確率計画法におけるナップサック制約の数理最適化問題について

神奈川大学大学院　理学研究科　数学領域　堀口研究室
鈴木陸斗　源隆哉　王瀚東　堀口正之
Rikuto Suzuki, Ryuya Minamoto, Wang Handong, & Masayuki Horiguchi
Field of Mathematics, Graduate School of Science, Kanagawa University

1　はじめに

本稿では、確率計画法におけるナップサック制約の数理最適化問題について考察する。ここで扱う問題は、確率変数で表される制約条件を満たす確率を閾値とする最適化問題である。まず、線形計画問題の定式化について示し、次に、整数計画問題として表現できるナップサック問題とその解法アルゴリズムとして、動的計画法と分枝限定法によるアプローチについて述べる。そしてさらに、確率計画法の数理最適化モデルについて述べ、最後に確率計画法と優先順位付きナップサック多面体 (PCKP) の関係をまとめる。

2　用語の定義と記号の準備

本論文では、以下のように、用語と記号を用いる。

目的関数　最大化あるいは最小化したい関数のこと。（$\sum_{j=1}^{n} c_j x_j$）
制約条件　その解が満たす必要のある条件のこと。（$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, ..., m$）
非負制約　負の値を取らない条件のこと。（$x_j \geq 0, \quad j = 1, ..., n$）
標 準 形　基準となる形式。以下の３つの特徴を持つ。目的関数の値を最大化にする。すべての変数に非負制約が付く。非負制約を除く全ての制約条件で左辺の値が右辺の値以下となる。
線形計画問題　目的関数が線形関数で、すべての制約条件が線形の等式もしくは不等式で表わされた最適化問題。
$x_j \in \mathbb{Z}_+$　x_j は 0 以上の整数。
実行可能解　制約条件を満たす解のこと。
実行不能　制約条件を満たす解が存在しないこと。

3　整数計画問題

線形計画問題は以下のように目的関数の最大化を不等式線形制約と、非負制約の条件による標準形で表される。

$$
\begin{aligned}
&\text{最大化} \sum_{j=1}^{n} c_j x_j \\
&\text{条　件} \sum_{j=1}^{n} a_{ij} x_j \leq b_i \ (i = 1, ..., m), \quad x_j \geq 0 (j = 1, ..., n)
\end{aligned}
$$

通常の線形計画問題は全ての変数で連続的な実数値をとるが、全ての変数が離散的な整数値のみをとる線形計画問題を整数計画問題と呼ぶ。上記の $x_j \geq 0$ を $x_j \in \mathbb{Z}_+ = \{0, 1, 2, \dots\}$ に置き換えたものが整数計画問題である。

4　ナップサック問題

以下のような制約条件付きの数理最適化問題として定式化される数理モデルをナップサック問題という。

$$\text{最大化} \sum_{j=1}^{n} p_j x_j$$
$$\text{条　件} \sum_{j=1}^{n} w_j x_j \leq C, \quad x_j \in \{0,1\}(j=1,...,n)$$

ナップサック問題とは整数計画問題の中でも、変数 x_j が 0 または 1 の整数値しかとれない最適化問題の一種で、1 つの袋と n 個の荷物が与えられて、袋に詰め込める重さの合計の上限を C、各荷物 j の重さを w_j $(0 < w_j < C)$、価値を $p_j(> 0)$、荷物 j を袋に詰めるならば $x_j = 1$、そうでなければ $x_j = 0$ のとき、価値 (目的関数 $\sum_{j=1}^{n} p_j x_j$ の値) の合計を最大にする荷物の詰め合わせを求める問題である。荷物の個数が少ない場合は最適解を求めることが容易だが、荷物の個数が多いと最適解が見つかるまでの計算に時間がかかる。そこで効率的に最適解を求める方法が必要である。ナップサック問題には次節以降で述べる代表的な 2 つの効率的な方法が存在する。

5　動的計画法

動的計画法とは与えられた問題を小さな部分問題に分割して解いたあとに、得られた小さな部分問題の 最適解を利用して、より大きな部分問題の最適解を求める手続きを繰り返していく手法である。ナップサック問題は動的計画法により効率的に求めることができる。p_j と w_j はすべて整数値をとると仮定する。袋の容量を $u(\leq C)$、詰め込む荷物の候補を $1,...,k(\leq n)$ に制限した部分問題を考える。

$$\text{最大化} \sum_{j=1}^{k} p_j x_j$$
$$\text{条　件} \sum_{j=1}^{k} w_j x_j \leq u, \quad x_j \in \{0,1\}(j=1,...,k)$$

この部分問題の最適値を表す関数を $f(k,u)$ とする。
このとき $f(1,u)$ の値は

$$f(1,u) = \begin{cases} 0, & u < w_1 \\ p_1, & u \geq w_1 \end{cases}$$

$f(k,u)$ の値は以下の漸化式 (最適方程式) で求められる:

$$f(k,u) = \begin{cases} f(k-1,u), & u < w_k \\ \max\{f(k-1,u), f(k-1,u-w_k)+p_k\}, & u \geq w_k \end{cases}$$

$u < w_k$ のときは、荷物 k を選択できないため、荷物の候補を $1,...,k-1$ に制限した部分問題の最適値が採用される。一方で $u \geq w_k$ ときは荷物 k を選択しなかった場合の最適値 $f(k-1,u)$ と、荷物 k を選択したときのナップサックの容量が $u - w_k$ に変化した場合の最適値 $f(k-1,u-w_k)+p_k$ を比較して値の大きい方を採用することが最適な詰め合わせである。

初期値 $f(1,u)$ を用いることで漸化式より $f(2,u)$ の値がわかる。これを繰り返すことで最後に求めた $f(n,C)$ がナップサック問題の最適値になる。

ナップサック問題に対する動的計画法のアルゴリズム

1. 上記の式にしたがって $f(1,u)$ $(u = 0,\cdots,C)$ を計算する。$k = 1$ とする。
2. $k = n$ ならば終了。そうでなければ $k = k + 1$ とする。
3. 上記の式にしたがって $f(k,u)$ $(u = 0,\cdots,C)$ を計算し、2 に戻る。

6　分枝限定法

分枝限定法は、直接解くことが難しい問題をいくつかの小規模な子問題に分割する分枝操作と、最適解が得られる見込みのない子問題を見つける限定操作の 2 つの操作を繰り返し適用する手法である。暫定解から得られる最適値の下界と、整数計画問題の $x_j \in \mathbb{Z}_+$ を非負実数値 $x_j \geq 0$ に緩和した線形計画緩和問題を解いて得られる最適値の上界を利用して限定操作を実現する。整数計画問題を緩和した線形計画緩和問題の例を図 1 に示す。

線形計画緩和問題の最適解が整数計画問題の実行可能解でなければ整数値をとらない変数 $\bar{x}_j$ が存在する。このとき x_j の値を $x_j \leq \lfloor \bar{x}_j \rfloor$ と $x_j \geq \lceil \bar{x}_j \rceil$ にそれぞれ制限することで 2 つの子問題を生成できる。P を子問題、$\overline{P}$ を P の線形計画緩和問題、$\overline{P}$ の最適解を $\bar{x}$、その時点における整数計画問題の暫定解を x^* とする。このとき「線形計画緩和問題 $\overline{P}$ が実行不能ならば子問題 P も実行不能」、「線形計画緩和問題 $\overline{P}$ の最適値 $z(\bar{x})$ が暫定値 $z(x^*)$ に対して $z(\bar{x}) \leq z(x^*)$ を満たすならば子問題 P は暫定解 x^* より良い実行可能解を持たない」、「線形計画緩和問題 $\overline{P}$ の最適解 $\bar{x}$ が整数計画問題の実行可能解 (すなわち整数解) ならば $\bar{x}$ は子問題 P の最適解である」、のいずれかが分かれば子問題 P への分枝操作の適用を終了する。

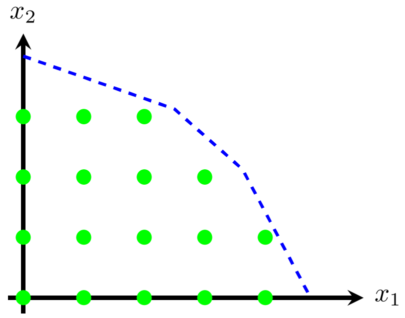

図 1　点の部分が整数計画問題の実行可能解、点線内が線形計画緩和問題の実行可能領域を表す

7　分枝限定法の適用

ナップサック問題に分枝限定法を適用するためには、$x_j \in \{0,1\}$ を $0 \leq x_j \leq 1$ に緩和した線形計画緩和問題を解いてナップサック問題の最適値の上界を求める必要がある。ナップサック問題に対する線形計画緩和問題について以下の解法が知られている。

定理 1（cf.［1］梅谷)

ナップサック問題は、$\frac{p_1}{w_1} \geq \frac{p_2}{w_2} \geq \cdots \geq \frac{p_n}{w_n}$ を満たすとする。このとき、線形計画緩和問題の最適解 $\bar{x} = (\bar{x}_1, \bar{x}_2, ...\bar{x}_n)^t$ は、$\sum_{j=1}^{k} w_j < C$ かつ $\sum_{j=1}^{k+1} w_j > C$ を満たす k を用いて、

$$\bar{x}_j = \begin{cases} 1 & j = 1, ..., k \\ \frac{C - \sum_{j=1}^{k} w_j}{w_{k+1}} & j = k+1 \\ 0 & j = k+2, ..., n \end{cases}$$

となる。

ナップサック問題に対する分枝限定法のアルゴリズム

1. 貪欲法などの方法で整数計画問題 P_0 の実行可能解を求めて、暫定解を x^* とする。$L = P_0$ とする。
2. 定理 1 を活用して P_0 の線形計画緩和問題 $\overline{P_0}$ を解き最適解 $\bar{x}$ が実行可能解 (整数解) ならば、$x^* = \bar{x}$(暫定解 x^* を $\bar{x}$ の値) に変更して終了する。最適解 $\bar{x}$ が実行可能解 (整数解) でなければ

P_0 に分枝操作を適用して生成した子問題を L に追加する。

3. $L = \emptyset$ となれば終了。
4. 子問題 $P \in L$ を 1 つ選ぶ。このとき $L = L \setminus \{P\}$ (L は子問題 P を除いた残りの L) とする。
5. 子問題 P の線形計画緩和問題 $\overline{P}$ を解き最適解 $\bar{x}$ を求める。実行不能もしくは $z(\bar{x}) \leq z(x^*)$(線形計画緩和問題 $\overline{P}$ の最適値 $z(\bar{x})$ が暫定値 $z(x^*)$ 以下) ならば 3 に戻る。
6. 最適解 $\bar{x}$ が整数計画問題 P_0 の実行可能解 (整数解) ならば $x^* = \bar{x}$(暫定解 x^*を$\bar{x}$ の値) に変更、実行可能解 (整数解) でなければ子問題 P に分枝操作を適用して生成した新たな子問題を L に追加して、3 に戻る。

8 確率計画問題

確率計画法 (stochastic programming) とは現実では目的関数および制約条件に不確実要素を伴う場合が多く。その不確実要素を直接モデルに組み入れた最適化手法のことを確率計画法という。確率計画法に関するアプローチは、**即時決定 (here and now)** と**待機決定 (wait and see)** に大別される。**即時決定 (here and now)** とは、確率変数の実現値を知る前に決定を下さなければならないアプローチである。**待機決定 (wait and see)** とは、確率変数の実現値を知ってから確定的な数理計画問題を取り扱うというアプローチである。

確率計画問題 (SP)

$$\begin{array}{lll} \text{最小化} & g_0(x, \tilde{\xi}) & \\ \text{条件} & g_j(x, \tilde{\xi}) \leq 0, & i = 1, ..., m \\ & x \in X \subset \mathbb{R}^n & \end{array}$$

このような問題を確率計画問題と呼ぶ (以後 SP と呼ぶ)。変数空間 X および関数 $g_j(x, \tilde{\xi}) : \mathbb{R}^n \to \mathbb{R}(i = 0, ..., m)$ は与えられているものとする。$\tilde{\xi}$ は N 次元確率変数ベクトルであり、そのとりうる値の集合を Ξ とする。確率空間 $(\Xi, \mathcal{F}, P)$ が与えられているものとする。決定変数 x が与えられたとき、関数 $g_j(x, \tilde{\xi}) : \Xi \to \mathbb{R}$ はこれ自身確率変数となり、確率 P は x に関して独立であると仮定する。上の問題は確率変数 Ξ の全てのとりうる値に対して、目的関数を最小化し、同時に制約条件を満たす解 x が必ず存在するとは言えない。つまり明確に定義されたものであるとは言えない。したがって、つぎのように 2 種類の**等価確定問題** (deterministic equivalent) に考え直す必要がある。

リコースを有する確率計画問題 (stochastic programming with recourse)

2 段階確率計画問題 (two-stage stochastic programming) とも呼ばれる。制約に確率変数が含まれるとき、制約を逸脱する度合いに応じて罰金を与え、**リコース**を含む目的関数を最小化する。ここでのリコースとは罰金に対する償還請求を表す。制約の満たされない場合の追加費用を重要視する。

確率的制約条件問題 (chance constrained program)

機会制約条件問題とも呼ばれる。制約条件の概念を拡張し、ある確率レベル（充足水準）で制約条件が満たされればよいとする**確率的制約条件**を用いる。制約が満たされる確率を問題にする。

次に、リコースを有する確率計画問題 (stochastic programming with recourse) を導出する。確率変数 $\tilde{\xi}$ の実現値を ξ とし、$\tilde{\xi}$ の実現値 ξ が知られた段階で (SP) の制約 $g_i(x, \xi) \leq 0,\ (i = 1, ..., m)$ を逸脱する度合いを $g_i{}^+(x, \xi)$ を次のように定義する。

$$g_i{}^+(x,\xi)=\begin{cases}0, & g_i(x,\xi)<0\text{ が成立する場合}\\ g_i(x,\xi), & \text{それ以外の場合}\end{cases}$$

このとき ξ が得られた段階での **2 段階変数**または、制約が侵されたことに対する**リコース変数** $y_i(\xi)(i=1,...,m)$ を各制約について考えることができる。$y_i(\xi)$ は制約が侵されたことに対する補償として、$g_i{}^+(x,\xi)-y_i(\xi)\leq 0$ となるように選ばれる。このように、新たな変数を加えて制約の充足を図ることは、余分な追加費用をもたらす。その制約の単位逸脱量あたりの罰金を q_i とすると、次の**リコース費用** $Q(x,\xi)$ が定義できる。この $Q(x,\xi)$ を求める問題をリコース問題あるいは 2 段階問題と呼ぶ。

$$Q(x,\xi)=\min{}_{y(\xi)}\left\{\sum_{i=1}^{m}q_iy_i(\xi)\middle|y_i(\xi)\geq g_i{}^+(x,\xi),i=1,...m\right\}$$

x を 1 段階変数と呼ぶ。(SP) および (リコース問題) の両者を考えると、総費用 $f_0(x,\xi)$ は ξ が与えられる前に決定 x を行った 1 段階費用 $g_0(x,\xi)$ とリコース費用 $Q(x,\xi)$ の総和に定義される。

$$f_0(x,\xi)=g_0(x,\xi)+Q(x,\xi)$$

次に線形リコース問題を考える。ここでは、$y(\xi)$ を $\bar{n}$ 次元の 2 段階変数ベクトルと定義し、$y(\xi)\in\mathbb{R}^{\bar{n}}$ は $Y\subset\mathbb{R}^{\bar{n}}$(ただし Y は$\{y|y\geq 0\}$のような多面体) に制限されるものとする。W を任意の $m\times\bar{n}$ 定数行列とし q を $\bar{n}$ 次元ベクトルと定義する。すると線形リコース問題を定義できる。

$$Q(x,\xi)=\min{}_{y(\xi)}\left\{q^Ty(\xi)\middle|Wy(\xi)\geq g^+(x,\xi),y(\xi)\in Y\right\}$$

この線形リコース問題で $\bar{n}=m$ のとき、$W=I$(単位行列) とすると、前述のリコース問題が得られる。次に線形リコース問題を一般化する。$\hat{q}(y(\xi)),H_i(y(\xi))$ を $\mathbb{R}^n\to\mathbb{R}$ の関数と定義する。すると一般化されたリコース問題を定義できる。

$$Q(x,\xi)=\min{}_{y(\xi)}\left\{\hat{q}(y(\xi))\middle|H_i(y(\xi))\geq g_i{}^+(x,\xi),y(\xi)\in Y,i=1,...,m\right\}$$

一般化されたリコース問題を導入し、また目的関数として $f_0(x,\tilde{\xi})$ の期待値を考えることにより、次のリコースを有する確率計画問題 (SPR) が (SP) の等価確定問題として定義できる。

$$\begin{array}{ll}\text{最小化} & E_{\tilde{\xi}}[f_0(x,\tilde{\xi})]\\ \text{条件} & f_0(x,\xi)=g_0(x,\xi)+Q(x,\xi),\qquad \xi\in\Xi\\ & Q(x,\xi)=\min{}_{y(\xi)}\{\hat{q}(y(\xi))|H(y(\xi))\geq g^+(x,\xi),y(\xi)\in Y\},\qquad \xi\in\Xi\end{array}$$

次のように、確率的制約条件問題 (chance constrained program) を定式化する。確率空間 $(\Omega,\mathcal{F},P)$ と可測空間 $x:\Omega\to\mathbb{R}^n$ の空間 Φ が与えられているものとする。次に、関数 $f:\Phi\to\mathbb{R}$、可測な制約関数 $g:\mathbb{R}^n\times\mathbb{R}^s\to\mathbb{R}^m$、確率 (変数) ベクトル $\xi:\Omega\to\mathbb{R}^s$、集合 $X\subset\Phi$ が与えられているとする。すると確率的制約条件問題は次のようになる。

- (1) 確率的制約条件問題 (プロトタイプ)

$$\begin{array}{ll}\text{最小化} & f(x)\\ \text{条件} & \mathbb{P}\{g(x(\omega),\xi(\omega))\geq 0\}\geq 1-\alpha\\ & x\in X\end{array}$$

記号 $\mathbb{P}$ は確率測度を表し、$\alpha\in(0,1)$ は規定の水準を表している。

ここで、確率 $p_1,...,p_N$ で発生する確率ベクトル ξ の実現値 $\xi^1,...,\xi^N$ が有限個しかない場合に焦点を当てる。これをシナリオと呼ぶ。この場合の確率的制約条件の定式化を容易にするために、指示関数 $\mathcal{X}:\mathbb{R}^m\to\{0,1\}$ を導入する。

$$\mathcal{X}(u)=\begin{cases}1, & u\geq 0\text{ の場合}\\ 0, & \text{その他の場合}\end{cases}$$

よって (1) 確率的制約条件問題 (プロトタイプ) はより具体的な形で表すことができる。

- (2) 確率的制約条件問題

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{条件} & \displaystyle\sum_{i=1}^{N} p_i \mathcal{X}(g(x^i, \xi^i)) \geq 1-\alpha \\ & x \in X \end{array}$$

多くの場合、シナリオの集合に半順序 $\preceq$ を定義できる。次に半順序について定義する。

半順序 (partial order) $\preceq$

$$i \preceq j \iff \xi^i \leq \xi^j$$

これは全部のシナリオの中で、限られた数のシナリオだけがこの問題に影響を与えることを示している。$\mathbb{P}\{\xi \leq \xi^i\} \geq 1-\alpha$ は、実現値の集合 ξ^i の最小点 (半順序 $\preceq$ の意味で) として定義される $(1-\alpha)$ 有効点である。

定義 1(一致性)

(i) $f(\bar{x}) \leq f(x)$

(ii) $\displaystyle\sum_{i=1}^{N} p_i \mathcal{X}(g(\bar{x}^i, \xi^i)) \geq \sum_{i=1}^{N} p_i \mathcal{X}(g(x^i, \xi^i))$

(iii) 全ての $i, j \in 1, ..., N$ に対して、 $(i \prec j) \wedge (g(\bar{x}^j, \xi^j) \geq 0) \implies (g(\bar{x}^i, \xi^i) \geq 0)$

となるような任意の $x \in X$ に対して、$\bar{x} \in X$ が存在する場合、$\{1, ..., N\}$ 上の半順序 $\preceq$ は (2) 確率的制約条件問題に一致であると言う。さらに $\bar{x} = x$ を満たすとき、強一致であると言う。

ちなみに $i \prec j$ は $(i \preceq j) \wedge (i \neq j)$ である。

プロトタイプ (1) に対して、次のような確率的制約条件を含む線形計画問題を考える。

$$\begin{array}{ll} \text{最小化} & C^T x \\ \text{条件} & \displaystyle\sum_{i=1}^{N} p_i \mathcal{X}(T^i x - h^i) \geq 1-\alpha \\ & x \in X \end{array}$$

シナリオ $i = 1, ..., N$ は $M \times N$ 確率行列 T と確率ベクトル $h \in \mathbb{R}^m$ の実現値 (T^i, h^i) によって特徴づけられる。コストベクトル $C \in \mathbb{R}^n$ および規定の水準 $\alpha \in (0, 1)$ は与えられているものとする。

次に (2) 確率的制約条件問題を混合整数計画問題として再定式化する。そのために、任意の $x \in X$ に対して、$g(x^i, \xi^i) + d^i \geq 0$ となるような各 $i = 1, ..., N$ ベクトル $d^i \in \mathbb{R}^m$ を探す。$g(\cdot, \xi^i)$ は連続でコンパクトであるため、このようなベクトルが存在する。これにより、(2) を混合整数計画問題に再定式化できる。

- (3)~(7) 混合整数計画の場合

$$\begin{array}{lll} \text{最小化} & f(x) & (3) \\ \text{条件} & g(x^i, \xi^i) + d^i z_i \geq 0, \qquad i = 1, ..., N & (4) \\ & \displaystyle\sum_{i=1}^{N} p_i z_i \leq \alpha & (5) \\ & x \in X & (6) \\ & z_i \in \{0, 1\}, \qquad i = 1, ..., N & (7) \end{array}$$

この問題を解こうとすると、シナリオの数 N が非常に大きい場合に困難であると思われる。そのため、その複雑さを軽減するために (2) に関する半順序を利用する。

次に、定義 1 から次の考察が得られる。

補題 3

半順序 $\preceq$ が (3)~(7) に対して、矛盾がない場合、任意の $i, j \in \{i, ..., N\}$ に対して、$(i \preceq j) \implies (z_i \leq z_j)$ となるような (3)~(7) の最適解 $(\hat{x}, \hat{z})$ が存在する。

したがって、(3)~(7) に $i \preceq j$ となるような任意の $i, j \in \{1, ..., N\}$ に対して、$z_i \leq z_j$(8 とする) という制約条件を加えることは、全ての最適解を取り除くわけではない。不等式 (5) と (8) は整数制約 (7) とともに、優先順位制約付きナップサック多面体 (precedence constrained knapsack polyhedron, PCKP) と定義される。PCKP の導入にあるアイディアを適用し求解の特徴的な方法を構築する。

次のように、集合 A_i を定義する。

$$A_i = \{j \in \{1, ..., N\} : i \preceq j\}, (i = 1, ..., N)$$

ナップサック制約の代わりに誘導被覆 (induced cover) の概念を用いる。

定義 2(誘導被覆 induced cover)

集合 $C \subseteq \{1, ..., N\}$ は$\mathbb{P}\left\{\bigcup_{i \in C} A_i\right\} > \alpha \quad (9)$ が成り立つとき誘導被覆と呼ばれる。

任意の誘導被覆 C に対して、有効な不等式

$$\sum_{i \in C} z_i \leq |C| - 1 \qquad (10)$$

が成り立つ。実際に、任意の $i \in C$ に対して、$z_i = 1$ ならば、(8) と A_i の定義は $k \in \bigcup_{i \in C} A_i$ に対して、$z_k = 1$ であることを意味する。したがって、(9) と (5) は矛盾する。

誘導被覆について例を使って説明する。図 2 に示されている確率変数 $\xi = (\xi_1, \xi_2)$ の実現値 $\xi^1, ..., \xi^{20}$ を考えてみる (簡単のため 8 つだけ番号を与える)。半順序 $\preceq$ は次のように定義されていると仮定する。$i \preceq j \iff \xi^i \leq \xi^j$ また、任意の実現値が任意の i に対して、等確率で $p_i = 0.05, \alpha = 0.25$ であるとする。このとき、A_i の定義によって, $A_1 = \{1, 4, 5, 6\}$, $A_2 = \{2, 5, 6, 7\}$, $A_3 = \{3, 6, 7, 8\}$ などの集合が得られる。例えば、集合 $C = \{1, 2\}$ は $\mathbb{P}\{A_1 \cup A_2\} = 0.05 \times 6 = 0.3 > 0.25 = \alpha$ なので誘導被覆である。誘導被覆は他にもたくさんあることが分かる。

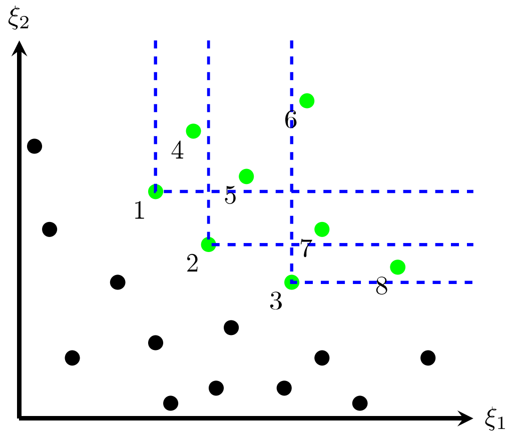

図 2　誘導被覆の例題（cf.　[7] Andrzej Ruszczynski)

しかし、シナリオの数が大きくなると、この方法では難しくなる。全ての適切な誘導被覆を列挙することは事実上不可能である。そこで 2 つの問題に焦点を当てる。与えられた分数の解を切断する被覆の不等式を見つけることと、それらのリフティングを決定することをそれぞれ考察する。

1 つ目の問題について考える。集合 $I \subset 1, ..., N$ と分数の点 $\tilde{Z}_i \in [0, 1]^N$ が与えられたときに、(10) が $\tilde{Z}_i$ を切断する誘導被覆 $C \subseteq I$ を見つける。つまり、

$$\textstyle\sum_{i \in C} \tilde{Z}_i > |C| - 1 \qquad (11)$$

(11) の両辺の差が最大となる被覆 C を見つけるために、シナリオ i が C に含まれるかどうかを決

定するための 2 値変数 (0 または 1)$v_i, i \in I$ を導入し、次のように最適化問題を定式化する。

$$\begin{array}{lll} \text{最小化} & \sum_{i \in I}(1-\tilde{Z}_i)v_i & (12) \\ \text{条件} & \mathbb{P}\left\{\bigcup_{i:v_i=1} A_i\right\} > \alpha & (13) \\ & v_i \in \{0,1\}, \quad i \in I & (14) \end{array}$$

定義 2 から次のことが得られる。

補題 4

I は誘導被覆だと仮定する。(12)~(14) の最適値が 1 よりも小さい場合､集合 $C = \{i \in I : v_i = 1\}$ は (11) を満たす誘導被覆を定義する。逆に、最適値が 1 以上の場合、(11) が成り立つような誘導被覆 $C \subseteq I$ は存在しない。

問題 (12)~(14) は依然として難しい最適解問題である。そこでブール・ボンフェローニの不等式を用いる。

$$\mathbb{P}\left\{\bigcup_{i\in C} A_i\right\} \geq \sum_{i\in C}\mathbb{P}\{A_i\} - \sum_{i,j\in C,i<j}\mathbb{P}\{A_i \cap A_j\} \qquad (15)$$

最初の近似ではこの不等式を制約条件 (13) に直接適用する。図 2 の例ではブール・ボンフェローニの不等式より、$\mathbb{P}\{A_1 \cup A_2 \cup A_3\} = \{1,2,3,4,5,6,7,8\} = 0.05 \times 8 = 0.4$, $\mathbb{P}\{A_1\} + \mathbb{P}\{A_2\} + \mathbb{P}\{A_3\} - (\mathbb{P}\{A_1 \cap A_2\} + \mathbb{P}\{A_1 \cap A_3\} + \mathbb{P}\{A_2 \cap A_3\}) = 0.6 - 0.25 = 0.35$ となり、$\mathbb{P}\{A_1 \cup A_2 \cup A_3\} \geq 0.35$ が得られるため、$\alpha = 0.25$ の場合にこれを適用することで、$\{1,2,3\}$ を被覆とすることができる。しかし $\alpha = 0.35$ の場合、$\mathbb{P}\{A_1 \cup A_2 \cup A_3\} = 0.4$ であっても見逃してしまう。

ブール・ボンフェローニの不等式を用いた混合整数計画問題で分離問題を近似するために、シナリオ i と j が両方とも被覆内にあるという決定を表す、追加の決定変数 $y_{ij}, i, j \in I, i < j$ を導入する。すると次が得られる。

$$\begin{array}{lll} \text{最小化} & \sum_{i \in I}(1-\tilde{Z}_i)v_i & (16) \\ \text{条件} & \sum_{i \in I} v_i\mathbb{P}\{A_i\} - \sum_{i,j \in I, i<j} y_{ij}\mathbb{P}\{A_i \cap A_j\} \geq \alpha + \epsilon & (17) \\ & y_{ij} \geq v_i + v_j - 1, \;\; y_{ij} \geq 0, \;\; i,j \in I, \;\; i < j & (18) \\ & v_i \in \{0,1\}, \quad i \in I & (19) \end{array}$$

ただし $0 < \epsilon < min_{1\leq i\leq N}P_i$ であり、ϵ の役割は強い不等式 (13) を弱い不等式 (17) に置き換えることを可能にすることである。

命題 1

(16)~(19) に解が存在する場合、集合 $C = i \in I : v_i = 1$ は誘導被覆である。さらに最適値が 1 よりも小さい場合、(11) は成り立つ。

ブール・ボンフェローニの不等式は明確でないが、問題 (16)~(19) は集合 A_i をクラスタリングすることで改善できる。

定義 3(適切な分割 proper partition)

以下の 3 つが成り立つとき、集合 $J_k \subseteq I, \quad k \in K$ は I の適切な分割と呼ばれる。

(i) $\bigcup_{k\in K} J_k = I$

(ii)$B_k = \bigcap_{i\in J_k} A_i \neq \emptyset, \qquad k \in K$

(iii) 任意の $k \in K, \quad i \in J_l, \quad l \neq k$ に対して、$B_k \cap A_i = \emptyset$

ブール・ボンフェローニの不等式による境界を明確にするために、適切な分割を用いる。任意の $i \in I$ に対して、$i \in J_{k(i)}$ となるような $k(i)$ が存在するものとする。

補題 5

$J_k,\quad k \in K$ は I の適切な分割である場合、

$$\mathbb{P}\left\{\bigcup_{i\in C} A_i\right\} \geq \sum_{k\in K}\mathbb{P}\{B_k\} + \sum_{i\in I}(\mathbb{P}\{A_i\} - \mathbb{P}\{B_{k(i)}\}) - \sum_{k\in K}\sum_{i,j\in J_k, i<j}(\mathbb{P}\{A_i\cap A_j\} - \mathbb{P}\{B_k\}) - \sum_{i,j\in I, i<j, k(i)\neq k(j)}\mathbb{P}\{A_i\cap A_j\} \tag{20}$$

が成り立つ。

補題 5 を用いて (16)~(19) を整理する。簡単のため、$\mu_i = \mathbb{P}\{A_i\}, \mu_{ij} = \mathbb{P}\{A_i \cap A_j\}, \rho_k = \mathbb{P}\{B_k\}$ とし、線形計画問題を考える。

$$\text{最小化} \quad \sum_{i\in I}(1-\tilde{Z}_i)v_i \tag{21}$$

$$\text{条件} \quad \sum_{k\in K}\rho_k\lambda_k + \sum_{i\in I}v_i(\mu_i - \rho_{k(i)}) - \sum_{k\in K}\sum_{i,j\in J_k, i<j}y_{ij}(\mu_{ij}-\rho_k) - \sum_{i,j\in I, i<j, k(i)\neq k(j)}y_{ij}\mu_{ij} \geq \alpha + \epsilon \tag{22}$$

$$y_{ij} \geq v_i + v_j - 1,\quad y_{ij}\geq 0,\quad i,j\in I,\quad i<j \tag{23}$$

$$\lambda_k \leq \sum_{i\in J_k} v_i,\quad \lambda_k \leq 1,\quad k\in K,\qquad v_i\in\{0,1\},\qquad i\in I \tag{24}$$

命題 2

(21)~(24) に解が存在する場合、集合 $C = i \in I : v_i = 1$ は誘導被覆である。さらに最適値が 1 よりも小さい場合、(11) は成り立つ。

次に、被覆のリフティングの問題について考える。誘導 $\beta-$ 被覆、つまり次のような集合 C を仮定する。

$$\textstyle\sum_{i\in C} Z_i \leq \beta \tag{25}$$

シナリオ $s \notin C$ では、以下の不等式を満たすような (γ_s, β_s) を発見したい。

$$\textstyle\sum_{i\in C} Z_i + \gamma_s\beta_s \leq \beta_s \tag{26}$$

これは PCKP に対して有効である。

図 2 に示す例では、全ての実現値の起こる可能性が等しく、$\alpha = 0.25$ であり、集合{1,2}は誘導被覆である。これは $\mathbb{P}A_1 \cup A_2 = 0.3$ だから。したがって、有効な不等式 $Z_1 + Z_2 \leq 1$ が得られる。ただし $\{1,3\}$ と $\{2,3\}$ も誘導被覆であるため、この不等式を $Z_1 + Z_2 + Z_3 \leq 1$ にリフティングすることができる。このアイディアを用いてリフティングのアルゴリズムを見つけ出したい。

次の場合について考えてみる。

$$s \notin \bigcup_{i\in C} A_i$$

$\beta_s = \beta$ と仮定し、(26) で $\gamma_s = 1$ とできるかどうかを調べながら、$\beta-$ 被覆の形でリフティングを探す。それは次の組合せ問題を解くことで決定できる。

$$\text{最大化} \quad \sum_{i\in C} v_i \tag{27}$$

$$\text{条件} \quad \mathbb{P}\{A_s \cup \bigcup_{i:v_i=1} A_i\} \leq \alpha \tag{28}$$

$$v_i \in \{0,1\},\qquad i\in C \tag{29}$$

この問題が β より小さい場合は、$\gamma_s = 1$ にできる。それ以外の場合は、$\gamma_s = 0$(リフティング失敗) になる。その後、次の候補変数を調べることができる。

問題 (27)~(29) は難しい組合せ最適化問題である。今の設定ではとても多くのシナリオを考慮すると、単純な形で解くことは非常に困難に思われる。そのために、問題 (27)~(29) の緩和法を開発する。まず (29) の左辺を計算しやすい下界に置き換える。1 つの方法は前の節と同様にブール・ボンフェローニの不等式 (15) を利用するとである。

ランダムな事象 $A_i, i \in I$ について p_m を $n = |I|$ 回の事象のうち確実に m 回発生する確率と定義する。確率 $p_m, m = 1, ..., n$ は二項モーメント方程式を満たす。

$$\sum_{m=r}^{n} \binom{m}{r} p_m = \sum_{i_1<i_2<...<i_r} \mathbb{P}\{A_{i_1} \cap A_{i_2} \cap ... \cap A_{i_r}\}, \quad r = 1, ..., n \tag{30}$$

これらの事象のうち少なくとも 1 回起こる確率は

$$\mathbb{P}\left\{\bigcup_{i \in I} A_i\right\} = \sum_{m=1}^{n} p_m \tag{31}$$

これらの関係を用いて次の線形計画問題を定式化する。

$$
\begin{aligned}
\text{最大化} \quad & \sum_{i \in I} v_i & (32)\\
\text{条件} \quad & \sum_{m=1}^{n} p_m \le \alpha & (33)\\
& \sum_{m=1}^{n} m p_m = \sum_{i \in I} v_i \mathbb{P}\{A_i\} & (34)\\
& \sum_{m=2}^{n} \binom{m}{2} p_m = \sum_{i<j} y_{ij} \mathbb{P}\{A_i \cap A_j\} & (35)\\
& y_{ij} \ge v_i + v_j - 1, \quad 0 \le y_{ij} \le min(v_i, v_j), \quad i, j \in I, \quad i < j & (36)\\
& v_i \in \{0, 1\}, \qquad i \in I & (37)\\
& p_m \ge 0, \quad m = 1, ..., n & (38)
\end{aligned}
$$

命題 3

$\bar{\beta}$ を問題 (32)~(38) の最適値とする。そのとき、$\sum_{i \in I} Z_i \le \bar{\beta}$ は有効な不等式である。

(25) の被覆 C をリフティングするには、上の結果を $I = C \cup \{s\}$ に適用し、$v_s = 1$ とする。最適値 $\bar{\beta}$ が β を超えない場合は、不等式に Z_s を追加できる。つまり、(25) で C を $C \cup \{s\}$ に置き換える。

制約 (33) は (28) の緩和であり、この緩和の質を向上させるために、より高次のモーメント制約を含めることができるが、確率計画法の内容では、考慮すべき事象 A_i の組み合わせの数が多いためとても現実的ではない。その代わりにすぐに得られる情報を駆使して、問題 (32)~(38) を改良してみる。最初に、各 A_i についての確率 $\delta_i = \mathbb{P}\left\{A_i \setminus \cup_{j \in I \setminus \{i\}} A_j\right\}$ を計算するのは簡単である。そうすると $p_1 \ge \sum_{i \in I} \delta_i v_i$ でなければばならない。つまり C は I の部分集合なので、不等式が必要になる。

次にクラスタリングを採用するとで、大幅な改善が図れる。ここで、$J_k, \ k \in K$ を適切な分割とする。前と同様に、$\mu_i = \mathbb{P}\{A_i\}, \mu_{ij} = \mathbb{P}\{A_i \cap A_j\}, \rho_k = \mathbb{P}\{B_k\}$ と表記すると

$$
\begin{aligned}
\text{最大化} \quad & \sum_{i \in I} v_i & (39)\\
\text{条件} \quad & \sum_{m=1}^{n} p_m + \sum_{k \in K} \rho_k \lambda_k \le \alpha & (40)\\
& \sum_{m=1}^{n} m p_m = \sum_{i \in I} (\mu_i - \rho_{k(i)}) v_i & (41)\\
& \sum_{m=2}^{n} \binom{m}{2} p_m = \sum_{k \in K} \sum_{i,j \in J_k, i<j} y_{ij}(\mu_{ij} - \rho_k) + \sum_{i,j \in I, i<j, k(i) \ne k(j)} y_{ij} \mu_{ij} & (42)
\end{aligned}
$$

$$y_{ij} \geq v_i + v_j - 1, \quad 0 \leq y_{ij} \leq min(v_i, v_j), \quad i,j \in I, \quad i < j \tag{43}$$
$$\sum_{j \in J_k} v_i \leq |J_k|\lambda_k, \quad k \in K \tag{44}$$
$$v_i \in \{0,1\}, \qquad i \in I \tag{45}$$
$$\lambda_k \in \{0,1\}, \qquad k \in K \tag{46}$$
$$p_1 \geq \sum_{i \in I} \delta_i v_i \tag{47}$$
$$p_m \geq 0, \quad m = 2, ..., n \tag{48}$$

不等式 (44) の役割はクラスター J_k 内のいくつかの集合が選択されるたびに、$\lambda_k = 1$ を強制する。だから ρ_k を (40) の左辺に追加する必要がある。命題 3 と同様に、命題 2 から次の結果が得られる。

命題 4

$\bar{\beta}$ を問題 (39)〜(48) の最適値とする。そのとき、$\sum_{i \in I} Z_i \leq \bar{\beta}$ は有効な不等式である。

次に、シナリオ

$$s \in \bigcup_{i \in C} A_i \tag{49}$$

に関するリフティングについて考えていく。C が最小の誘導被覆である場合は、十分に研究されており、次の結果より定式化することができる。

補題 6

C を誘導被覆とし、$J_k, \ k \in K$ を C の適切な分割とし、$j_k \in \bigcap_{i \in J_k} A_i$ とする。すると不等式

$$\sum_{i \in C} Z_i + \sum_{k \in K} (|J_k| - 1)(1 - Z_{j_k}) \leq |C| - 1 \tag{50}$$

は PCKP にとって有効な不等式である。

残念ながら (49) を満たすシナリオに関してリフティングされた被覆不等式の実用的な関連性はかなり限られている。そのため (49) を満たすシナリオに関しては、リフティングすることを検討しない。

9 まとめと今後の課題

ナップサック問題は変数が 0 または 1 しかとらないため容易に思われるが、実際は全ての荷物の詰め合わせを試すと 2^n 通りとなり、荷物の個数が多くなると計算に時間がかかる。しかし動的計画法や分枝限定法を用いることで効率的に求められる。ナップサック制約の代わりに誘導被覆 (induced cover) を用いるというアイディアを用いて、確率制約を含むナップサック問題として定式化される実際の問題に適用したものを研究していきたい。

参考文献

[1] 梅谷俊治『しっかり学ぶ数理最適化』(講談社、2020)

[2] 藤澤克樹、後藤順哉、安井雄一郎 『Excel で学ぶ OR』(オーム社、2011)

[3] 穴井宏和、斉藤努『今日から使える！組合せ最適化』(講談社、2015)

[4] 茨木俊秀『C によるアルゴリズムとデータ構造』(オーム社、2014)

[5] David G. Luenberger『Introduction to Linear and Nonlinear Programming』(Addison Wesely Publishing Company、1973)

[6] 椎名孝之『確率計画法』(朝倉書店、2015)

[7] Andrzej Ruszczynski『Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra』(Math.Program.,Ser.A93:195-215(2002))

[8] 鈴木陸斗『ナップザック問題に対する数理最適化モデルと効率的な解法』
(神奈川大学理学部数理・物理学科 2023 年度卒業研究論文要旨集:45-46、2024.3)