPの，表9–表12にGAPの詳細な計算実験結果を示す．各行がそれぞれの問題例に対応しており，各列はそれぞれコア選択法（CS），反復双対置換法（iDS），シナリオ固定法（fix）に対応している．各手法に対して，obj列が制限時

間（180 秒）内に得られた最も良い解の目的関数値（最大リグレット）に，time 列がその解に辿り着くまでに要した時間（秒）に，iter 列がその解に辿り着くまでに要した反復回数に対応している．なお，シナリオ固定法は反復を行わないため iter 列は存在しない．

表 2: KP に対する実験結果 (1)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
6-50-01-45-10	760	2.556	10	760	8.086	2	760	0.212
6-50-01-45-20	1691	4.514	8	1691	20.050	5	1698	0.086
6-50-01-45-30	2559	0.274	3	2559	2.469	1	2559	0.100
6-50-01-50-10	740	0.611	5	740	2.525	1	740	0.094
6-50-01-50-20	1705	0.080	1	1705	5.079	1	1707	0.105
6-50-01-50-30	2560	0.074	1	2560	2.915	1	2560	0.097
6-50-01-55-10	743	1.236	8	743	2.299	1	743	0.108
6-50-01-55-20	1706	0.652	4	1706	4.055	1	1706	0.087
6-50-01-55-30	2535	0.535	4	2535	2.798	1	2544	0.087
6-50-10-45-10	7851	1.097	7	7851	3.240	1	7851	0.094
6-50-10-45-20	12921	0.220	3	12921	2.552	1	12993	0.102
6-50-10-45-30	23190	0.734	5	23190	3.384	1	23277	0.091
6-50-10-50-10	7904	7.203	10	7904	3.525	1	7919	0.129
6-50-10-50-20	13240	0.690	4	13240	10.479	3	13250	0.111
6-50-10-50-30	24190	1.031	6	24190	4.840	1	24190	0.105
6-50-10-55-10	7589	5.472	8	7589	12.889	5	7603	0.123
6-50-10-55-20	13073	0.392	4	13073	19.004	6	13120	0.125
6-50-10-55-30	24324	0.157	2	24324	4.878	1	24324	0.097
6-60-01-45-10	991	0.370	3	991	4.546	1	991	0.094
6-60-01-45-20	2213	0.667	4	2213	24.317	1	2213	0.089
6-60-01-45-30	3157	2.375	10	3157	23.942	2	3159	0.117
6-60-01-50-10	998	0.323	3	998	26.562	3	998	0.093
6-60-01-50-20	2240	0.209	2	2240	60.856	1	2240	0.081
6-60-01-50-30	3175	0.058	1	3175	15.744	1	3175	0.112
6-60-01-55-10	1005	0.712	5	1005	11.086	1	1005	0.101
6-60-01-55-20	2245	0.396	3	2245	57.463	1	2251	0.091
6-60-01-55-30	3163	2.209	8	3163	18.798	1	3163	0.101
6-60-10-45-10	9667	1.166	5	9667	8.423	1	9667	0.096
6-60-10-45-20	16849	7.683	8	16849	26.555	5	16878	0.094
6-60-10-45-30	27417	4.846	11	27417	12.786	1	27518	0.121
6-60-10-50-10	9774	0.330	3	9774	12.197	1	9774	0.179
6-60-10-50-20	17333	16.303	12	17333	88.164	8	17348	0.131
6-60-10-50-30	28524	0.957	5	28524	34.128	1	28524	0.116
6-60-10-55-10	9441	3.930	7	9441	11.852	2	9462	0.102
6-60-10-55-20	17123	3.583	6	17123	15.284	2	17123	0.089
6-60-10-55-30	28476	0.507	5	28476	26.317	1	28476	0.146
6-70-01-45-10	1090	0.492	4	1090	19.141	1	1093	0.155
6-70-01-45-20	2554	0.611	3	2560	180.000	1	2554	0.097
6-70-01-45-30	3481	0.537	4	3481	39.661	1	3481	0.085
6-70-01-50-10	1109	0.503	4	1109	36.830	1	1109	0.102
6-70-01-50-20	2597	0.528	4	2597	180.000	1	2597	0.091
6-70-01-50-30	3501	0.690	5	3501	57.479	1	3501	0.090
6-70-01-55-10	1117	1.926	7	1117	52.592	1	1117	0.092
6-70-01-55-20	2601	0.572	4	2608	180.000	1	2601	0.089
6-70-01-55-30	3475	0.663	4	3475	116.607	2	3478	0.108
6-70-10-45-10	11060	0.595	5	11060	58.227	1	11078	0.106
6-70-10-45-20	18673	45.894	17	18673	179.196	6	18697	0.090
6-70-10-45-30	31130	24.164	16	31130	28.366	1	31301	0.132
6-70-10-50-10	11054	0.738	5	11054	50.759	1	11054	0.158
6-70-10-50-20	18890	0.642	3	18890	52.663	1	18959	0.100
6-70-10-50-30	32207	0.488	2	32207	135.632	3	32230	0.103
6-70-10-55-10	10722	5.731	6	10722	25.614	1	10722	0.107
6-70-10-55-20	18653	8.500	9	18653	53.278	2	18759	0.105
6-70-10-55-30	32067	1.079	5	32067	137.583	1	32067	0.099

表 3: KP に対する実験結果 (2)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
7-50-01-45-10	833	0.431	5	833	2.716	1	833	0.082
7-50-01-45-20	1661	2.735	7	1661	9.814	3	1669	0.114
7-50-01-45-30	2346	0.062	1	2346	3.514	1	2346	0.082
7-50-01-50-10	810	0.619	6	810	4.430	2	810	0.089
7-50-01-50-20	1679	0.649	5	1679	7.738	2	1683	0.104
7-50-01-50-30	2382	0.703	8	2382	2.987	1	2382	0.094
7-50-01-55-10	803	0.665	5	803	2.801	1	803	0.105
7-50-01-55-20	1674	0.301	3	1674	3.181	1	1674	0.088
7-50-01-55-30	2385	0.547	5	2385	5.967	2	2385	0.118
7-50-10-45-10	7519	0.959	5	7519	15.061	6	7530	0.112
7-50-10-45-20	13464	6.671	11	13464	9.112	3	13476	0.091
7-50-10-45-30	25515	7.081	10	25515	4.900	1	25682	0.105
7-50-10-50-10	7600	8.670	10	7600	2.927	1	7600	0.104
7-50-10-50-20	13920	2.173	5	13920	5.666	1	13947	0.107
7-50-10-50-30	26163	0.226	2	26163	7.885	1	26163	0.088
7-50-10-55-10	7372	4.008	8	7372	4.948	2	7377	0.095
7-50-10-55-20	13728	3.261	9	13728	21.273	5	13728	0.124
7-50-10-55-30	25509	0.284	3	25509	3.785	1	25509	0.093
7-60-01-45-10	1061	0.373	4	1061	9.415	1	1061	0.084
7-60-01-45-20	2193	0.506	4	2193	21.113	1	2197	0.089
7-60-01-45-30	3059	8.965	11	3059	7.322	1	3073	0.103
7-60-01-50-10	1073	0.211	2	1073	13.209	1	1073	0.100
7-60-01-50-20	2222	1.986	7	2222	131.494	3	2223	0.109
7-60-01-50-30	3117	2.156	6	3117	91.114	4	3118	0.097
7-60-01-55-10	1072	0.532	4	1072	6.079	1	1072	0.100
7-60-01-55-20	2220	1.657	6	2220	63.623	2	2220	0.100
7-60-01-55-30	3120	0.359	3	3120	19.050	1	3120	0.072
7-60-10-45-10	9335	11.001	10	9335	6.578	1	9355	0.105
7-60-10-45-20	17262	18.221	15	17262	47.458	6	17327	0.097
7-60-10-45-30	30203	11.989	10	30203	23.731	1	30233	0.077
7-60-10-50-10	9499	6.440	8	9499	9.608	1	9502	0.112
7-60-10-50-20	17964	1.057	4	17964	18.147	1	17964	0.084
7-60-10-50-30	30978	0.384	2	30978	63.640	1	30978	0.088
7-60-10-55-10	9252	0.922	4	9252	16.416	3	9322	0.126
7-60-10-55-20	17776	8.805	9	17776	76.678	6	17780	0.077
7-60-10-55-30	30456	0.267	3	30456	37.610	1	30456	0.124
7-70-01-45-10	1157	0.521	4	1157	26.501	1	1157	0.097
7-70-01-45-20	2532	0.498	3	2532	104.915	1	2532	0.116
7-70-01-45-30	3326	10.016	12	3326	36.076	1	3330	0.112
7-70-01-50-10	1179	0.069	1	1179	52.811	1	1179	0.088
7-70-01-50-20	2562	0.154	2	2562	141.498	1	2562	0.084
7-70-01-50-30	3399	0.585	4	3399	87.918	1	3399	0.094
7-70-01-55-10	1183	0.739	5	1183	179.030	3	1185	0.093
7-70-01-55-20	2578	1.037	3	2582	180.000	1	2581	0.096
7-70-01-55-30	3412	7.969	9	3412	179.559	2	3413	0.091
7-70-10-45-10	10698	10.028	9	10698	87.635	2	10719	0.124
7-70-10-45-20	19238	24.714	14	19238	60.991	1	19301	0.110
7-70-10-45-30	33427	20.004	12	33427	86.666	1	33606	0.109
7-70-10-50-10	10747	3.380	8	10747	86.733	2	10765	0.152
7-70-10-50-20	19557	7.387	6	19557	131.967	1	19557	0.107
7-70-10-50-30	34350	2.081	4	34357	180.000	1	34357	0.102
7-70-10-55-10	10466	13.313	8	10466	34.252	2	10475	0.113
7-70-10-55-20	19273	1.066	4	19282	98.429	2	19373	0.091
7-70-10-55-30	34054	1.154	5	34054	180.000	1	34054	0.087

表 4: MKP に対する実験結果 (1)

問題例	CS obj	CS time	CS iter	iDS obj	iDS time	iDS iter	fix obj	fix time
0510010-01	662	20.297	7	662	1.598	1	662	1.040
0510010-02	663	12.641	4	663	3.606	2	663	4.496
0510010-03	426	22.302	5	426	24.748	5	457	3.325
0510010-04	809	44.651	5	809	10.355	1	809	4.797
0510010-05	737	0.211	1	737	1.437	1	755	3.922
0510010-06	482	26.679	5	482	7.540	2	482	1.727
0510010-07	603	8.610	3	603	1.489	1	603	0.533
0510010-08	406	0.257	1	406	0.665	1	482	1.802
0510010-09	469	6.847	3	469	4.348	2	499	2.951
0510010-10	587	78.898	6	587	26.796	2	644	2.449
0510020-01	1963	38.057	7	1963	62.890	4	1968	3.838
0510020-02	1434	27.159	6	1434	23.660	3	1434	0.987
0510020-03	1363	0.160	1	1363	1.655	1	1474	3.687
0510020-04	2190	21.837	4	2190	9.723	1	2240	0.666
0510020-05	1732	3.920	2	1732	8.469	1	1732	4.812
0510020-06	1364	9.202	2	1364	0.755	1	1364	0.863
0510020-07	1679	14.106	4	1679	7.594	1	1742	1.873
0510020-08	1577	9.558	3	1577	2.130	1	1577	1.000
0510020-09	1393	0.150	1	1393	11.627	2	1424	0.744
0510020-10	1707	15.924	4	1707	12.365	2	1707	0.955
0510030-01	2666	160.812	12	2666	35.361	2	2672	0.724
0510030-02	3187	12.546	3	3187	28.360	1	3327	2.811
0510030-03	2912	10.333	3	2912	25.416	2	2912	0.390
0510030-04	3927	30.672	6	3927	115.658	2	3927	3.182
0510030-05	3046	3.829	2	3046	15.922	1	3109	3.940
0510030-06	3122	16.017	3	3122	11.738	1	3122	1.180
0510030-07	3445	9.243	3	3445	25.061	2	3500	0.858
0510030-08	2556	24.504	2	2556	3.188	1	2644	1.063
0510030-09	2350	7.412	2	2350	1.723	1	2423	1.378
0510030-10	3022	35.421	5	3022	16.123	1	3022	1.800
0525010-01	1033	144.982	2	940	179.392	2	973	18.314
0525010-02	1097	1.266	1	993	117.538	1	993	23.491
0525010-03	978	1.015	1	959	111.324	1	959	15.427
0525010-04	1411	0.910	1	1321	180.000	1	1321	159.300
0525010-05	1068	1.404	1	919	92.262	1	937	41.269
0525010-06	1381	1.140	1	1237	180.000	1	1209	28.657
0525010-07	1218	107.048	2	1218	180.000	1	1263	75.232
0525010-08	1117	159.881	2	1073	128.742	1	1073	23.542
0525010-09	1253	0.740	1	1154	121.241	1	1154	40.513
0525010-10	1115	108.000	3	1086	83.868	1	1086	40.481
0525020-01	3404	69.652	2	3490	180.000	1	3404	8.686
0525020-02	3496	87.483	2	3531	180.000	1	3492	13.727
0525020-03	4205	0.872	1	4205	180.000	1	4253	19.568
0525020-04	5272	0.705	1	5067	180.000	1	5025	134.984
0525020-05	3362	0.787	1	3359	180.000	1	3317	13.525
0525020-06	4071	168.638	2	4005	180.000	1	4020	116.513
0525020-07	3482	0.499	1	3289	180.000	1	3318	31.282
0525020-08	3994	0.886	1	3803	180.000	1	3837	93.802
0525020-09	3319	0.857	1	3356	180.000	1	3324	10.529
0525020-10	3176	1.279	1	3168	180.000	1	3104	18.833
0525030-01	7838	1.214	1	7801	180.000	1	7806	52.701
0525030-02	7122	174.451	2	7162	180.000	1	7011	24.155
0525030-03	8258	170.069	2	8207	180.000	1	8202	57.343
0525030-04	7610	154.691	3	7736	180.000	1	7601	26.090
0525030-05	6690	98.967	2	6698	180.000	1	6678	40.211
0525030-06	7178	1.768	1	7177	180.000	1	7055	12.617
0525030-07	6828	2.708	1	6762	180.000	1	6754	14.321
0525030-08	8441	61.595	2	8629	180.000	1	8441	10.626
0525030-09	7172	131.139	3	7172	180.000	1	7172	5.119
0525030-10	6727	1.298	1	6690	180.000	1	6642	22.094

表 5: MKP に対する実験結果 (2)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
1010010-01	809	125.580	5	798	25.391	1	847	20.949
1010010-02	615	0.313	1	474	9.219	1	474	5.158
1010010-03	789	35.046	2	759	21.248	1	789	16.789
1010010-04	1104	48.620	2	1093	66.480	1	1148	112.603
1010010-05	753	53.166	2	753	150.404	2	881	23.107
1010010-06	825	72.616	2	807	53.956	1	807	56.722
1010010-07	650	113.894	5	650	96.419	3	699	9.463
1010010-08	684	49.471	2	684	13.939	1	731	12.953
1010010-09	613	92.441	4	611	40.135	2	613	5.361
1010010-10	593	0.516	1	593	10.734	1	593	5.480
1010020-01	1827	102.337	3	1827	125.935	2	1877	21.469
1010020-02	1666	0.642	1	1666	20.197	1	1746	13.139
1010020-03	1671	139.951	2	1671	12.873	1	1671	11.319
1010020-04	2052	0.310	1	2061	179.499	2	2074	77.382
1010020-05	1870	19.250	2	1870	14.728	1	1870	1.299
1010020-06	2310	100.594	2	2232	53.423	1	2282	11.282
1010020-07	1992	0.690	1	1969	40.630	2	2019	7.244
1010020-08	1997	59.875	3	1997	23.476	1	1997	7.392
1010020-09	1652	0.606	1	1652	21.846	2	1687	2.860
1010020-10	2346	0.814	1	2230	32.522	1	2318	9.055
1010030-01	3027	42.203	2	3027	20.652	1	3027	1.663
1010030-02	3219	0.455	1	3219	29.129	1	3306	25.019
1010030-03	3200	0.548	1	3200	17.052	1	3200	5.925
1010030-04	4068	179.210	5	4068	180.000	1	4069	17.791
1010030-05	3385	179.549	3	3336	31.297	1	3530	8.542
1010030-06	4284	7.055	2	4328	175.996	1	4415	18.991
1010030-07	2637	30.531	2	2628	15.778	1	2845	5.911
1010030-08	3183	56.369	3	3183	72.191	2	3268	11.562
1010030-09	2937	141.854	4	2937	15.211	1	2937	0.752
1010030-10	3601	1.228	1	3601	78.245	1	3601	4.531

表 6: SCP に対する実験結果 (1)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
B40110	21	1.065	1	21	1.318	1	21	0.695
B40130	67	9.847	5	67	1.316	1	71	0.464
B40150	124	25.485	12	124	9.066	4	136	0.445
B40210	23	1.130	1	23	1.002	1	23	0.460
B40230	93	8.574	5	93	12.494	5	95	0.455
B40250	163	2.865	2	163	7.814	3	175	0.453
B40310	12	2.785	2	12	0.970	1	14	0.449
B40330	86	11.523	6	86	1.833	1	92	0.474
B40350	190	1.271	1	190	38.391	7	192	0.442
B40410	20	1.176	1	20	1.096	1	21	0.461
B40430	85	3.101	2	85	1.640	1	87	0.446
B40450	160	7.405	4	160	5.034	1	170	0.568
B40510	13	1.020	1	13	0.929	1	13	0.445
B40530	73	16.334	8	73	16.137	7	77	0.500
B40550	165	5.845	3	165	2.783	1	171	0.502
B40610	16	8.452	5	16	4.478	3	19	0.542
B40630	84	19.344	9	84	15.031	7	86	0.469
B40650	178	50.738	16	178	25.905	5	185	0.453
B40710	17	1.218	1	17	0.938	1	19	0.502
B40730	59	21.779	10	59	3.455	2	68	0.517
B40750	104	22.929	12	104	3.366	2	115	0.462
B40810	24	8.697	5	24	1.030	1	27	0.506
B40830	54	6.977	4	54	5.078	3	56	0.472
B40850	155	3.111	2	155	3.481	1	160	0.450
B40910	33	1.248	1	33	1.080	1	34	0.463
B40930	99	14.051	6	99	9.799	4	103	0.443
B40950	263	1.522	1	263	13.374	1	266	0.475
B41010	17	1.099	1	17	0.890	1	18	0.488
B41030	63	4.787	3	63	1.285	1	63	0.465
B41050	141	14.042	8	141	23.818	8	151	0.504

表 7: SCP に対する実験結果 (2)

問題例	CS obj	CS time	CS iter	iDS obj	iDS time	iDS iter	fix obj	fix time
K401-1	14467	179.835	12	14440	34.828	1	15596	1.234
K401-2	16357	180.000	6	16247	180.000	1	16863	1.316
K401-3	12974	180.000	9	12974	106.414	1	13294	0.547
K402-1	14650	171.528	7	14235	121.592	1	15578	0.874
K402-2	16971	180.000	5	16335	180.000	1	18522	1.450
K402-3	14306	29.100	2	14327	180.000	1	15158	0.978
K403-1	15139	83.639	8	15139	66.354	1	15968	0.610
K403-2	16898	141.373	6	16625	180.000	1	17286	1.170
K403-3	13613	66.813	7	13613	46.343	1	14831	0.676
K404-1	13472	129.988	15	13472	18.061	1	15069	0.576
K404-2	15969	160.780	5	15876	180.000	1	17252	1.238
K404-3	14881	101.612	10	14898	179.851	2	15494	0.663
K405-1	16537	158.216	4	16345	180.000	1	17670	1.268
K405-2	16337	31.337	3	16355	180.000	1	17223	1.619
K405-3	14549	68.762	5	14554	180.000	1	15568	1.067
K406-1	14367	89.487	5	14354	180.000	1	15259	1.161
K406-2	14998	136.160	4	14555	180.000	1	16319	1.874
K406-3	14096	150.703	6	14093	180.000	1	14915	0.910
K407-1	14997	172.650	8	15112	180.000	1	15627	0.999
K407-2	15979	179.706	6	16229	180.000	1	16735	1.055
K407-3	15268	79.916	6	15262	179.745	2	15842	0.844
K408-1	13739	165.147	7	13654	139.682	1	14547	1.021
K408-2	15843	180.000	9	15843	177.933	1	17413	0.792
K408-3	13574	3.892	1	13566	59.428	1	14096	0.575
K409-1	15006	82.542	6	15063	180.000	1	15680	1.032
K409-2	13971	69.167	4	13971	180.000	1	14963	0.905
K409-3	14686	53.046	5	14414	109.543	1	15399	0.833
K410-1	15397	178.590	9	15287	180.000	1	16495	0.888
K410-2	16131	142.017	7	15903	180.000	1	17072	1.124
K410-3	16780	89.080	4	16886	180.000	1	16922	1.097

表 8: SCP に対する実験結果 (3)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
M401-1	3160	23.372	8	3160	8.698	1	3304	0.484
M401-2	3495	21.615	11	3495	8.740	1	3636	0.487
M401-3	3382	44.063	13	3382	6.738	1	3491	0.484
M402-1	3610	91.183	26	3610	9.739	1	3788	0.488
M402-2	2986	25.981	10	2986	10.050	1	3188	0.545
M402-3	4294	146.741	28	4294	10.004	1	4467	0.505
M403-1	3233	77.132	29	3233	7.208	1	3375	0.488
M403-2	4205	41.411	14	4205	6.154	1	4326	0.501
M403-3	3589	15.906	7	3589	18.568	2	3884	0.495
M404-1	3628	74.926	26	3628	5.130	1	3884	0.492
M404-2	3954	38.317	13	3954	7.086	1	4148	0.485
M404-3	2957	11.887	7	2957	10.347	1	3115	0.481
M405-1	3546	7.512	3	3546	7.252	1	3980	0.499
M405-2	3589	22.862	10	3589	7.295	1	3931	0.469
M405-3	3698	45.639	17	3698	7.364	1	3856	0.494
M406-1	3549	60.417	22	3549	10.411	1	3692	0.480
M406-2	2975	18.307	9	2975	28.617	4	3000	0.502
M406-3	3199	14.387	7	3199	4.991	1	3296	0.505
M407-1	3188	22.620	10	3115	5.329	1	3320	0.462
M407-2	4732	59.773	15	4674	87.605	12	4840	0.669
M407-3	4136	27.064	11	4136	6.151	1	4367	0.491
M408-1	3802	166.020	46	3802	8.905	1	3882	0.468
M408-2	3302	142.907	44	3302	17.516	2	3512	0.476
M408-3	3147	11.431	6	3147	8.014	1	3339	0.507
M409-1	3602	14.302	8	3602	7.328	1	3923	0.499
M409-2	4222	22.714	11	4222	7.403	1	4436	0.472
M409-3	3365	26.679	12	3365	6.612	1	3494	0.500
M410-1	3000	36.520	18	3000	3.656	1	3200	0.476
M410-2	4340	51.128	20	4340	8.499	1	4439	0.455
M410-3	3235	33.123	17	3235	5.059	1	3323	0.469

表 9: GAP に対する実験結果 (1)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
a0504010-1	16	0.395	2	16	0.548	1	16	0.121
a0504010-2	18	0.184	1	18	0.155	1	18	0.143
a0504010-3	5	0.172	1	5	0.135	1	5	0.138
a0504010-4	18	0.462	2	18	0.796	1	18	0.138
a0504010-5	12	0.176	1	12	0.287	1	13	0.137
a0504025-1	80	1.910	6	80	5.780	2	82	0.144
a0504025-2	67	0.175	1	67	1.495	1	67	0.134
a0504025-3	59	0.186	1	59	1.586	1	59	0.138
a0504025-4	71	0.435	2	71	4.384	1	71	0.142
a0504025-5	84	0.958	4	84	2.180	2	89	0.134
a0504050-1	261	0.380	2	261	1.620	1	261	0.121
a0504050-2	204	0.166	1	204	1.686	1	205	0.120
a0504050-3	164	0.166	1	164	0.748	1	164	0.140
a0504050-4	227	0.168	1	227	1.719	1	227	0.120
a0504050-5	219	0.563	3	219	1.235	1	219	0.137
a0508010-1	23	0.631	1	23	2.501	1	23	0.266
a0508010-2	29	2.103	3	29	1.658	1	29	0.253
a0508010-3	19	3.213	5	19	8.002	2	20	0.272
a0508010-4	27	1.297	2	27	1.206	1	28	0.257
a0508010-5	27	1.261	2	27	0.782	1	27	0.294
a0508025-1	141	0.521	1	141	3.120	1	142	0.282
a0508025-2	108	1.286	2	108	2.501	1	108	0.250
a0508025-3	91	1.894	3	91	2.437	1	92	0.270
a0508025-4	116	0.485	1	116	3.284	1	116	0.250
a0508025-5	105	0.501	1	105	2.889	1	105	0.251
a0508050-1	427	1.342	2	427	51.259	1	427	0.252
a0508050-2	388	0.515	1	388	28.347	1	388	0.256
a0508050-3	487	0.520	1	487	39.197	1	487	0.248
a0508050-4	390	4.963	6	390	10.976	1	395	0.254
a0508050-5	418	6.381	8	418	13.475	1	421	0.268
a1004010-1	14	2.943	7	14	0.439	1	14	0.195
a1004010-2	16	1.458	3	16	0.455	1	16	0.182
a1004010-3	14	0.332	1	14	0.407	1	14	0.188
a1004010-4	13	0.905	2	13	0.533	1	15	0.187
a1004010-5	16	4.135	7	16	0.934	1	17	0.184
a1004025-1	78	15.229	12	78	1.371	1	80	0.184
a1004025-2	54	0.991	2	54	0.912	1	56	0.193
a1004025-3	64	2.228	5	64	3.343	1	64	0.204
a1004025-4	45	0.880	2	45	3.581	1	45	0.183
a1004025-5	73	1.150	3	73	8.833	4	75	0.196
a1004050-1	210	31.035	19	210	34.717	9	214	0.188
a1004050-2	182	0.868	2	182	3.500	1	185	0.189
a1004050-3	206	1.537	3	206	5.189	1	211	0.182
a1004050-4	169	3.967	7	169	1.749	1	170	0.203
a1004050-5	205	3.375	8	205	12.450	3	208	0.189
a1008010-1	22	4.433	3	22	1.681	1	23	0.505
a1008010-2	34	4.314	3	34	7.268	1	34	0.546
a1008010-3	39	1.033	1	39	3.359	1	39	0.502
a1008010-4	16	1.017	1	16	1.435	1	16	0.486
a1008010-5	29	2.266	2	29	2.249	1	29	0.511
a1008025-1	87	1.019	1	87	3.033	1	87	0.523
a1008025-2	117	5.858	4	117	4.256	1	120	0.520
a1008025-3	126	2.620	2	126	9.030	1	126	0.542
a1008025-4	113	9.581	6	113	11.366	1	114	0.504
a1008025-5	109	2.687	2	109	3.531	1	109	0.521
a1008050-1	367	1.030	1	367	71.810	1	367	0.522
a1008050-2	433	10.867	5	433	130.199	1	438	0.506
a1008050-3	419	25.182	9	419	132.920	1	431	0.521
a1008050-4	321	10.235	6	321	24.483	3	324	0.520
a1008050-5	376	90.746	22	376	24.859	1	390	0.523

表 10: GAP に対する実験結果 (2)

問題例	CS obj	CS time	CS iter	iDS obj	iDS time	iDS iter	fix obj	fix time
b0504010-1	18	3.587	9	18	2.145	2	18	0.448
b0504010-2	19	2.104	6	19	0.722	1	22	0.331
b0504010-3	12	1.731	5	12	2.051	1	12	0.235
b0504010-4	25	2.679	8	25	2.714	4	28	0.404
b0504010-5	22	4.655	10	22	1.911	2	23	0.454
b0504025-1	84	14.336	10	84	1.888	1	84	0.520
b0504025-2	64	3.441	8	64	0.925	1	64	0.329
b0504025-3	38	0.785	4	38	0.468	1	45	0.273
b0504025-4	66	15.738	15	66	12.856	10	66	0.357
b0504025-5	77	7.639	10	77	1.594	1	77	0.643
b0504050-1	248	3.812	7	248	2.331	1	252	0.457
b0504050-2	227	0.599	3	227	1.433	1	227	0.393
b0504050-3	108	0.379	2	108	0.718	1	108	0.241
b0504050-4	215	1.852	6	215	2.343	1	215	0.375
b0504050-5	191	1.001	5	191	2.188	1	200	0.565
b0508010-1	31	11.687	8	31	4.533	1	33	0.663
b0508010-2	29	31.020	12	29	3.331	1	29	0.655
b0508010-3	30	1.094	2	30	2.363	1	30	1.033
b0508010-4	45	11.275	8	45	5.863	1	47	1.410
b0508010-5	31	5.244	5	31	2.226	1	31	0.694
b0508025-1	111	5.943	5	111	37.800	1	111	1.296
b0508025-2	122	3.316	5	122	6.124	1	128	0.439
b0508025-3	147	66.009	16	147	29.933	1	154	2.086
b0508025-4	131	4.374	4	131	25.683	1	138	1.716
b0508025-5	107	4.736	8	107	8.422	1	110	0.939
b0508050-1	345	147.379	18	346	88.588	1	353	0.641
b0508050-2	415	22.019	10	415	113.037	1	416	0.470
b0508050-3	402	22.174	12	402	136.619	1	406	1.263
b0508050-4	421	33.218	13	421	180.000	1	426	1.065
b0508050-5	398	24.610	15	398	180.000	1	398	0.596
b1004010-1	21	7.980	9	21	5.014	4	22	0.351
b1004010-2	16	4.734	10	16	2.884	3	17	0.326
b1004010-3	16	21.097	16	16	0.893	1	16	0.401
b1004010-4	14	4.339	11	14	5.801	1	14	0.408
b1004010-5	28	50.648	18	28	6.629	4	28	0.422
b1004025-1	102	52.359	24	102	14.090	3	104	0.423
b1004025-2	71	26.859	18	71	5.551	3	72	0.224
b1004025-3	63	26.092	17	63	3.764	2	63	0.342
b1004025-4	58	48.802	28	58	1.363	1	65	0.330
b1004025-5	76	5.690	10	76	14.859	3	76	0.449
b1004050-1	193	4.500	6	193	11.862	2	206	0.267
b1004050-2	189	3.685	7	189	2.875	1	198	0.230
b1004050-3	187	12.421	14	187	119.024	7	197	0.377
b1004050-4	199	18.270	18	199	6.809	1	203	0.359
b1004050-5	195	18.358	14	195	87.728	2	200	0.709
b1008010-1	24	6.761	6	24	2.766	1	24	0.764
b1008010-2	40	17.359	8	40	34.880	2	40	3.914
b1008010-3	42	170.651	31	43	49.349	3	44	1.605
b1008010-4	25	61.953	17	25	6.157	2	28	1.042
b1008010-5	25	27.583	12	25	4.325	1	29	1.275
b1008025-1	91	22.093	11	91	7.565	1	95	0.899
b1008025-2	170	51.463	16	175	180.000	1	175	2.902
b1008025-3	124	19.996	9	124	76.064	2	124	1.423
b1008025-4	116	34.811	11	116	102.131	1	119	1.256
b1008025-5	116	7.824	5	116	99.070	1	116	0.947
b1008050-1	337	27.738	10	338	180.000	1	338	1.113
b1008050-2	443	29.780	11	453	180.000	1	444	2.228
b1008050-3	411	141.623	14	417	180.000	1	422	2.196
b1008050-4	379	35.233	11	381	180.000	1	381	0.847
b1008050-5	346	118.467	19	348	180.000	1	356	0.905

表 11: GAP に対する実験結果 (3)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
c0504010-1	18	0.820	4	18	0.550	1	18	0.469
c0504010-2	16	1.000	5	16	0.535	1	16	0.292
c0504010-3	14	3.034	5	14	1.373	1	14	0.407
c0504010-4	17	2.507	8	17	2.217	1	17	0.320
c0504010-5	23	4.982	11	23	0.837	1	23	0.493
c0504025-1	47	4.088	9	47	1.188	1	47	0.422
c0504025-2	60	8.121	11	60	1.112	1	60	0.229
c0504025-3	103	7.213	10	103	1.618	1	103	0.344
c0504025-4	83	1.018	5	83	0.609	1	83	0.606
c0504025-5	85	7.408	8	85	1.728	1	93	0.467
c0504050-1	160	21.780	14	160	1.818	1	160	0.467
c0504050-2	202	3.078	8	202	3.204	1	212	0.225
c0504050-3	217	23.687	11	217	4.392	1	217	0.727
c0504050-4	173	17.719	15	173	1.352	1	176	0.361
c0504050-5	269	0.786	4	269	15.415	1	269	0.504
c0508010-1	25	27.327	11	25	1.296	1	25	0.854
c0508010-2	28	2.234	4	28	2.272	1	28	0.940
c0508010-3	35	10.048	8	35	3.494	1	35	1.229
c0508010-4	39	10.499	11	39	3.295	1	43	1.964
c0508010-5	28	9.847	7	28	2.204	1	28	1.266
c0508025-1	132	0.500	1	132	3.372	1	132	0.313
c0508025-2	112	18.710	11	112	4.813	1	112	1.113
c0508025-3	131	1.660	3	131	20.155	1	131	1.234
c0508025-4	141	3.935	6	141	130.367	3	149	0.752
c0508025-5	113	16.225	8	113	42.523	2	120	1.106
c0508050-1	404	33.594	10	404	180.000	1	404	0.875
c0508050-2	389	19.293	9	389	110.385	1	393	0.550
c0508050-3	418	4.970	8	419	180.000	1	418	1.237
c0508050-4	503	2.332	4	503	180.000	1	503	0.700
c0508050-5	365	17.137	11	365	82.474	1	365	0.620
c1004010-1	20	40.422	28	20	9.411	8	23	0.518
c1004010-2	15	19.092	15	15	0.961	1	17	0.371
c1004010-3	24	18.056	15	24	2.891	2	24	0.392
c1004010-4	26	26.563	17	26	0.998	1	26	0.516
c1004010-5	25	48.735	19	25	0.932	1	28	0.503
c1004025-1	67	21.579	19	67	10.223	5	72	0.303
c1004025-2	58	13.093	18	58	7.510	4	59	0.440
c1004025-3	70	13.382	11	70	1.848	1	75	0.278
c1004025-4	65	59.204	29	65	47.937	17	70	0.663
c1004025-5	83	21.586	15	83	20.495	6	85	0.344
c1004050-1	212	152.566	37	214	108.157	5	212	0.470
c1004050-2	219	5.789	11	219	6.250	1	219	0.299
c1004050-3	197	34.050	19	197	44.717	3	199	0.550
c1004050-4	187	3.627	7	187	5.457	1	187	0.299
c1004050-5	226	8.719	10	226	153.886	5	226	0.975
c1008010-1	29	46.460	15	29	3.788	1	30	1.975
c1008010-2	41	83.842	13	41	179.845	2	44	5.133
c1008010-3	33	34.342	14	33	6.320	2	34	1.595
c1008010-4	37	32.109	13	37	37.972	3	37	1.931
c1008010-5	32	109.275	21	32	7.809	1	32	1.779
c1008025-1	87	44.891	13	87	14.512	1	89	1.214
c1008025-2	155	143.248	16	161	180.000	1	158	6.559
c1008025-3	128	41.352	13	129	180.000	1	132	2.125
c1008025-4	124	23.945	8	120	41.714	1	129	1.364
c1008025-5	144	65.219	14	143	180.000	1	144	3.418
c1008050-1	349	30.758	10	346	180.000	1	356	2.374
c1008050-2	436	73.208	15	450	180.000	1	444	3.285
c1008050-3	398	149.196	19	398	180.000	1	406	1.989
c1008050-4	424	27.578	14	431	180.000	1	440	2.527
c1008050-5	388	26.355	12	404	180.000	1	402	3.904

表 12: GAP に対する実験結果 (4)

問題例	CS			iDS			fix	
	obj	time	iter	obj	time	iter	obj	time
e0504010-1	224	10.440	10	224	1.376	1	229	0.609
e0504010-2	184	13.524	14	184	1.768	1	191	1.315
e0504010-3	133	35.078	15	133	0.975	1	133	0.470
e0504010-4	190	20.724	13	190	1.982	1	190	0.502
e0504010-5	221	50.615	18	221	35.183	4	237	0.455
e0504025-1	826	6.126	9	826	26.206	1	860	0.956
e0504025-2	706	17.311	13	706	19.172	1	707	0.534
e0504025-3	659	74.567	23	660	17.921	1	664	0.347
e0504025-4	639	23.550	13	639	15.187	1	691	0.518
e0504025-5	670	13.933	11	670	19.359	1	670	0.358
e0504050-1	2056	11.672	11	2056	40.714	1	2092	0.390
e0504050-2	2028	91.900	15	2028	116.630	1	2028	1.514
e0504050-3	1519	4.447	8	1519	17.730	1	1519	0.493
e0504050-4	1670	17.531	15	1689	78.230	1	1754	0.539
e0504050-5	1731	49.451	16	1731	24.167	1	1778	0.331
e0508010-1	337	21.994	12	337	101.733	1	331	1.311
e0508010-2	312	31.518	11	318	180.000	1	316	1.490
e0508010-3	342	154.684	20	341	179.533	2	347	2.432
e0508010-4	400	96.936	17	407	180.000	1	418	1.553
e0508010-5	264	23.131	10	264	19.682	1	277	2.035
e0508025-1	1245	179.508	18	1263	180.000	1	1248	0.728
e0508025-2	1260	179.460	15	1293	180.000	1	1272	2.941
e0508025-3	1282	30.388	11	1294	180.000	1	1282	0.799
e0508025-4	1537	179.913	14	1562	180.000	1	1555	2.299
e0508025-5	1122	31.910	11	1133	180.000	1	1167	2.686
e0508050-1	3991	65.154	13	4073	180.000	1	4026	1.571
e0508050-2	3591	39.640	15	3608	180.000	1	3608	4.048
e0508050-3	3677	52.390	15	3748	180.000	1	3696	0.609
e0508050-4	3711	136.467	17	3780	180.000	1	3822	1.354
e0508050-5	3448	145.327	14	3487	180.000	1	3548	2.047
e1004010-1	275	124.464	24	293	180.000	1	285	12.340
e1004010-2	244	139.004	22	266	179.497	2	251	12.436
e1004010-3	270	14.191	14	282	180.000	1	270	10.783
e1004010-4	251	140.589	25	255	134.077	1	255	14.193
e1004010-5	210	132.907	29	191	164.738	3	198	2.397
e1004025-1	784	49.376	23	808	180.000	1	778	5.422
e1004025-2	684	163.762	25	670	180.000	1	677	4.956
e1004025-3	722	30.259	16	722	180.000	1	689	12.993
e1004025-4	829	86.714	19	829	180.000	1	832	4.672
e1004025-5	640	49.827	18	686	180.000	1	654	1.995
e1004050-1	1647	87.402	30	1686	180.000	1	1706	1.820
e1004050-2	1678	107.027	19	1739	180.000	1	1734	13.263
e1004050-3	1667	61.012	22	1690	180.000	1	1737	10.801
e1004050-4	1988	55.122	17	2079	180.000	1	1988	4.041
e1004050-5	1456	43.974	16	1456	180.000	1	1580	1.998
e1008010-1	628	180.000	19	843	180.000	1	590	81.509
e1008010-2	589	180.000	18	689	180.000	1	579	25.948
e1008010-3	609	173.675	12	501	180.000	1	434	13.245
e1008010-4	631	20.681	14	738	180.000	1	609	24.932
e1008010-5	625	15.399	12	620	180.000	1	547	16.147
e1008025-1	2061	174.756	15	2120	180.000	1	1903	6.307
e1008025-2	1942	12.756	10	2124	180.000	1	1879	118.620
e1008025-3	1666	147.608	17	1757	180.000	1	1616	5.363
e1008025-4	1810	154.115	16	1934	180.000	1	1732	27.926
e1008025-5	1765	180.000	19	1969	180.000	1	1776	14.879
e1008050-1	4670	81.544	16	4777	180.000	1	4681	5.017
e1008050-2	4452	39.213	14	4614	180.000	1	4461	29.375
e1008050-3	4368	171.103	15	4489	180.000	1	4283	25.810
e1008050-4	4092	150.951	15	4108	180.000	1	3987	17.844
e1008050-5	4149	107.756	17	4223	180.000	1	4105	29.931

最大リグレット最小化0-1整数計画問題に対するコア選択法

長谷川 和樹・呉 偉（静岡大学）
柳浦 睦憲（名古屋大学）

- **最大リグレット最小化0-1整数計画問題**
  - 問題説明
  - 既存解法の紹介
- **最大リグレット最小化0-1整数計画問題に対するコア選択法**
  - コア選択法
  - 問題説明
  - 計算実験結果
- **まとめ**

- **最大リグレット最小化0-1整数計画問題**
  - 問題説明
  - 既存解法の紹介

- **最大リグレット最小化0-1整数計画問題に対するコア選択法**
  - コア選択法
  - 問題説明
  - 計算実験結果

- **まとめ**

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 問題説明

<u>0-1整数計画問題（binary integer programming problem, BIP）</u>：

- 数理計画問題のうち，決定変数が0か1しか取らないもの
- 多くの問題がBIPとして定式化可能

BIPとして定式化可能な問題の例：

- **最短路問題，最小全域木問題** ← **多項式時間で解決可能**
- **ナップサック問題，集合被覆問題，一般化割当問題** ← **NP困難**

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 問題説明

$$\min_{x \in X} \sum_{i \in N} c_i x_i$$

※$X \subseteq \{0,1\}^n$ は**線形制約**で構成された実行可能領域

※一般性を失わないので**目的関数は最小化**とする

- 古典的なBIPでは入力データは確定
- 現実世界の問題には不確定性が内在
- **最大リグレット最小化0-1整数計画問題（min-max regret BIP, MMR-BIP）**では目的関数に関わる入力 $\boldsymbol{c}$ に不確定さを考慮し，$c_i \in [c_i^-, c_i^+]$ とする（入力は $\boldsymbol{c}^-, \boldsymbol{c}^+$ ）

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 問題説明
- リグレット

- $\boldsymbol{c}$ の決定例をシナリオと呼ぶ

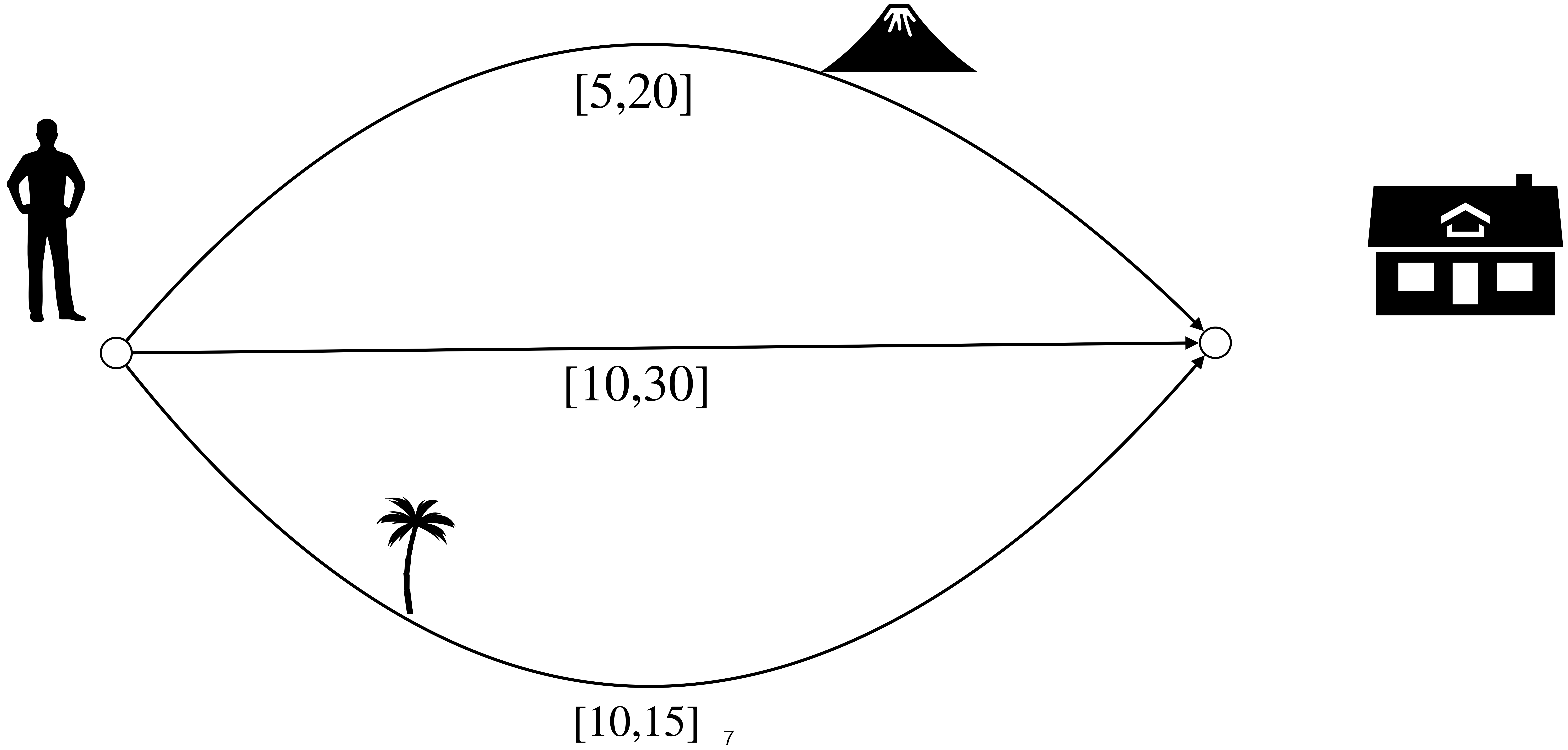
- ある解 $\boldsymbol{x}$ とシナリオ $\boldsymbol{c}$ について，**リグレット（後悔値）**$r(\boldsymbol{x}, \boldsymbol{c})$ とは，$\boldsymbol{c}$ の下での最適値と解 $\boldsymbol{x}$ の目的関数値の差

- 最小化問題の場合は以下のように表現できる:

$$\boxed{r(\boldsymbol{x}, \boldsymbol{c}) = \sum_{i \in N} c_i x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i y_i}$$

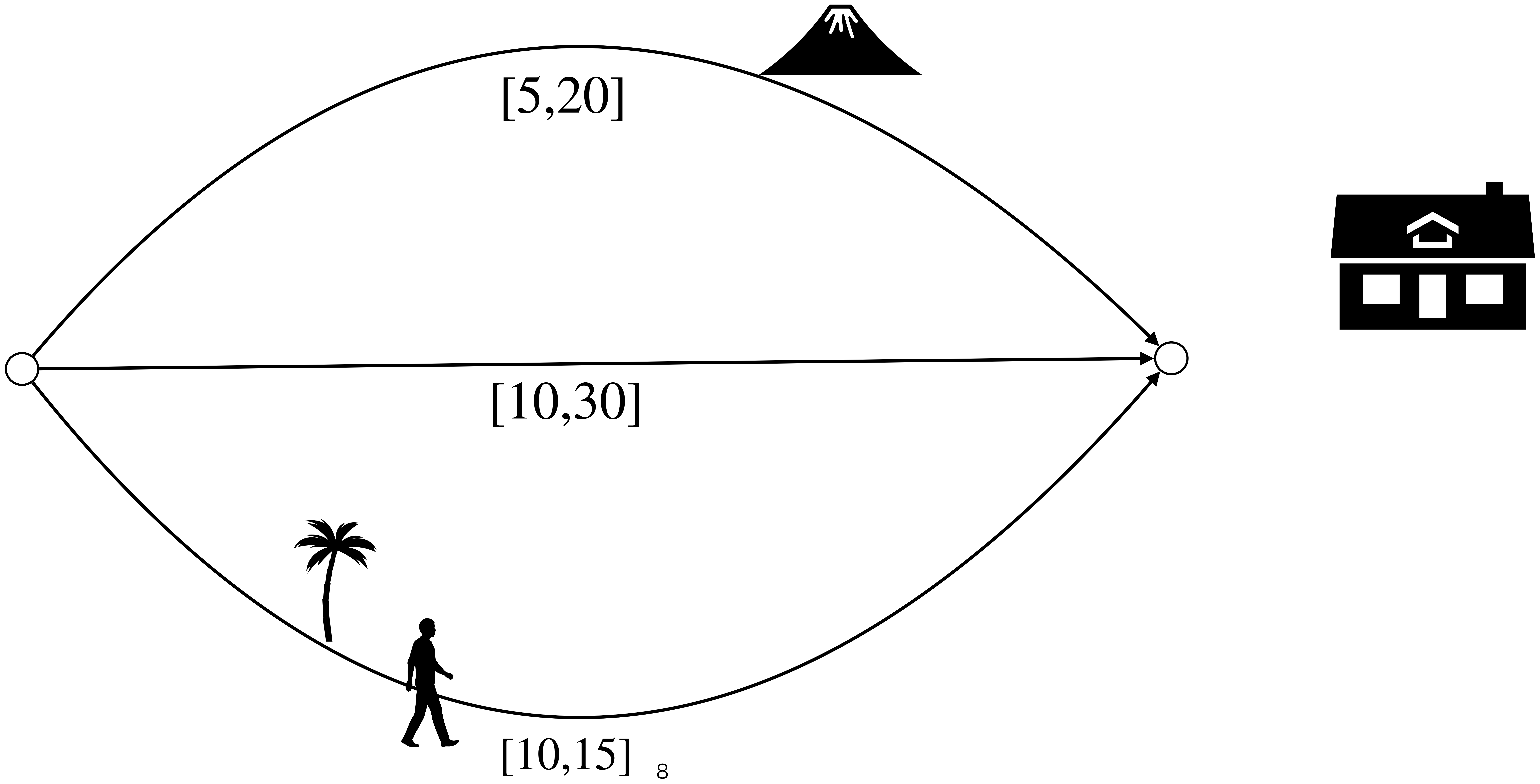
最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 問題説明
  - リグレット（イメージ）



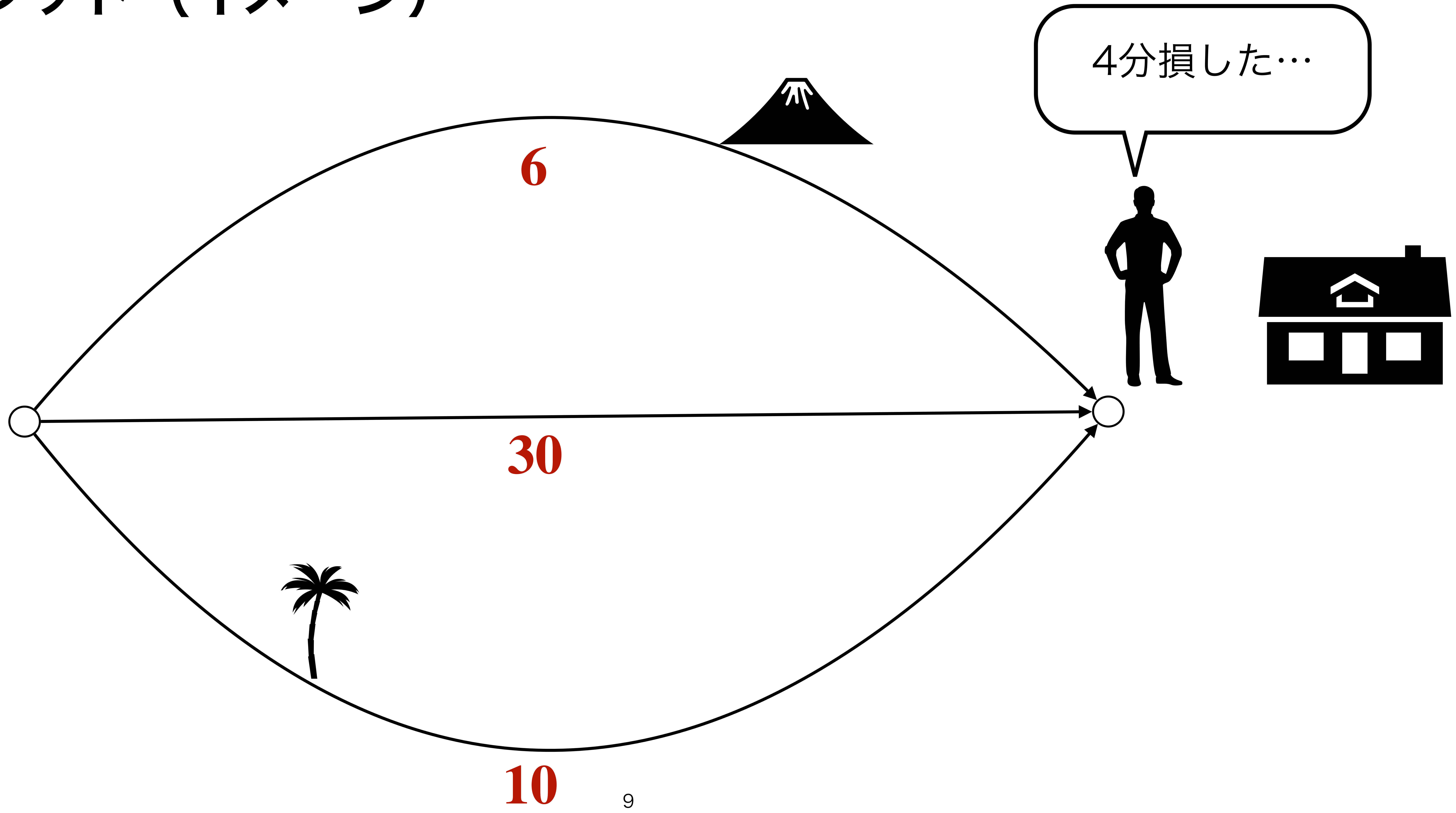
最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 問題説明
  - リグレット（イメージ）



最大リグレット最小化0-1整数計画問題 (MMR-BIP)

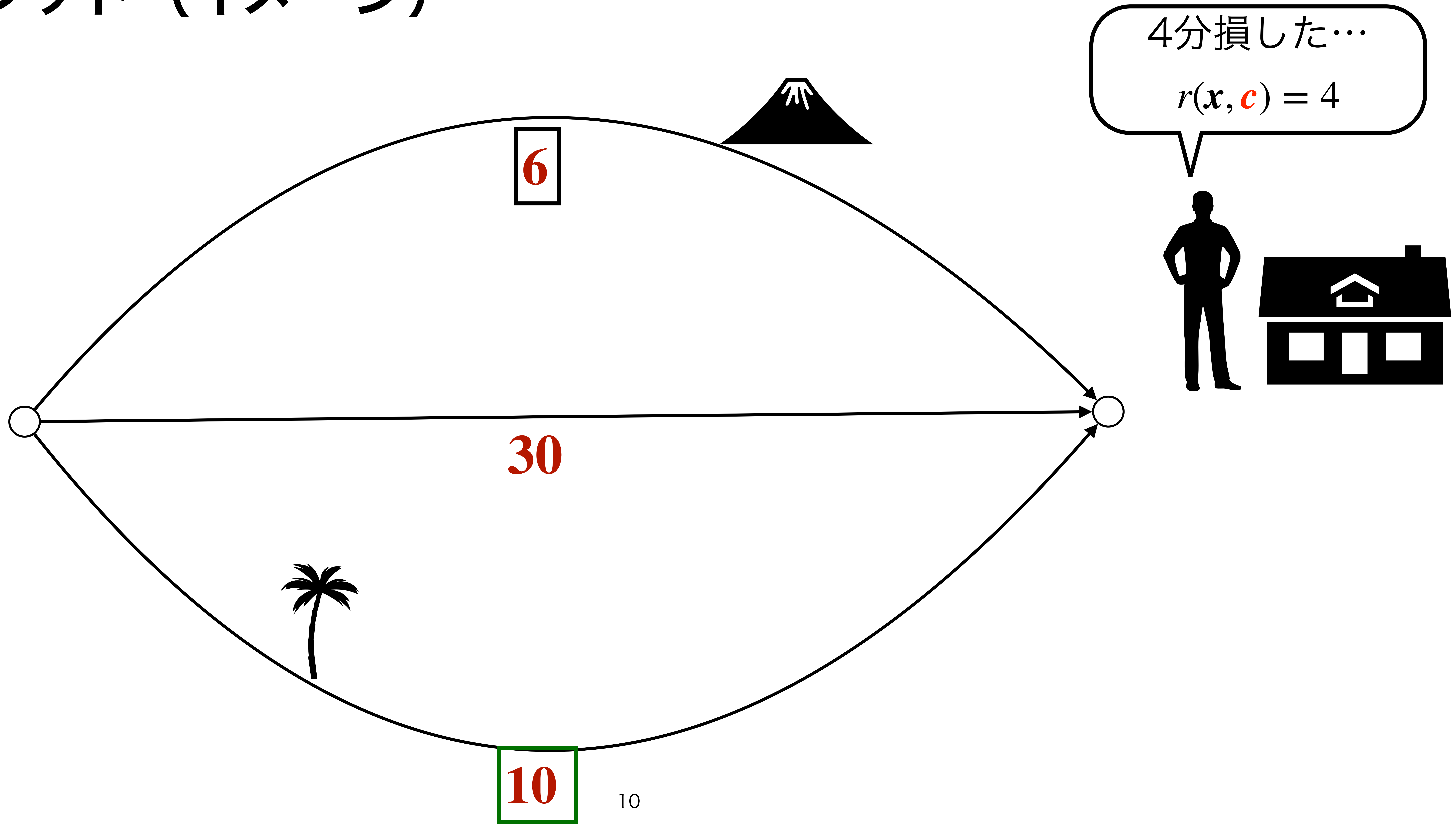
- 問題説明
  - リグレット（イメージ）



最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 問題説明

- リグレット（イメージ）



最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 問題説明
- 最大リグレット

- リグレットはシナリオに依存

- 最大リグレット $r_{\max}(\boldsymbol{x})$ ：各実行可能解 $\boldsymbol{x}$ に対して最大のリグレットと定義

$$\boxed{r_{\max}(\boldsymbol{x}) = \max_{\boldsymbol{c} \in U} \left(\sum_{i \in N} c_i x_i - \min_{y \in X} \sum_{i \in N} c_i y_i \right)}$$

※U は実行可能なシナリオの集合

- $r_{\max}(\boldsymbol{x})$ を最小化させるような $\boldsymbol{x}$ を求めることが目標

最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 問題説明

- 最大リグレット（最悪シナリオ）

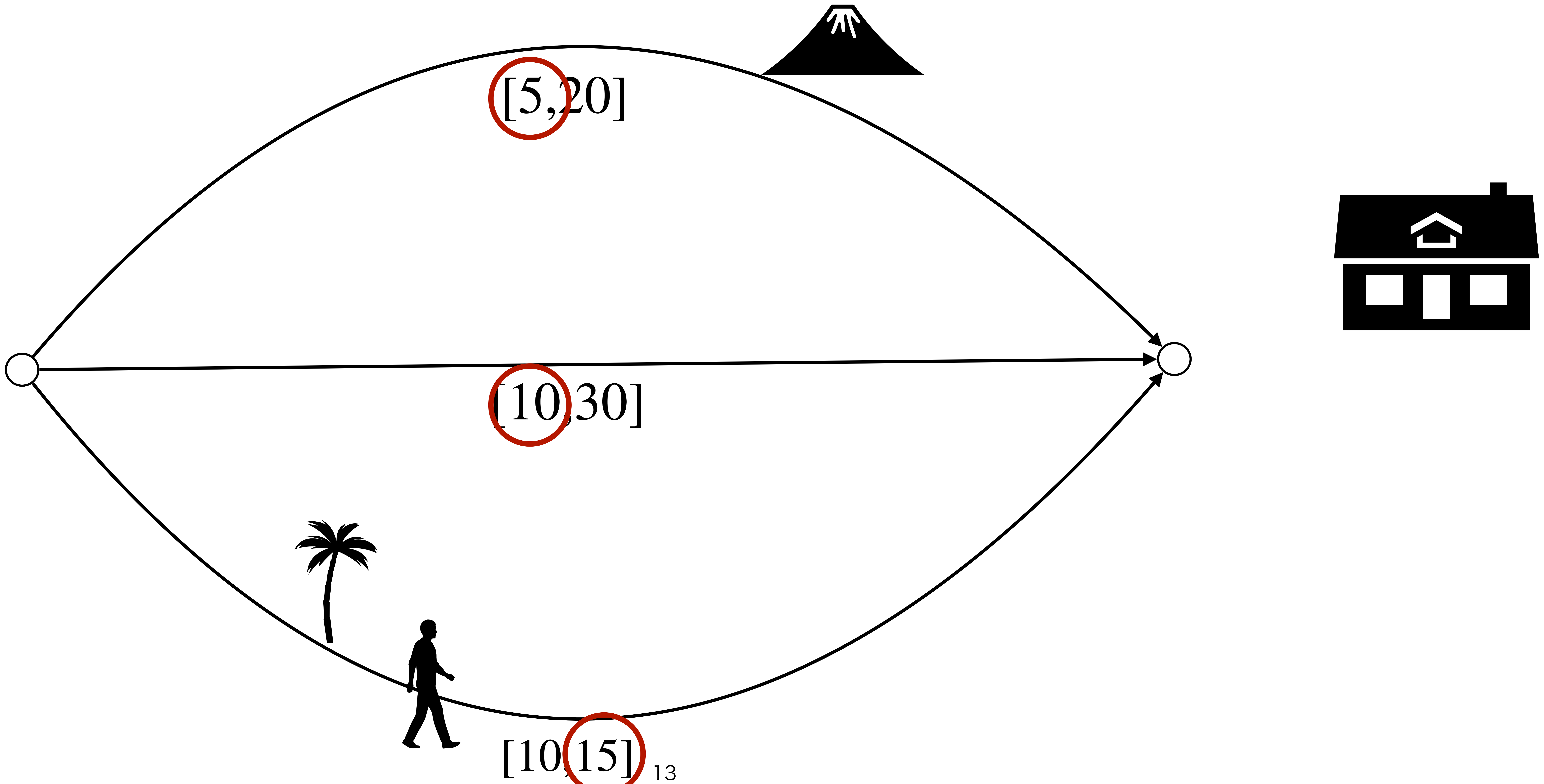
- 最大リグレットを与えるようなシナリオを**最悪シナリオ**と呼ぶ

- 最悪シナリオの一つ $\boldsymbol{c}^{\sigma(\boldsymbol{x})}$ は以下のように表現可能：

$$\boxed{c_i^{\sigma(\boldsymbol{x})} = \begin{cases} c_i^+ & (\text{if} \quad x_i = 1) \\ c_i^- & (\text{if} \quad x_i = 0) \end{cases}}$$

最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 問題説明
  - 最大リグレット（最悪シナリオ）



最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 問題説明
- 最大リグレット

- 最悪シナリオ $c_i^{\sigma(\boldsymbol{x})}$ を用いると最大リグレットは以下のように定義可能:

$$\boxed{r_{\max}(\boldsymbol{x}) = \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} y_i}$$

- つまりMMR-BIPは以下の問題である:

$$\boxed{\min_{\boldsymbol{x} \in X} \quad \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} y_i}$$

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 問題説明

$$\boxed{\min_{x \in X} \sum_{i \in N} c_i x_i}$$

※$X \subseteq \{0,1\}^n$ は**線形制約**で構成された実行可能領域

※一般性を失わないので**目的関数は最小化**とする

- MMR-BIPはBIPの一般化である
（$c_i^- = c_i^+ \quad \forall i \in N$ のとき，MMR-BIPとBIPは等価）

- MMR-BIPの決定版はΣ_2^{P}完全である

→$P \neq NP$**である限りMMR-BIPの解に対する多項式時間の厳密評価ができない**

→$P \neq NP$**である限りMMR-BIPに近似アルゴリズムは存在しない**

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 既存解法の紹介
- 発見的解法（シナリオ固定法）

- BIPの実行可能解 $\Longleftrightarrow$ MMR-BIPの実行可能解

- あるシナリオの下でBIPを解くことでMMR-BIPの実行可能解を得ることができる

- この手法を**シナリオ固定法**と呼ぶ

- 中間値シナリオ $\boldsymbol{c}^{\mathrm{med}}$（$c_i^{\mathrm{med}} = \dfrac{c_i^- + c_i^+}{2} \quad \forall i \in N$）を用いると，シナリオ固定法で得られる解の目的関数値は**最適値の2倍以内**

最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 既存解法の紹介

- 厳密解法（双対置換法）

$$\boxed{\min_{x \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} y_i}$$

- MMR-BIPが難しいのは**外の問題**と**中の問題**の二段階の最適化問題であるため

- **双対置換(dual substitution, DS)法**では**中の問題**をその線形緩和問題の双対問題で置換することでMMR-BIPを一つの大きな最小化問題として書き換えることができる

※ 線形緩和を挟むため，双対置換法に最適性の保証はない

最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 既存解法の紹介

- 厳密解法（反復双対置換法）

$$\min_{x \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} x_i - \min_{\boldsymbol{y} \in X} \sum_{i \in N} c_i^{\sigma(\boldsymbol{x})} y_i$$

- **反復双対置換(iterated dual substitution, iDS)法**ではDS法を反復的に利用する

- 反復方法を2種類紹介

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 既存解法の紹介
- 厳密解法（反復双対置換法・ハミング距離制約）

$$\sum_{\substack{i \in N: \\ \hat{x}_i = 0}} x_i + \sum_{\substack{i \in N: \\ \hat{x}_i = 1}} (1 - x_i) \geq r \qquad \forall \hat{\boldsymbol{x}} \in \hat{X}\,.$$

※ $\hat{X} \subseteq \{0,1\}^n$ は探索済み解集合

- **ハミング距離制約**は，既知の解$\hat{\boldsymbol{x}}$からハミング距離 r 離れることを意味する制約

- $r = 1$の場合は対象の解$\hat{\boldsymbol{x}}$のみを排除

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 既存解法の紹介
- 厳密解法（反復双対置換法・最良シナリオ制約）

- 解$\boldsymbol{x}$について，最小リグレットを与えるシナリオ$\phi(\boldsymbol{x})$を**最良シナリオ**と呼び，以下のように定義することができる:

$$c_i^{\phi(\boldsymbol{x})} = \begin{cases} c_i^- & (\text{if} \quad x_i = 1) \\ c_i^+ & (\text{if} \quad x_i = 0) \end{cases}$$

最大リグレット最小化0-1整数計画問題 (MMR-BIP)
- 既存解法の紹介
- 厳密解法（反復双対置換法・最良シナリオ制約）

$$c_i^{\phi(\boldsymbol{x})} = \begin{cases} c_i^- & (\text{if} \quad x_i = 1) \\ c_i^+ & (\text{if} \quad x_i = 0) \end{cases}$$

・解$\boldsymbol{x}$と解$\boldsymbol{y}$について，

$$\sum_{i \in N} c_i^{\phi(\boldsymbol{x})} x_i \geq \sum_{i \in N} c_i^{\phi(\boldsymbol{x})} y_i$$

であるとき，$r_{\max}(\boldsymbol{x}) \geq r_{\max}(\boldsymbol{y})$ が成り立つことが知られている

最大リグレット最小化0-1整数計画問題 (MMR-BIP)

- 既存解法の紹介

- 厳密解法（反復双対置換法・最良シナリオ制約）

$$c_i^{\phi(\boldsymbol{x})} = \begin{cases} c_i^- & (\text{if} \quad x_i = 1) \\ c_i^+ & (\text{if} \quad x_i = 0) \end{cases}$$

・この性質を用いると，**最良シナリオ制約**を以下のように設定できる:

$$\sum_{i \in N} c_i^{\phi(\boldsymbol{x})} x_i < \sum_{i \in N} c_i^{\phi(\boldsymbol{x})} \hat{x}_i \qquad \forall \hat{\boldsymbol{x}} \in \hat{X}$$

※ $\hat{X} \subseteq \{0,1\}^n$ は探索済み解集合

- **最大リグレット最小化0-1整数計画問題**
  - 問題説明
  - 既存解法の紹介
- **最大リグレット最小化0-1整数計画問題に対するコア選択法**
  - コア選択法
  - 問題説明
  - 計算実験結果
- **まとめ**

最大リグレット最小化0-1整数計画問題に対するコア選択法

- コア選択法

- MMR-BIPを解くことを考えると，計算時間は決定要素の個数$|N|$の増加に伴って爆発的に増加する傾向がある
- N の代わりに $C \subseteq N$ を考え，C のみを対象にMMR-BIPを解く手法を考える
- この手法を**コア選択法**と呼び，C を**コア**と呼ぶ

最大リグレット最小化0-1整数計画問題に対するコア選択法

- コア選択法

C の要素を増やす (速さ優先)	⇄	C 内の要素のみを対象に解く (解の質優先)

- 発見的解法を用いてある程度よい実行可能解を求める
- ↑で求めた解 $\boldsymbol{x}'$ について，$x'_i = 1$となるような i を C に追加する
- Cの要素がある程度増えた段階で厳密に解く

この反復を$C = N$となるか時間制限が来るまで行う

最大リグレット最小化0-1整数計画問題に対するコアの選択法

- コア選択法

C の要素を増やす
（速さ優先）

C 内の要素のみを対象に解く
（解の質優先）

発見的解法を用いてある程度よい実行可能解を求める

$=1$となるよ

階で厳密に解く

シナリオ固定法を利用

iDS法（最良シナリオ制約）を利用

この反復を$C=N$となるか時間制限が来るまで行う

最大リグレット最小化0-1整数計画問題に対するコア選択法

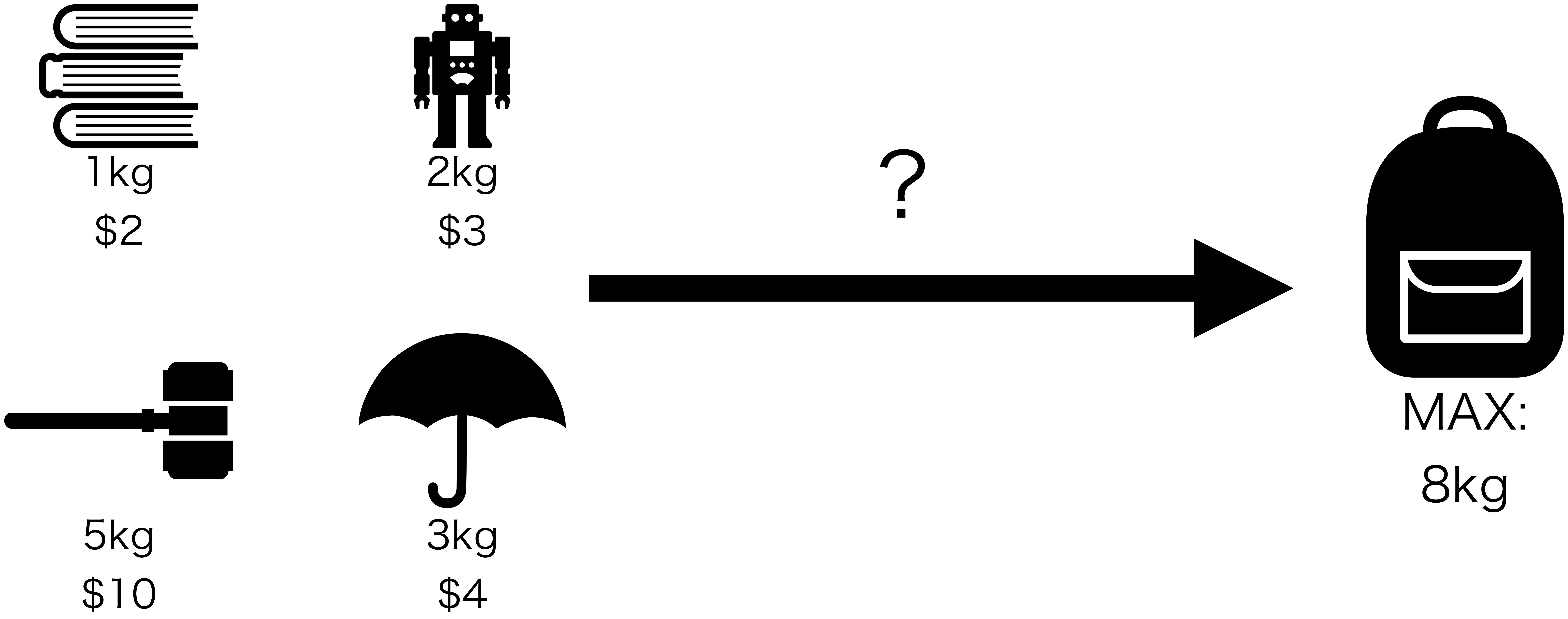
- コア選択法は**最大リグレット最小化巡回セールスマン問題**に非常に良い性能を示していた[2]
  →MMR-BIPにはどうだろうか？
- 先行研究[1]で公開されていた，**ナップサック問題**，**集合被覆問題**，**一般化割当問題**の最大リグレット最小化ベンチマーク問題例に対して計算実験を行ってみる
- ↑これらの問題には最良シナリオ制約を用いたiDS法が非常に良い性能を示していた

[1]Kazuki Hasegawa and Wei Wu. A heuristic approach for the robust traveling salesmanproblem. In 2022 IEEE International Conference on Industrial Engineering and En- gineering Management (IEEM), pp. 0561– 0565. IEEE, 2022.
[2]Wei Wu, Manuel Iori, Silvano Martello, and Mutsunori Yagiura. An iterated dual substitution approach for binary integer programming problems under the min-max regret criterion. INFORMS Journal on Computing, Vol. 34, No. 5, pp. 2523-2539, 2022.

最大リグレット最小化0-1整数計画問題に対するコア選択法
- 問題説明

・ **ナップサック問題**（knapsack problem, **KP**）：

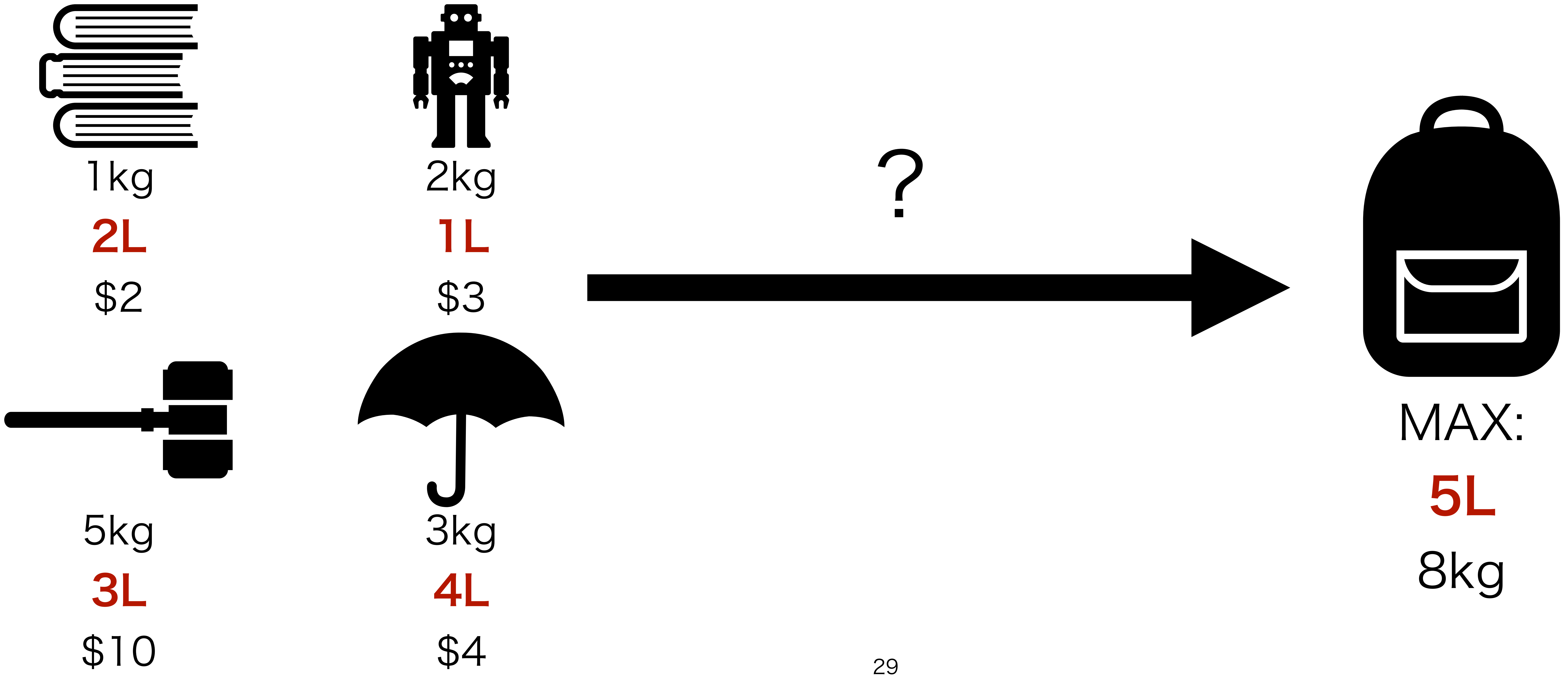
重量などの制限を守りつつ，価値を最大化できるようにアイテムをナップサックに詰めることを目指す問題



最大リグレット最小化0-1整数計画問題に対するコア選択法
- 問題説明

- **多次元ナップサック問題**（multidimensional knapsack problem, **MKP**）

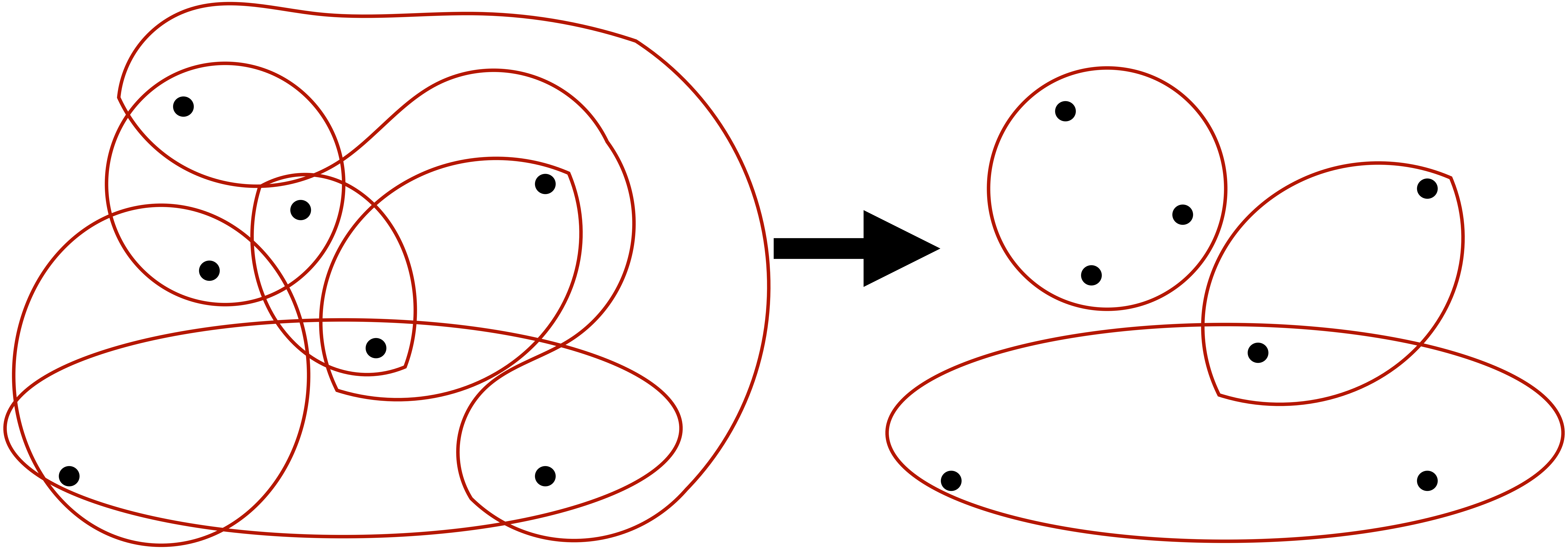
ナップサック問題の制約を多次元に拡張した問題



最大リグレット最小化0-1整数計画問題に対するコア選択法
- 問題説明

・ **<u>集合被覆問題</u>**（set covering problem, **SCP)**

ある集合の要素をすべて被覆するような集合族の集合のうち，最も総コストが小さなものを選択する問題



最大リグレット最小化0-1整数計画問題に対するコア選択法
- 問題説明

- **<u>一般化割当問題</u>**（generalized assignment problem, **GAP**）
  - n個の仕事をm台の機械に，割当コストが最小になるよう割り当てる問題
  - ある仕事をある機械に割当てると負荷がかかり，各機械には負荷の上限がある

最大リグレット最小化0-1整数計画問題に対するコア選択法
-計算実験結果

- 先行研究*にて公開されていた問題例計528題に対して計算実験を行った
  - ナップサック問題（KP）：108題
  - 多次元ナップサック問題（MKP）：90題
  - 集合被覆問題（SCP）：90題
  - 一般化割当問題（GAP）：240題
- 混合整数計画問題を解く際には**Nuorium Optimizer V26.1.1**を使用し，実装にはPython3.12.7を使用した

**Wei Wu, Manuel Iori, Silvano Martello, and Mutsunori Yagiura. An iterated dual substitution approach for binary integer programming problems under the min-max regret criterion. INFORMS Journal on Computing, Vol. 34, No. 5, pp. 2523-2539, 2022.*

最大リグレット最小化0-1整数計画問題に対するコア選択法
-計算実験結果

※コア選択法の 勝ち数-負け数

※180秒で辿り着いた最良解の質で評価

※括弧内は同じ目的関数値の場合に，その解に辿り着いた早さの勝敗

問題	問題例数	iDS法（最良シナリオ）		シナリオ固定法	
KP	108	6–0	(97–5)	51–0	(4–53)
MKP	90	9–31	(25–25)	35–28	(4–23)
SCP	90	8–18	(10–54)	86–0	(0–4)
GAP	240	52–8	(96–84)	127–17	(0–96)
合計	528	74–57	(229–168)	299–45	(8–176)

Wei Wu, Manuel Iori, Silvano Martello, and Mutsunori Yagiura. An iterated dual substitution approach for binary integer programming problems under the min-max regret criterion. INFORMS Journal on Computing, Vol. 34, No. 5, pp. 2523-2539, 2022.

最大リグレット最小化0-1整数計画問題に対するコア選択法
-計算実験結果

問題	問題例数	iDS法（最良シナリオ）		シナリオ固定法	
KP	108	6–0	(97–5)	51–0	(4–53)
MKP	90	9–31	(25–25)	35–28	(4–23)
SCP	90	8–18	(10–54)	86–0	(0–4)
GAP	240	52–8	(96–84)	127–17	(0–96)
合計	528	74–57	(229–168)	299–45	(8–176)

- シナリオ固定法には明確に勝っている
- iDS法には，KPとGAPには明確に優っており，MKPとSCPにはおおよそ敗れているが一部の問題例には優っている
- 特にGAPには**<u>大きく勝っている</u>**

Wei Wu, Manuel Iori, Silvano Martello, and Mutsunori Yagiura. An iterated dual substitution approach for binary integer programming problems under the min-max regret criterion. INFORMS Journal on Computing, Vol. 34, No. 5, pp. 2523-2539, 2022.

- **最大リグレット最小化0-1整数計画問題**
  - 問題説明
  - 既存解法の紹介

- **最大リグレット最小化0-1整数計画問題に対するコア選択法**
  - コア選択法
  - 問題説明
  - 計算実験結果

- **まとめ**

まとめ

- ロバスト最適化問題の一種である最大リグレット最小化0-1整数計画問題（MMR-BIP）に対していくつかの手法を確認した
- 最大リグレット最小化巡回セールスマン問題に対して良い性能を示した**コア選択法**をMMR-BIPに拡張し，性能を確認した結果，既存の手法と少なくとも互角程度の性能があり，一部の問題では**既存手法を超える性能**があった
- 今後もコア選択法の性能を向上する方法を考えたい

謝辞

提供いただきました **Nuorium Optimizer**によって，非常に有意義な研究を行うことができました．株式会社NTTデータ数理システム様に心より感謝申し上げます．