日本オペレーションズ・リサーチ学会
2005年春季研究発表会

# 逆凸制約付き線形計画問題に対する分枝限定法

02402070　筑波大学　*永井 秀稔　NAGAI Hidetoshi
01107230　筑波大学　久野 誉人　KUNO Takahito

## 1　逆凸制約付き線形計画問題

本研究で対象とする逆凸制約付き線形計画問題 (Linear Programs with an additional Reverse Convex constraint) とは，線形計画問題に逆凸制約が加えられた問題である．行列 $\mathbb{A} \in \mathbb{R}^{m \times n}$, 列ベクトル $\mathbf{b} \in \mathbb{R}^m$, 行ベクトル $\mathbf{c} \in \mathbb{R}^n$, 凸関数 $g : \mathbb{R}^n \mapsto \mathbb{R}^1$, に対し、以下のように定式化される:

$$P \left| \begin{array}{ll} \min\ z = & \mathbf{cx} \\ \text{s.t.} & \mathbb{A}\mathbf{x} \leq \mathbf{b} \\ & g(\mathbf{x}) \geq 0 \\ & \mathbf{x} \geq \mathbf{0}. \end{array} \right.$$

また，$D = \{\mathbf{x} \mid \mathbb{A}\mathbf{x} \leq \mathbf{b},\ \mathbf{x} \geq \mathbf{0}\}$, $G = \{\mathbf{x} \mid g(\mathbf{x}) < 0\}$ とすれば，$\min\{\mathbf{cx} \mid \mathbf{x} \in D \setminus G\}$ とも表現できる．凸関数 $g$ に対し，制約が $g(\mathbf{x}) \leq 0$ であれば実行可能領域は凸集合となるのだが，不等号が逆向きであるが故に，非凸集合となるどころか，非連結となることもあり得る．この問題は，製品開発や規模の経済効果を考慮した制約を持つ問題などをはじめ，多くの分野で見受けられる．本研究の目的は，この逆凸制約付き線形計画問題の大域的最適解を効率的に生成するアルゴリズムを構築することである．よって以降，単に最適解と言えば，大域的最適解を指すものとする．

$P$ に対し，次の仮定を置く:

**仮定 1.**　$\mathbf{c} \geq \mathbf{0}$, $\mathbf{b} \geq \mathbf{0}$, $g(\mathbf{0}) < 0$.

この仮定の下で，$P$ の最適解について以下の性質を持つことが知られている:

**性質 1.**　$\partial G = \{\mathbf{x} \mid g(\mathbf{x}) = 0\}$ と $D$ の辺との交点上に最適解が存在する．

この問題に対し，先行研究で提案されているアルゴリズムは，概ね以下の 4 種に分けられる:

- Simplex-type pivoting [2,6]: pivot により，$G$ に含まれる $D$ の端点をすべて列挙する．各端点に対し，これと，隣接する端点とを結ぶ辺上に局所解が無いか調べる．
- Alternating local search and concave minimization [3,9]: $\partial G = \{\mathbf{x} \mid g(\mathbf{x}) = 0\}$ 上の局所解を求めることと，大域性確認のための凹最小化問題を解くことを交互に行う．
- Outer approximation [1,4]: concavity cut や facial cut などを用いて，最適解を除いた $D$ の辺上の点をすべて cut する．
- Conical branch-and-bound [7,8]: 錐で分割しながら，凸関数 $g$ を線形緩和した問題を解いていく．

これらに対し本研究では，Conical branch-and-bound に属するアルゴリズムを構築する．

## 2　Conical branch-and-bound

錐分枝限定法 (Conical branch-and-bound) は分枝操作として，$\mathbf{x} \geq \mathbf{0}$ からなる錐を分割していく．錐 $\Delta$ を表現するのに $n$ 本のベクトル $\mathbf{v}^i$, $i = 1, \ldots, n$ の非負結合を用い，$\mathbb{V} = (\mathbf{v}^1, \ldots, \mathbf{v}^n)$ とすると，$\Delta = \{\mathbf{x} \mid \mathbf{x} = \mathbb{V}\boldsymbol{\lambda},\ \boldsymbol{\lambda} \geq \mathbf{0}\}$ となる．初期錐 $\Delta^0 = \{\mathbf{x} \mid \mathbf{x} \geq \mathbf{0}\}$ は，$\mathbf{v}^i$ を単位ベクトルで与えることで得られる．分割は $\cup_k \Delta^k = \Delta^0$ と，$\forall k, \ell\ (k \neq \ell)$ : $\mathrm{int}(\Delta^k) \cap \mathrm{int}(\Delta^\ell) = \emptyset$ を満たすように行う．よく用いられる分割方法は，ある錐 $\Delta$ に対し，その錐を構成する $\mathbf{v}^i$ 間の距離が最大となる 2 点 $\mathbf{v}^p, \mathbf{v}^q$ を選び，その中点 $\mathbf{v}' = (\mathbf{v}^p + \mathbf{v}^q)/2$ を求め，$\mathbf{v}^p$ あるいは $\mathbf{v}^q$ を $\mathbf{v}'$ に置き換えて 2 分割する方法である [5]. 分割された各子問題は次のように定式化される:

$$P(\Delta) \left| \begin{array}{ll} \min\ z = & \mathbf{cx} \\ \text{s.t.} & \mathbb{A}\mathbf{x} \leq \mathbf{b} \\ & g(\mathbf{x}) \geq 0 \\ & \mathbf{x} \in \Delta. \end{array} \right.$$

$P(\Delta)$ を解くことは $P$ を解くことと同程度に難しいので，より簡単な問題に緩和し，$P(\Delta)$ の下界値を求めることを考える．各 $\mathbf{v}^i$ に対し，$g(\alpha^i \mathbf{v}^i) = 0$ となる正数 $\alpha^i$ を求め，$\mathbf{W} = (\alpha^1 \mathbf{v}^1, \ldots, \alpha^n \mathbf{v}^n)$ とすると，改めて $\Delta$ は $\Delta = \{\mathbf{x} \mid \mathbf{x} = \mathbf{W}\boldsymbol{\lambda},\ \boldsymbol{\lambda} \geq \mathbf{0}\}$ と表現できるが，このとき関数 $g$ の凸性より，$\forall \mathbf{x} \in \Delta \setminus G$ : $\mathbf{e}\boldsymbol{\lambda} \geq 1$, ($\mathbf{e}$ は要素がすべて 1 の行ベクトル) が成り立つ．よって，制約 $g(\mathbf{x}) \geq 0$ の代わりに $\mathbf{e}\boldsymbol{\lambda} \geq 1$ を導入することで，線形緩和問題が得られる:

$$\overline{P}(\Delta) \left| \begin{array}{lrr} \min z = \mathbf{cx} & & \\ \text{s.t.} & \mathbf{Ax} & \leq \mathbf{b} \\ & & \mathbf{e}\boldsymbol{\lambda} \geq 1 \\ & & \mathbf{x} - \mathbf{W}\boldsymbol{\lambda} = \mathbf{0} \\ & & \boldsymbol{\lambda} \geq \mathbf{0}. \end{array} \right.$$

基本的なアルゴリズムの枠組みは以下のようになる:

**Step 0.** $\mathcal{H} := \{\Delta^0\}$. $z^* := +\infty$.

**Step 1.** $\mathcal{H} = \emptyset$ ならば，終了．そうでなければ，$\mathcal{H}$ から一つの $\Delta$ を選び出す．$\mathcal{H} := \mathcal{H} \setminus \{\Delta\}$.

**Step 2.** $\overline{P}(\Delta)$ を解き，最適解 $\overline{\mathbf{x}}(\Delta)$ と最適値 $\overline{z}(\Delta)$ を求める．$\overline{\mathbf{x}}(\Delta)$ から局所探索により，$P$ の実行可能解 $\widetilde{\mathbf{x}}(\Delta)$ とその目的関数値 $\widetilde{z}(\Delta)$ を求める．

**Step 3** (限定操作). $\widetilde{z}(\Delta) < z^*$ であれば，$z^* := \widetilde{z}(\Delta)$, $\mathbf{x}^* := \widetilde{\mathbf{x}}(\Delta)$. $\overline{z}(\Delta) \geq z^*$ であれば，Step 1 へ．

**Step 4** (分枝操作). $\Delta$ を分割し，それらを $\mathcal{H}$ に加える．Step 1 へ．

アルゴリズム終了時の $\mathbf{x}^*, z^*$ がそれぞれ $P$ の最適解，最適値となる．

## 3 下界値の強化

$\overline{P}(\Delta)$ の双対問題:

$$D\overline{P}(\Delta) \left| \begin{array}{llll} \max z = -\boldsymbol{\pi}\mathbf{b} + \eta & & & \\ \text{s.t.} & -\boldsymbol{\pi}\mathbf{A} & +\boldsymbol{\rho} & = \mathbf{c} \\ & & \eta\mathbf{e} - \boldsymbol{\rho}\mathbf{W} & \leq \mathbf{0} \\ & & \boldsymbol{\pi}, \eta & \geq 0 \end{array} \right.$$

の $\mathbf{Ax} \leq \mathbf{b}$ に対する双対最適解 $\widetilde{\boldsymbol{\pi}}$ を用い，ラグランジュ緩和により下界値の強化を図る．$P(\Delta)$ の任意の実行可能解で $\mathbf{cx} \geq \overline{z}(\Delta)$ なので，この制約を $P(\Delta)$ に加え，一方 $\mathbf{Ax} \leq \mathbf{b}$ を $\widetilde{\boldsymbol{\pi}}$ でラグランジュ緩和する:

$$L(\Delta, \widetilde{\boldsymbol{\pi}}) \left| \begin{array}{lr} \min z = (\mathbf{c} + \widetilde{\boldsymbol{\pi}}\mathbf{A})\mathbf{x} - \widetilde{\boldsymbol{\pi}}\mathbf{b} & \\ \text{s.t.} & \mathbf{cx} \geq \overline{z}(\Delta) \\ & g(\mathbf{x}) \geq 0 \\ & \mathbf{x} \in \Delta. \end{array} \right.$$

この問題は，$\mathbf{cx} = \overline{z}(\Delta)$ と $\mathbf{x} \in \Delta$ からなる単体の各辺と，$g(\mathbf{x}) \geq 0$ との交点を調べるだけで最適解が得られるので，高速に解くことができる．また，$\widetilde{\boldsymbol{\pi}}$ が双対最適解であることより，以下の関係が保証されるので，下界値の改善が期待できる．

**命題 1.** $L(\Delta, \widetilde{\boldsymbol{\pi}})$ の最適値を $z(\Delta, \widetilde{\boldsymbol{\pi}})$ とすると，$z(\Delta, \widetilde{\boldsymbol{\pi}}) \geq \overline{z}(\Delta)$.

## 参考文献


[1] J. Fülöp: A finite cutting plane method for solving linear programs with an additional reverse convex constraint, *European Journal of Operational Research*, **44** (1990), 395–409.

[2] R. J. Hillestad: Optimization problems subject to a budget constraint with economies of scale, *Operations Research*, **23** (1975), 1091–1098.

[3] R. J. Hillestad and S. E. Jacobsen: Linear programs with an additional reverse convex constraint, *Applied Mathematics and Optimization*, **6** (1980), 257–269.

[4] R. J. Hillestad and S. E. Jacobsen: Reverse convex programming, *Applied Mathematics and Optimization*, **6** (1980), 63–78.

[5] R. Horst and H. Tuy: *Global Optimization: Deterministic Approaches*, Springer-Verlag, Berlin, 3 edition (1996).

[6] S. E. Jacobsen and K. Moshirvaziri: Computational experience using an edge search algorithm for linear reverse convex programs, *Journal of Global Optimization*, **9** (1996), 153–167.

[7] K. Moshirvaziri and M. A. Amouzegar: A subdivision scheme for linear programs with an additional reverse convex constraint, *Asia-Pacific Journal of Operations Research*, **15** (1998), 179–192.

[8] L. D. Muu: A convergent algorithm for solving linear programs with an additional reverse convex constraint, *Kybernetika*, **21** (1985), 428–435.

[9] N. V. Thuong and H. Tuy: A finite algorithm for solving linear programs with an additional reverse convex constraint, *Lecture Note in Economics and Mathematical Systems*, **255** (1984), 291–302.