半正定値計画の問題記述&解決能力

中田　和秀

半正定値計画問題（SDP）は線形計画問題・凸2次計画問題・2次錐計画問題などを含んだ非常に適用範囲の広い最適化問題である．本稿では，SDPの概要について述べた後，SDPのモデル記述能力と問題解決能力について説明を行う．

キーワード：SDP，モデル化，内点法

1. はじめに

半正定値計画問題（SemiDefinite Programming: SDPと略）は非常に強力なモデル記述能力を持ち，主双対内点法により実用的に解くことができるという特徴を持った最適化問題である．また，この問題を解くためのソフトウェアも数多く開発されている．歴史的な経緯をみると，元々線形計画問題の解法であった主双対内点法に対し，そのアルゴリズムが適用できる問題自体を拡張することによってSDP（の標準形）が導出された．そのため，SDPは主双対内点法を適用するには都合のよい表現であるものの，現実問題を記述するには直感的ではなく分かりづらい．その結果，SDPとして解けば効率よく最適解が得られるにもかかわらず，その機会が失われていることも多いというのが現状である．このため本稿では，最適化問題をSDPとして記述する方法を中心に解説を行う．

2. SDPとは

本節では，SDPの標準形とその特徴について述べる．SDPの詳細について知りたい場合には，文献[7][11][12]などをご覧いただきたい．

2.1 標準形

$\mathbb{R}^{m\times n}$を$m\times n$の実行列の集合，$\mathcal{S}^n$を$n\times n$の実対称行列の集合と定義する．今，$\boldsymbol{F}_k\in\mathcal{S}^n(k=0,1,\cdots,m)$と$\boldsymbol{c}\in\mathbb{R}^m$がデータとして与えられているとしよう．このとき，次のように定式化される最適化問題がSDPの主問題と双対問題である．

主問題：

$$
\begin{aligned}
&\text{最小化} && \sum_{k=1}^{m} c_k x_k\\
&\text{制約条件} && \boldsymbol{X}=\sum_{k=1}^{m}\boldsymbol{F}_k x_k-\boldsymbol{F}_0,\\
& && \boldsymbol{X}\succeq\boldsymbol{O},\quad \boldsymbol{X}\in\mathcal{S}^n.
\end{aligned}
\tag{1}
$$

双対問題：

$$
\begin{aligned}
&\text{最大化} && \boldsymbol{F}_0\bullet\boldsymbol{Y}\\
&\text{制約条件} && \boldsymbol{F}_k\bullet\boldsymbol{Y}=c_k(k=1,2,\cdots,m),\\
& && \boldsymbol{Y}\succeq\boldsymbol{O},\quad \boldsymbol{Y}\in\mathcal{S}^n.
\end{aligned}
\tag{2}
$$

ただし，$\boldsymbol{U}\bullet\boldsymbol{V}$は行列$\boldsymbol{U}$と$\boldsymbol{V}$の内積を意味し，$\boldsymbol{U}\bullet\boldsymbol{V}\equiv\sum_{i=1}^{n}\sum_{j=1}^{n}U_{ij}V_{ij}$である．また，$\boldsymbol{U}\succeq\boldsymbol{V}$は行列$\boldsymbol{U}-\boldsymbol{V}$が半正定値行列，$\boldsymbol{U}\succ\boldsymbol{V}$は行列$\boldsymbol{U}-\boldsymbol{V}$が正定値行列であることを意味する[1]．問題(1)と(2)は双対関係にあるため，どちらの問題を解いても同じことである[2]．制御分野などで現れる最適化問題は(1)の形に，組合せ最適化のSDP緩和などは(2)の形に定式化し，それを解くことが多い．

2.2 特徴

まず，よく知られた線形計画問題（LP）との関係をみてみよう．$\boldsymbol{f}_k\in\mathbb{R}^n(k=0,1,\cdots,m)$と$\boldsymbol{c}\in\mathbb{R}^m$を定数とする次のようなLPの主問題と双対問題を考える．

$$
\begin{aligned}
&\text{最小化} && \sum_{k=1}^{m} c_k x'_k\\
&\text{制約条件} && \boldsymbol{x}=\sum_{k=1}^{m}\boldsymbol{f}_k x'_k-\boldsymbol{f}_0,\\
& && \boldsymbol{x}\geq\boldsymbol{0},\quad \boldsymbol{x}\in\mathbb{R}^n.\\
&\text{最大化} && \boldsymbol{f}_0^T\boldsymbol{y}\\
&\text{制約条件} && \boldsymbol{f}_k^T\boldsymbol{y}=c_k(k=1,2,\cdots,m),\\
& && \boldsymbol{y}\geq\boldsymbol{0},\quad \boldsymbol{y}\in\mathbb{R}^n.
\end{aligned}
$$

なかた　かずひで
東京工業大学 大学院社会理工学研究科経営工学専攻
〒152-8552 目黒区大岡山2-12-1

[1] このように定義した$\succeq$は半順序関係となる．
[2] 4節で述べる主双対内点法では両方の問題を同時に解く．

LPにおけるベクトル $\boldsymbol{f}_i (i=0, 1, \cdots, m)$，$\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ と，SDPにおける行列 $\boldsymbol{F}_i (i=0, 1, \cdots, m)$，$\boldsymbol{X}, \boldsymbol{Y} \in \mathcal{S}^n$ が対応している．また，LPの変数ベクトル $\boldsymbol{x}$，$\boldsymbol{y}$ に対する非負条件が，SDPでは変数行列 $\boldsymbol{X}$，$\boldsymbol{Y}$ の固有値に対する非負条件に置き換わっている[3]．このことより，SDPはLPの（ある種の）拡張であると捉えることができる．

次に，SDPと凸計画問題の関係について考察する．主問題(1)と双対問題(2)の実行可能領域は，共に線形制約で表現できる集合（アフィン空間）と半正定値行列の集合の共通部分になっている．また，目的関数は線形関数である．凸集合である実行可能領域上で凸関数を最小化するため，SDPは凸計画問題（CP）の枠組みに属する．ただし，同様に凸計画問題であるLPや凸2次計画問題（QP）がかなり限定された問題群であるのに対し，SDPは3節で述べるように非常に多様なタイプの制約を記述できる．そのため，SDPは線形計画問題や凸2次計画問題よりも相当大きな問題群であり，凸計画問題に近いという実感がある（図1を参照）．

一般に凸計画問題は次のように定式化される．

凸計画問題：

最小化　$f(\boldsymbol{x})$

制約条件　$f_k(\boldsymbol{x}) \leq 0 \quad (k=1, 2, \cdots, m)$,

$\boldsymbol{x} \in S.$

凸計画問題が問題の構造を凸関数 $f(\boldsymbol{x})$ や $f_k(\boldsymbol{x})$ で表現しているのに対し，SDPでは問題の構造を主に行列の半正定値条件で表現している所が特徴的である(残りの部分は線形でしかない)．半正定値行列の集合には非常に優れた性質があり，その性質を利用して内点法などの解法が構築されている．しかし，現実に解きたい問題は，SDPの標準形とは程遠い形をしていることも多く，それを主問題(1)あるいは双対問題(2)の形式に帰着させなければならないのが難点である．

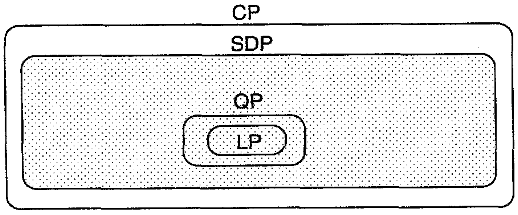

図1　様々な凸計画問題の包含関係

3．SDPで解決できるモデル

本節では，SDPに帰着できるタイプの制約や問題を紹介する．さらに詳しい解説が文献[1]に書かれているので，興味がある方はそちらを参照していただきたい．

3.1　ブロック対角行列

次のような半正定値条件が複数存在する問題を解きたいとしよう．ただし，$\boldsymbol{F}_{ki} \in \mathcal{S}^{n_i} (i=1, 2, \cdots, l, k=1, 2, \cdots, m)$ は定数である．

主問題2：

最小化　$\sum_{k=1}^{m} c_k x_k$

制約条件　$\boldsymbol{X}_i = \sum_{k=1}^{m} \boldsymbol{F}_{ki} x_k - \boldsymbol{F}_{0i} (i=1, 2, \cdots, l)$,

$\boldsymbol{X}_i \succeq \boldsymbol{O}, \quad \boldsymbol{X}_i \in \mathcal{S}^{n_i} (i=1, 2, \cdots, l)$

双対問題2：

最大化　$\sum_{i=1}^{l} \boldsymbol{F}_{0i} \bullet \boldsymbol{Y}_i$

制約条件　$\sum_{i=1}^{l} \boldsymbol{F}_{ki} \bullet \boldsymbol{Y}_i = c_k (k=1, 2, \cdots, m)$,

$\boldsymbol{Y}_i \succeq \boldsymbol{O}, \quad \boldsymbol{Y}_i \in \mathcal{S}^{n_i} (i=1, 2, \cdots, l)$.

なお，2つの変数の値が等しいという条件は，線形制約として表現できるため，変数 $\boldsymbol{X}_i (i=1, 2, \cdots, l)$ の各変数は完全に独立でなくても構わない．$\boldsymbol{Y}_i (i=1, 2, \cdots, l)$ についても同様である．この問題は，ブロック対角行列

$$\boldsymbol{F}_k = \begin{pmatrix} \boldsymbol{F}_{k1} & \boldsymbol{O} & \cdots & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{F}_{k2} & \cdots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{F}_{kl} \end{pmatrix}$$

を作ることにより，標準形(1)と(2)に帰着できる．このとき，最適解 $\boldsymbol{X}$，$\boldsymbol{Y}$（の一つ）は，$\boldsymbol{F}_k$ と同じ構造を持ったブロック対角行列となる．そのブロック対角部分を抜き出すと，**主問題2**と**双対問題2**の最適解になる．

もっとも，行列変数の空間として，$\mathcal{S}^{\sum_{i=1}^{l} n_i}$ ではなく，$\mathcal{S}^{n_1} \times \mathcal{S}^{n_2} \times \cdots \times \mathcal{S}^{n_l}$ を選んでいるという見方も可能であり，実はそちらの解釈の方が自然である[4]．

3.2　様々なタイプの制約

半正定値条件で記述できる制約条件の例を挙げる．関数 $f(\boldsymbol{x})$ の最小化は，新しい変数 $t \in \mathbb{R}$ を導入し，

[3] 行列が半正定値であることと，行列のすべての固有値が非負であることは必要十分である．

[4] この解釈を利用して内点法を実装すると，不必要な演算を省略でき計算効率がよくなる．

$f(\boldsymbol{x}) \leq t$ という制約条件の元で，関数 t の最小化に変形することができる．すると，目的関数は変数 $(t, \boldsymbol{x})$ に対して線形となり，目的関数 $f(\boldsymbol{x})$ の非線形性による複雑さ（難しさ）を，制約条件に移すことができる．よって，以下では目的関数については考えない．

以下では，大文字のボールド体（$\boldsymbol{X}, \boldsymbol{A}$ など）は対称行列，チルダ付き（$\tilde{\boldsymbol{X}}, \tilde{\boldsymbol{A}}$ など）は対称とは限らない一般行列，小文字のボールド体（$\boldsymbol{x}, \boldsymbol{a}$ など）はベクトル，小文字のイタリック体（x, a など）は実数を意味することにする．また，$\boldsymbol{A}$，$\boldsymbol{B}$，$\boldsymbol{C}$，$\boldsymbol{D}$，$\boldsymbol{E}$，$\boldsymbol{b}$，c は定数であり，残りは変数である[5]．$\boldsymbol{I}$ は適当なサイズの単位行列である．

ベクトルに対する制約

- $\boldsymbol{x} \geq \boldsymbol{0} \iff \begin{pmatrix} x_1 & 0 & \cdots & 0 \\ 0 & x_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x_n \end{pmatrix} \geq \boldsymbol{O}$
- $\|\boldsymbol{x}\|_2 \leq y \iff \begin{pmatrix} y & \boldsymbol{x}^T \\ \boldsymbol{x} & y\boldsymbol{I} \end{pmatrix} \geq \boldsymbol{O}$
- $x^2 \leq yz,\ y, z \geq 0 \iff \begin{pmatrix} y & x \\ x & z \end{pmatrix} \geq \boldsymbol{O}$
- $\boldsymbol{x}^T\boldsymbol{x} \leq yz,\ y, z \geq 0 \iff \begin{pmatrix} y & \boldsymbol{x}^T \\ \boldsymbol{x} & z\boldsymbol{I} \end{pmatrix} \geq \boldsymbol{O}$
- $\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x} + c \leq 0$（$\boldsymbol{A}$ は半正定値行列）
  $\iff \begin{pmatrix} -\boldsymbol{b}^T\boldsymbol{x} - c & \boldsymbol{x}^T\boldsymbol{A}^{1/2} \\ \boldsymbol{A}^{1/2}\boldsymbol{x} & \boldsymbol{I} \end{pmatrix} \geq \boldsymbol{O}$

これらの結果より，SDP は線形計画問題・凸 2 次計画問題・2 次制約付き 2 次計画問題・2 次錐計画問題をすべて含んだ問題群であることが分かる．

固有値や特異値を用いた制約

$\lambda_k(\boldsymbol{X})$ は対称行列 $\boldsymbol{X}$ の中の（重複を含め）k 番目に大きな固有値，$\sigma_k(\tilde{\boldsymbol{X}})$ は行列 $\tilde{\boldsymbol{X}}$ の中の（重複を含め）k 番目に大きな特異値を意味する．

- $\lambda_1(\boldsymbol{X}) \leq y \iff y\boldsymbol{I} - \boldsymbol{X} \geq \boldsymbol{O}$
- $\sum_{i=1}^{k} \lambda_i(\boldsymbol{X}) \leq y \iff \begin{cases} y - ks - \boldsymbol{Z}\cdot\boldsymbol{I} \geq 0 \\ \boldsymbol{Z} \geq \boldsymbol{O} \\ \boldsymbol{Z} - \boldsymbol{X} + s\boldsymbol{I} \geq \boldsymbol{O} \end{cases}$
- $\sigma_1(\tilde{\boldsymbol{X}}) \leq y \iff \|\tilde{\boldsymbol{X}}\|_2 \leq y$
  $\iff \begin{pmatrix} y\boldsymbol{I} & \tilde{\boldsymbol{X}}^T \\ \tilde{\boldsymbol{X}} & y\boldsymbol{I} \end{pmatrix} \geq \boldsymbol{O}$
- $\sum_{i=1}^{k} \sigma_i(\tilde{\boldsymbol{X}}) \leq y \iff \begin{cases} \boldsymbol{X} = \begin{pmatrix} \boldsymbol{O} & \tilde{\boldsymbol{X}}^T \\ \tilde{\boldsymbol{X}} & \boldsymbol{O} \end{pmatrix} \\ y - ks - \boldsymbol{Z}\cdot\boldsymbol{I} \geq 0 \\ \boldsymbol{Z} \geq \boldsymbol{O} \\ \boldsymbol{Z} - \boldsymbol{X} + s\boldsymbol{I} \geq \boldsymbol{O} \end{cases}$
- $t \leq \det(\boldsymbol{X})^{1/n},\quad \boldsymbol{X} \geq \boldsymbol{O} \iff$
  $\begin{pmatrix} \boldsymbol{X} & \boldsymbol{L} + \boldsymbol{D} \\ \boldsymbol{L}^t + \boldsymbol{D} & \boldsymbol{D} \end{pmatrix} \geq \boldsymbol{O},\ t^n \leq D_{11}D_{22}D_{nn}$
  ただし，$\boldsymbol{L}$ は対角部分を含まない下三角行列で，$\boldsymbol{D}$ は対角行列とする．右辺の 2 番目の制約は半正定値条件で表現できるが，ここでは割愛する．

以上のように，SDP では固有値情報を直接利用した最適化が行える．このことを利用して，行列近似や楕円体の体積を最小化する問題を解くことができる．

LMI（線形行列不等式）による制約

- $\tilde{\boldsymbol{X}}\boldsymbol{Y}^{-1}\tilde{\boldsymbol{X}}^T \leq \boldsymbol{Z},\quad \boldsymbol{Y} > \boldsymbol{O}$
  $\iff \begin{pmatrix} \boldsymbol{Y} & \tilde{\boldsymbol{X}}^T \\ \tilde{\boldsymbol{X}} & \boldsymbol{Z} \end{pmatrix} \geq \boldsymbol{O},\quad \boldsymbol{Y} > \boldsymbol{O}$
- $\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}} + \boldsymbol{Y} + \boldsymbol{A} \leq \boldsymbol{O} \iff \begin{pmatrix} -\boldsymbol{Y} - \boldsymbol{A} & \tilde{\boldsymbol{X}}^T \\ \tilde{\boldsymbol{X}} & \boldsymbol{I} \end{pmatrix} \geq \boldsymbol{O}$
- $(\boldsymbol{AXB})(\boldsymbol{AXB})^T + \boldsymbol{CXD} + (\boldsymbol{CXD})^T + \boldsymbol{E} \leq \boldsymbol{Y}$
  $\iff \begin{pmatrix} \boldsymbol{I} & (\boldsymbol{AXB})^T \\ \boldsymbol{AXB} & \boldsymbol{Y} - \boldsymbol{E} - \boldsymbol{CXD} - (\boldsymbol{CXD})^T \end{pmatrix} \geq \boldsymbol{O}$

システムと制御・構造最適化・量子化学などの分野に現れる最適化問題の中には，LMI によって表現できるものも多い．なお制御分野では，例えば $\boldsymbol{AX} + \boldsymbol{XA}^T$ が負定値となるような対称行列 $\boldsymbol{X}$ の「存在性」を知りたいことがある．その場合，$\boldsymbol{AX} + \boldsymbol{XA}^T - t\boldsymbol{I}$ が半負定値という制約条件の元で，変数 t を最小化する SDP を解き，最適値が 0 未満になるかどうかで判断できる．

非負多項式による制約

$a_k \geq 0, r_k$ を 1 以上の有理数とする[6]．このとき，

$$\sum_{k=1}^{t} a_k x^{r_k} \leq y,\quad x \geq 0$$

という制約は，半正定値条件で記述できる．以下では，手順を踏んでそのことを確認する．

1. $x_0^4 \leq x_1x_2x_3x_4,\ x_1, x_2, x_3, x_4 \geq 0$
   $\iff \begin{cases} y_1 \leq x_1x_2, & y_2 \leq x_3x_4 \\ x_0 \leq y_1y_2, & x_1, x_2, x_3, x_4, y_1, y_2 \geq 0 \end{cases}$
   が成り立つ．すでに述べたように $x^2 \leq yz, y, z$

[5] 変数の一部または全部を固定して定数にしてもよい．

[6] 有理数は実数に対して稠密なので，r_k は近似的に 1 以上の実数とみなしてもよいかもしれない．

≥ 0 という条件は半正定値条件で記述できるため，左辺が半正定値条件で記述できることがわかる．

2. 1.の手順を繰り返すことにより，2の冪乗 s に対し，
$x_0^s \leq x_1 x_2 \cdots x_s,\quad x_1, x_2, \cdots x_s \geq 0$
は，半正定値条件で記述することができる．

3. $p, q \in \mathbb{N}, q \geq p$ に対し，
$x^{q/p} \leq y,\ x \geq 0 \iff x^s \leq y^p x^{s-q} 1^{q-p},\ x, y \geq 0$
が成り立つ．ただし，s は q より大きな2の冪乗とする．右辺は2. の特殊な場合であるから，左辺は半正定値条件で記述することができる[7]．

4. $r_k = q_k/p_k$ としたとき，
$$\begin{cases}\sum_{k=1}^{t} a_k x^{r_k} \leq y \\ x \geq 0\end{cases} \iff \begin{cases} x^{p_k/q_k} \leq y_k \quad (k=1, 2, \cdots, t) \\ \sum_{k=1}^{t} a_k y_k \leq y,\ x \geq 0\end{cases}$$
となる．3.を用いると，$\sum_{k=1}^{t} a_k x^{r_k} \leq y, x \geq 0$ は，半正定値条件で記述できる．

非負多項式による制約を他の制約と組み合わせることにより，様々なタイプの制約条件を半正定値条件に帰着させることができる．それにより，下方部分積率をリスクとしたポートフォリオ選択問題などをSDPとして解くことができる．

注意点

ある最適化問題をSDPの標準形に帰着させる方法は，一般に複数存在する．それらは，最適化問題としては同等かもしれないが，そのSDPを解いたときの計算時間は大きく異なる可能性がある．このため，ただSDPの標準形を作成できたらよいのではなく，解きやすい標準形にすることが望ましい．しかしながら，問題の解きやすさはアルゴリズムや実装に依存することもあり，一概に扱うことは難しい．ここでは，効率よく解けるSDPを作るための一般的な指針を3つ述べるに留める．

- SDPの問題サイズに直結する n や m を小さくする．
- 定数行列 $\boldsymbol{F}_k (k=1, 2, \cdots, m)$ に含まれる非ゼロ要素の数を減らす．
- 3.1節で述べたブロック対角な構造をできるだけ活用する．なお，多くのソルバーではブロック対角構造を入力することができる．

[7] もっとも，q に比例する程度にSDPのサイズは大きくなる．

また，主問題(1)と双対問題(2)では，実行可能集合を定義するためのアフィン空間の表現方法が異なっている．主問題(1)では行列の線形結合で表しており，双対問題(2)では線形方程式の解として表している．解きたい最適化問題をどちらの形式で記述するかによって計算効率が変わるため，吟味する必要がある．

3.3 SDP緩和

組合せ最適化問題をSDPに緩和して解き，その最適解から元問題の近似解を構成したり，得られた下界値を分枝限定法に利用することがよく行われている．本節ではその手順について説明をする．ここでは，次のような組合せ最適化問題を考える．

組合せ問題1：

$$\begin{array}{ll} \text{最小化} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{制約条件} & \boldsymbol{a}_k^T \boldsymbol{x} = b_k \quad (k=1, 2, \cdots, m), \\ & \boldsymbol{x} \in \{0, 1\}^n. \end{array}$$

制約 $x_i \in \{0, 1\}$ は，制約 $x_i^2 - x_i = 0$ と同値である．ここで，$\boldsymbol{x}\boldsymbol{x}^T = \boldsymbol{X}$ を満たすような新しい変数 $\boldsymbol{X}$ を導入する．すると，制約 $x_i^2 - x_i = 0$ は，制約 $X_{ii} - x_i = 0$ と書くことができる．これが変数 $(\boldsymbol{x}, \boldsymbol{X})$ に対して線形制約となっていることに注意すると，次のような最適化問題に等価変換できる．

組合せ問題2：

$$\text{最小化} \quad \begin{pmatrix} 0 & \boldsymbol{c}^T \\ \boldsymbol{c} & \boldsymbol{O} \end{pmatrix} \cdot \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix}$$

$$\text{制約条件} \quad \begin{pmatrix} 0 & \boldsymbol{a}_k^T \\ \boldsymbol{a}_k & \boldsymbol{O} \end{pmatrix} \cdot \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} = 2b_k \quad (k=1, 2, \cdots, m),$$

$$\begin{pmatrix} 0 & -\boldsymbol{e}_i^T \\ -\boldsymbol{e}_i & 2\boldsymbol{E}_{ii} \end{pmatrix} \cdot \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} = 0 \quad (i=1, 2, \cdots, n),$$

$$\boldsymbol{x}\boldsymbol{x}^T = \boldsymbol{X}.$$

ただし，$\boldsymbol{e}_i$ は i 番目の要素が1で残りが0となるベクトル，$\boldsymbol{E}_{ii}$ は i 行 i 列目の要素が1で残りが0となる行列である．ここで，$\boldsymbol{x}\boldsymbol{x}^T = \boldsymbol{X}$ という制約を緩和し，$\boldsymbol{x}\boldsymbol{x}^T \preceq \boldsymbol{X}$ という制約に変更する．これは

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \succeq \boldsymbol{O} \tag{3}$$

という半正定値条件と同値である．よって，**組合せ問題2**の3番目の制約 $\boldsymbol{x}\boldsymbol{x}^T = \boldsymbol{X}$ を(3)式に置き換えれば，元問題の緩和問題となるようなSDPが生成できる．

$x_i \in \{0, 1\}$ の代わりに $x_i \in \{-1, 1\}$ という制約で考えることもできる．その場合，同様の手順によって，

2番目の制約を$\begin{pmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \boldsymbol{E}_{ii} \end{pmatrix} \cdot \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} = 1$に変更すればよい．

今回考えた組合せ最適化問題では，線形の目的関数と制約条件から構成されていた．ここで，$\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}+2\boldsymbol{b}^T\boldsymbol{x}+c=\begin{pmatrix} c & \boldsymbol{b}^T \\ \boldsymbol{b} & \boldsymbol{A} \end{pmatrix} \cdot \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix}$となることに注意すると，2次までの多項式からなる問題も，同様の手順でSDP緩和が可能である．3次以上の多項式に対しても，新たな変数を導入すれば，複数の2次以下の多項式にすることができる．よって，これも同様の手順でSDP緩和することは可能である．しかしながら，多項式からなる最適化問題（多項式最適化問題）に対するSDP緩和には，SOS多項式を利用したもっと洗練されたアプローチがある．その詳細については，本特集の一つである脇隼人氏の「多項式最適化問題に対する半正定値計画緩和」で述べられている．

3.4 モデリング言語

現実問題をSDPの標準形に帰着させるのは，かなりの負担になる．もし，MATLABが使える環境であれば，YALMIP[9]というモデリング言語を使用すると，ある程度自動で行ってくれる．例えば，次のような問題を考えよう．変数は$n\times n$の対称行列$\boldsymbol{X}$，$\boldsymbol{Y}$で，定数は$n\times n$の対称行列$\boldsymbol{A}$，$\boldsymbol{B}$と単位行列$\boldsymbol{I}$である．$\boldsymbol{A}\geq\boldsymbol{O}$は$\boldsymbol{A}$の各成分が非負であることを意味する．

$$\begin{array}{ll} \text{最大化} & \mathrm{Tr}(\boldsymbol{Y}) \\ \text{制約条件} & \boldsymbol{X}\succeq\boldsymbol{O},\quad \boldsymbol{Y}\succeq\boldsymbol{O}, \\ & \boldsymbol{X}+\boldsymbol{Y}=\boldsymbol{I},\quad \boldsymbol{X}\geq\boldsymbol{O}, \\ & (\boldsymbol{A}\boldsymbol{X}+\boldsymbol{X}\boldsymbol{A})\preceq\boldsymbol{B}. \end{array}$$

この問題をYALMIPで記述すると，次のようになる．

```
X=sdpvar (n, n) ;
Y=sdpvar (n, n) ;
I=eye (n) ;
F1=set(X>=0)+set(Y>=0) ;
F2=set(X+Y==I)+set(X(:)>=0) ;
F3=set(A*X+X*A<=B) ;
solvesdp(F1+F2+F3, trace(Y)) ;
```

最初の6行で変数や制約条件を定義しており，最後の1行が最適化問題を解くことを意味している．1つ注意しておくと，問題の記述が容易になっても，SDPを解くことが容易になるわけではない．この問題は見掛け以上の難しさを含んでいるため，現状ではnが50から100程度の問題を解くのが限界であろう．

4. SDPに対する解法

SDPの解法として様々な手法が提案されているが，大別すると次の2つに分類することができる．一つはbundle法[3]や低ランク分解法[2][5]に代表される1階の微分情報のみを利用した反復解法であり，もう一つは内点法[4][6][10]や一般拡張ラグランジュ法[8]などの2階の微分情報を利用した反復解法である．組合せ最適化問題のSDP緩和など特殊な構造を持った問題や，非常に粗い精度の解が得られれば十分であるような特殊なケースでは，1階の微分情報を用いた解法が有効であるかもしれない．一般的なSDPをある程度の精度で解く場合には，2階の微分情報を利用した解法の方が安定して最適解を求めることができる．以下では，SDPを解く際に最もよく利用されている，パス追跡タイプの主双対内点法の概略を説明する．

4.1 主双対内点法

SDPの主問題(1)と双対問題(2)には実行可能内点解[8]が存在することを仮定する．このとき双対定理により，SDPの最適解は以下の条件を満たす．

$$\begin{cases} \boldsymbol{X}=\sum_{k=1}^{m}\boldsymbol{F}_k x_k-\boldsymbol{F}_0, \\ \boldsymbol{F}_k\cdot\boldsymbol{Y}=c_k(k=1,2,\cdots,m), \\ \boldsymbol{X}\boldsymbol{Y}=\boldsymbol{O},\quad \boldsymbol{X}\succeq\boldsymbol{O},\ \boldsymbol{Y}\succeq\boldsymbol{O}. \end{cases}$$

ただし，この条件を満たす解$(\boldsymbol{x},\boldsymbol{X},\boldsymbol{Y})$を直接見つけることは難しい．そこで，線形計画問題に対する主双対内点法と同様に，中心パスというものを導入する．中心パス$\mathcal{P}$は以下のように定義される曲線である．

$$\mathcal{P}=\left\{(\boldsymbol{x},\boldsymbol{X},\boldsymbol{Y})\ \middle|\ \begin{array}{l} \boldsymbol{X}=\sum_{k=1}^{m}\boldsymbol{F}_k x_k-\boldsymbol{F}_0, \\ \boldsymbol{F}_k\cdot\boldsymbol{Y}=c_k(k=1,2,\cdots,m), \\ \exists\mu>0,\ \boldsymbol{X}\boldsymbol{Y}=\mu\boldsymbol{I}, \\ \boldsymbol{X}\succeq\boldsymbol{O},\ \boldsymbol{Y}\succeq\boldsymbol{O}. \end{array}\right\}$$

中心パス$\mathcal{P}$はSDPの実行可能解の集合内部にある滑らかな曲線となっており，$\mu\to0$のとき最適解（の一つ）に収束することが知られている．パス追跡タイプの主双対内点法は，中心パス$\mathcal{P}$に沿ってμが0となる方向に進み，最適解に近づいていく反復解法である．アルゴリズムの概略を次に示した（図2も参照）．

主双対内点法

Step 0：初期反復点$(\boldsymbol{x},\boldsymbol{X},\boldsymbol{Y})$と，パラメータ$\mu>0$，定数$0<\sigma<1.0$を選ぶ．ただし，$\boldsymbol{X}\succ\boldsymbol{O}$，$\boldsymbol{Y}\succ\boldsymbol{O}$

[8] 制約条件をすべて満たし，さらに$\boldsymbol{X}$と$\boldsymbol{Y}$が正定値である解．

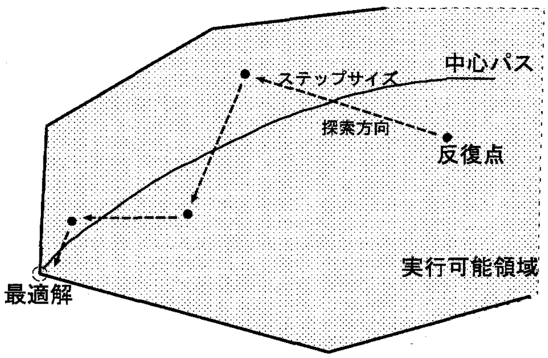

図2　主双対内点法の概略

となるようにする．

Step 1：もし，現在の反復点$(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Y})$が終了条件を満たしていたならば，反復を終了する．

Step 2：μに対応する中心パス上の点に対し，その点に向かう探索方向$(d\boldsymbol{x}, d\boldsymbol{X}, d\boldsymbol{Y})$をNewton法で計算する．その計算途中で，次の演算を行う．

Step 2-1：Schur補完行列$\boldsymbol{B}$を計算する．
$B_{ij} := \boldsymbol{Y}\boldsymbol{F}_i\boldsymbol{X}^{-1}\bullet\boldsymbol{F}_j\ (i, j=1, 2, \cdots, m)$

Step 2-2：線形方程式系$\boldsymbol{B}d\boldsymbol{x}=\boldsymbol{r}$を解く．

Step 3：$\boldsymbol{X}+\alpha d\boldsymbol{X}\succ\boldsymbol{O}$，$\boldsymbol{Y}+\alpha d\boldsymbol{Y}\succ\boldsymbol{O}$になるようなステップサイズ$\alpha$を計算する．

Step 4：反復点$(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Y})$とパラメータ$\mu$を更新し，
$\mu := \sigma\mu$, $(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Y}) := (\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Y})+\alpha(d\boldsymbol{x}, d\boldsymbol{X}, d\boldsymbol{Y})$
Step 1に戻る．

このアルゴリズムにおいて，反復点$(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Y})$は実行可能集合の相対的内点（つまり$\boldsymbol{X}$や$\boldsymbol{Y}$は正定値行列）となっている．これが内点法と呼ばれる理由である．

適切に探索方向$(d\boldsymbol{x}, d\boldsymbol{X}, d\boldsymbol{Y})$，ステップサイズ$\alpha$，定数$\sigma$を選べば，このアルゴリズムは問題の入力サイズの多項式時間内で（近似）最適解を得ることが証明されている．興味がある方は，文献[4][6][10]などを参照してもらいたい．また，多項式時間アルゴリズムであるという理論面だけでなく，実際に数値計算を行うと，現実的な計算資源（計算機・時間）で最適解を求めることができるため，実用性も非常に高い．

参考文献

[1] A. Ben-Tal and A. Nemirovskii, Lectures on Modern Convex Optimization Analysis, Algorithms, and Engineering Applications, *SIAM*, Philadelphia (2001).

[2] S. Burer and R. D. C. Monteiro, A nonlinear programming algorithm for solving semidefinite program via low-rank factorization, *Mathematical Programming*, 95 (2003) 329-357.

[3] C. Helmberg and F. Rendl, A spectral bundle method for semidefinite programming, *SIAM Journal on Optimization*, 10 (2000) 673-696.

[4] C. Helmberg, F. Rendl, R. J. Vanderbei and H. Wolkowicz, An interior-point method for semidefinite programming, *SIAM Journal on Optimization*, 6 (1996) 342-361.

[5] S. Homer and M. Peinado, Design and performance of parallel and distributed approximation algorithms for maxcut, *SIAM Journal on Optimization*, 6 (1997) 48-61.

[6] M. Kojima, S. Shindoh and S. Hara, Interiorpoint methods for the monotone semidefinite linear complementarity problem in symmetric matrices, *SIAM Journal on Optimization*, 7 (1997) 86-125.

[7] 小島正和, 土谷隆, 水野眞治, 矢部博, 内点法, 朝倉書店 (2001).

[8] M. Kŏcvara and M. Stingl, PENNON—a code for convex nonlinear and semidefinite programming, *Optimization Methods and Software*, 8 (2003) 317-333.

[9] J. Löfberg, Dualize it: Software for automatic primal and dual conversions of conic programs, *Optimization Methods and Software*, 24 (2009) 313-325.

[10] R. D. C. Monteiro, Primal-dual path-following algorithms for semidefinite programming, *SIAM Journal on Optimization*, 7 (1997) 663-678.

[11] M. J. Todd, Semidefinite optimization, *Acta Numerica*, 10 (2001) 515-560.

[12] H. Wolkowicz, R. Saigal and L. Vandenberghe, eds., Handbook of Semidefinite Programming, Theory, Algorithms, and Applications, *Kluwer Academic Publishers*, Massachusetts (2000).