演習3 (6/9)

31010 佐藤秀明

概要

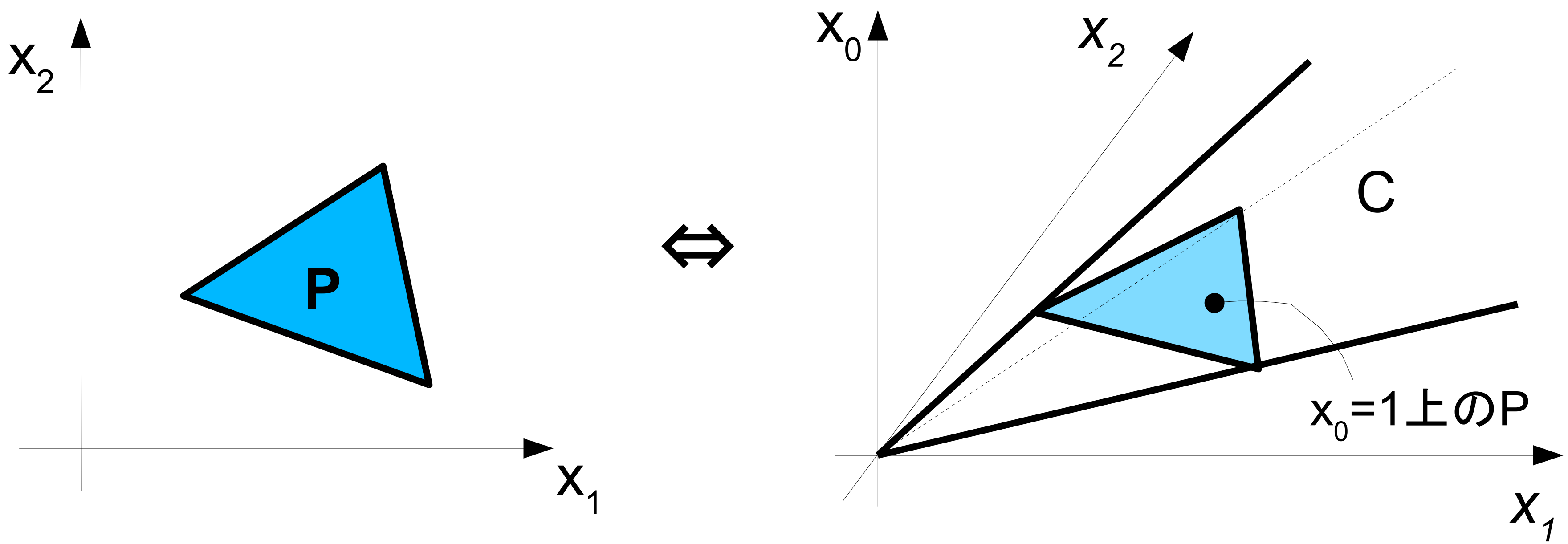
- 凸包計算アルゴリズム
  - 話を簡単にするために
    - 斉次化
    - 極変換
    - 双対性
  - Double Description Method
    - Fourier-Motzkinとの関連性
    - 基本アルゴリズムとその改良
- これからの目標

H表現とV表現(復習)

- 錘のH表現: $\left\{ \boldsymbol{x} \in \mathbb{R}^d : A\boldsymbol{x} \leq \mathbf{0} \right\}$
  - Aの各行の不等式を満たす領域
- 錘のV表現: $\left\{ \boldsymbol{x} \in \mathbb{R}^d : \exists \boldsymbol{t} \in \mathbb{R}^m : \boldsymbol{x} = R\boldsymbol{t}, \boldsymbol{t} \geq \mathbf{0} \right\}$
  - Rの各列のベクトルで張られる領域

話を簡単にするために(1)

- 斉次化
  - 多面集合Pを錘Cに置き換えて考える
    - 新しい座標x_0を導入し、次元を1つ上げる
    - 超平面$x_0=1$とCの共通部分がPと等しくなるようにする
  - PからC、CからPの相互変換が可能



話を簡単にするために(2)

- 斉次化(続き)
  - H表現

$$P=\{\boldsymbol{x}\in\mathbb{R}^d : A\boldsymbol{x}\le\boldsymbol{z}\}$$

$$C=\left\{\begin{pmatrix}x_0\\ \boldsymbol{x}\end{pmatrix}\in\mathbb{R}^{d+1} : \begin{pmatrix}-1 & 0\ldots0\\ -\boldsymbol{z} & A\end{pmatrix}\begin{pmatrix}x_0\\ \boldsymbol{x}\end{pmatrix}\le\begin{pmatrix}0\\ \mathbf{0}\end{pmatrix}\right\}$$

  - V表現

$$P=\{\boldsymbol{x}\in\mathbb{R}^d : \exists\boldsymbol{t}\in\mathbb{R}^m, \boldsymbol{u}\in\mathbb{R}^{m'} : \boldsymbol{x}=V\boldsymbol{t}+Y\boldsymbol{u}, \boldsymbol{t}\ge\mathbf{0}, \boldsymbol{u}\ge\mathbf{0}, \sum_i t_i=1\}$$

$$C=\left\{\begin{pmatrix}x_0\\ \boldsymbol{x}\end{pmatrix}\in\mathbb{R}^{d+1} : \exists\begin{pmatrix}\boldsymbol{t}\\ \boldsymbol{u}\end{pmatrix}\in\mathbb{R}^{m+m'} : \begin{pmatrix}x_0\\ \boldsymbol{x}\end{pmatrix}=\begin{pmatrix}1..1 & 0..0\\ V & Y\end{pmatrix}\begin{pmatrix}\boldsymbol{t}\\ \boldsymbol{u}\end{pmatrix}, \begin{pmatrix}\boldsymbol{t}\\ \boldsymbol{u}\end{pmatrix}\ge\begin{pmatrix}\mathbf{0}\\ \mathbf{0}\end{pmatrix}\right\}$$

話を簡単にするために(3)

- 極
  - 錘Cに対するH表現

    $$C=\{\boldsymbol{x}\in\mathbb{R}^d : A\boldsymbol{x}\leq\mathbf{0}\}$$

  - から、Cの極C$^\triangle$はV表現で以下のように表される。

    $$C^\triangle=\{\boldsymbol{x}\in\mathbb{R}^d : \exists\boldsymbol{t}\in\mathbb{R}^m : \boldsymbol{x}=A^T\boldsymbol{t}, \boldsymbol{t}\geq\mathbf{0}\}$$

  - $(C^\triangle)^\triangle=C$

話を簡単にするために(4)

- 錘のH表現をV表現に変換する方法を考えるだけでよい
  - 極に関する双対性に注目

PのH表現 ⇔ CのH表現 → CのV表現 ⇔ PのV表現

斉次化　↑極　極↓　斉次化

$C^{\triangle}$のV表現 ← $C^{\triangle}$のH表現

Double Description Method(1)

- 概要
  - 集合Kを{i:1≦i≦m}の部分集合とおく。
  - H表現の表現行列Aを構成する各行の不等式に、1つずつ識別番号i(1≦i≦m)を割り振る。
  - 不等式の集合A_Kを以下のようにおく。

$$A_K = \{ A_i : i \in K \}$$

  - H表現A_Kと同値なV表現Rが分かっているときに、

$$A_{K+i} \left(i \notin K \right)$$

    と同値なV表現R´を求める。後はこの操作を、すべての不等式がA_Kに加えられるまで繰り返す。

Double Description Method(2)

- 基本アルゴリズム
  1. 適当な初期状態の組(A_K, R)を用意する
  2. {1, 2, ... m}\Kから1つ取り出してiとする
  3. Rに属する各ベクトルr_jを以下の3種類に分類する

$$J^{+}=\{j\in R : A_i \boldsymbol{r}_j > 0\}$$

$$J^{0}=\{j\in R : A_i \boldsymbol{r}_j = 0\}$$

$$J^{-}=\{j\in R : A_i \boldsymbol{r}_j < 0\}$$

Double Description Method(3)

- 基本アルゴリズム(続き)
  4. 一方がJ^+に、他方がJ^-に属するすべての列ベクトルの対(r^+, r^-)について、2者の内分点でかつA_i上にあるようなベクトルr^{+-}を新たに作る。ここで新たに作られたベクトルの集合をJ^{+-}としてまとめておく
  5. $R := J^{+-} \cup J^0 \cup J^-$、$K := K+i$ とする
  6. 2.～5.を繰り返す

Double Description Method(4)

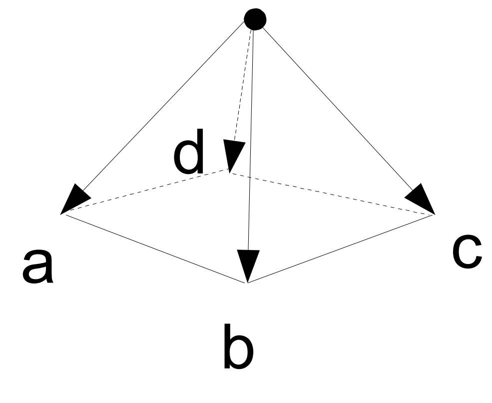
- 初期状態(A_K, R)の選び方
  - 1次独立な不等式l個の集合をA_Kとし、

    $$A_K R = -I_l$$

    をみたすようにRの各列ベクトルをとればよい。
  - Cの取りうる空間がある程度予想できているのなら、Cを真に含むような錘を初期状態としてとってもよい。

Double Description Method(5)

- 改良案(1)
  - J^+とJ^-のすべての対について計算するのは無駄
  - 一方がJ^+に、他方がJ^-にそれぞれ属し、かつ2者が「隣接」しているベクトルの対についてのみ、内分点を求めることにする
    - 「rとr´が隣接」＝錘Cの面がrとr´を含み、かつそれ以外のベクトルr´´∈Rを含まないこと
    - $A_i r = A_i r´ = 0$をみたすようなすべての不等式Aiで構成される行列A_kのrankがd-2であることと同値
      - rank計算は重そう…



Double Description Method(6)

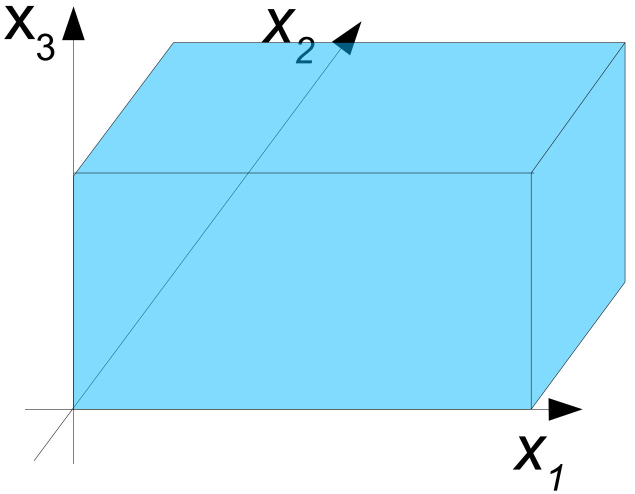
- 改良案(1)の例:

$$A = \begin{pmatrix} 3 & -2 & -2 \\ -2 & 3 & -2 \\ -2 & -2 & 3 \end{pmatrix}$$

- 初期状態を$x_1, x_2, x_3 \geqq 0$、すなわち

$$A_K = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

とし、Aの各行を順に加えてゆく。



Double Description Method(7)

- 改良案(1)の例(続き)

$$A_1 = \begin{pmatrix} 3 & -2 & 3 \end{pmatrix}$$

について

$$J^+ = \left\{ r_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right\},\ J^- = \left\{ r_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, r_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$$

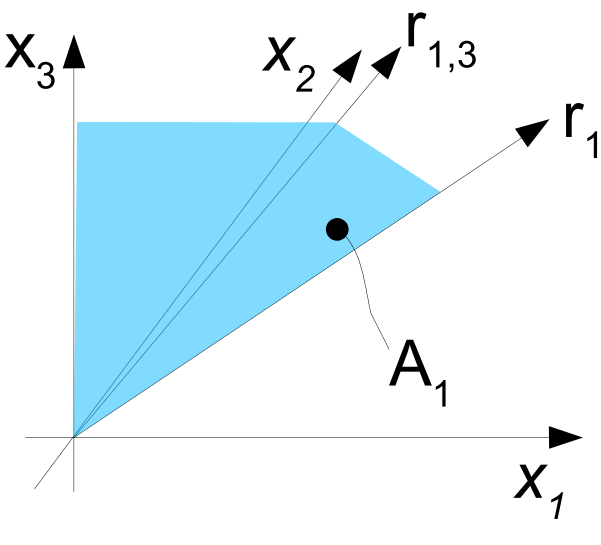
この中で隣接する対は $(1,2),(1,3)$

だから

$$J^{+-} = \left\{ r_{1,2} = \begin{pmatrix} 2 \\ 3 \\ 0 \end{pmatrix}, r_{1,3} = \begin{pmatrix} 2 \\ 0 \\ 3 \end{pmatrix} \right\}$$

以上より

$$R' = \{ r_2, r_3, r_{1,2}, r_{1,3} \}$$

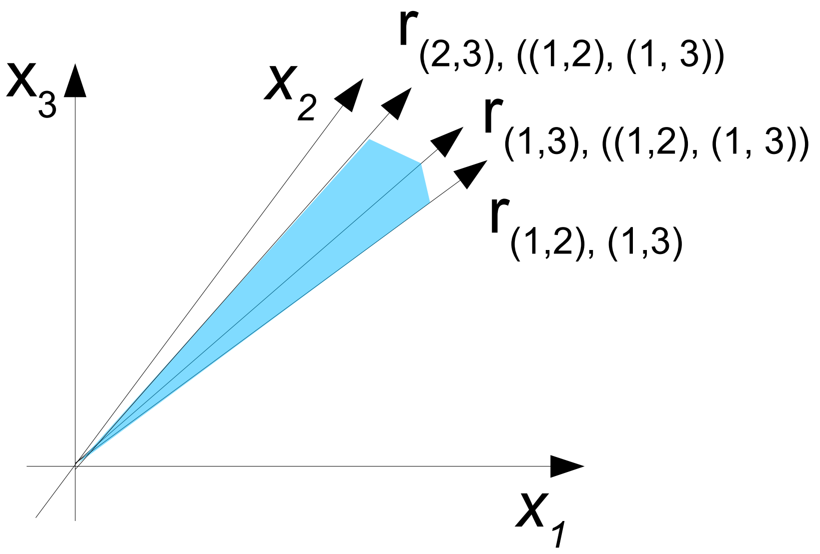


Double Description Method(8)

- 改良案(1)の例（続き）
  - A_2,A_3について同じことを繰り返すと

$$R=\left\{r_{(1,2),(1,3)}=\begin{pmatrix}2\\2\\1\end{pmatrix}, r_{(1,3),((1,2),(1,3))}=\begin{pmatrix}2\\1\\2\end{pmatrix}, r_{(2,3),((1,2),(1,3))}=\begin{pmatrix}1\\2\\2\end{pmatrix}\right\}$$

が得られる。



Double Description Method(9)

- 改良案(2)
  - 超平面A_iの導入により新たにベクトルrを発生させることが明らかな隣接関係のペアを、前もってリストL_iに保持
  - 隣接性計算の繰り返しを抑止
  - i回目の繰り返しにおける作業
    1. L_iに記録されている隣接関係の対(r, r´)について、rとr´の内分点でかつA_i上に存在するベクトルを計算し、R´を作る
    2. R´に含まれ、かつ少なくとも一方がA_i上に存在するようなベクトルの対(r, r´)について、それらがk回目の繰り返し($i<k\leqq m$)においてJ^+とJ^-に分断されて新たなベクトルr´´を発生することが明らかであれば、rとr´が隣接であることを確認して、(r, r´)をL_kに追加する

Double Description Method(10)

- 改良案(3)
  - 隣接しているかどうかの判定を先に必要条件で絞り込むことにより、rankの計算をできるだけ回避する
  - $(r, r')\in(J^+, J^-)$が
    1. 他の隣接を継承してできた隣接であれば(一方が他方の親であれば・・・これには自分の親を保持する必要がある) または
    2.
       1. 今回の繰り返しより前に導入された平面A_l上に2点とも存在して(これには各ベクトルが誕生した時期を保持する必要がある)、かつ
       2. $A_l r = A_l r' = 0$をみたすようなA_lが今までにd-2個以上導入されていて、かつ
       3. (r, r')が隣接であれば

  その内分点r''をRに追加する

Double Description Method(11)

- 実験結果(論文より)
  - 改良案(2)は改良案(1)より明らかに速い
  - 超平面A_iが隣接ペアを分断することが多い場合は改良案(2)が改良案(3)よりやや優勢
    - 平面上にベクトルが乗ることが多い場合は改良案(3)のほうが有利と考えられる
  - 不等式を適用する順序

    1.lexmin …係数の辞書式順序(static)

    2.mincutoff …$|J^+|$を最小にする(dynamic)

    3.maxcutoff …$|J^+|$を最大にする(dynamic)

    - 速さはlexmin > mincutoff > maxcutoff = ランダム

References

- Double Description Method
  - Komei Fukuda and Alain Prodon, “Double Description Method Revisited”, 1996
  - http://www.cs.mcgill.ca/~fukuda/download/paper/ddrev960315.ps.gz
- 斉次化・双対性
  - G.M.ツィーグラー、凸多面体の科学(シュプリンガー・フェアラーク東京)