線形計画法*

半場 滋†

1 目的

本実験の目的は, プログラムを作成することを通じて, システム工学の重要な一分野である線形計画法に関する理解を深めることである。

2 原理

2.1 線形計画問題

線形計画問題とは, 与えられた線形な等式および不等式制約のもとで, 線形目的関数を最大化あるいは最小化する問題である。まず例を挙げる。

例 1 パソコンショップの店主である A さんは, ある顧客から「PC/AT 互換機のシステムを組んで欲しい」という依頼を受けた。顧客の要望は,

> メモリ 100MB 以上, ディスク 5GB 以上, 予算はディスクとメモリを合わせて 10 万円以下, 予算の範囲でメモリもディスクも可能な限りたくさん載せたい,

というものであった。

話を簡単にするため, (非現実的な仮定だが) メモリおよびディスクの大きさは任意の正の実数値を取るものとする。メモリの価格は 1MB あたり 100 円, ディスクの価格は 1GB あたり 2,500 円とする。また, 顧客の依頼したパソコンは, メモリは最大 800MB まで, ディスクは無限に増設できるものとする。A さんの店では, メモリは 1MB あたり 10 円, ディスクは 1GB あたり 200 円の利益があるものとする。A さんとしては, 顧客の予算を目一杯使って, なるべく多くの利益をあげたい。さて, どのような意思決定をするのが良いだろうか?

メモリの大きさを x MB, ディスクの大きさを y GB としよう。すると, A さんの利益 (目的関数) は,

$$10x + 200y \tag{1}$$

となる。よって, A さんが解くべき問題は, 「$10x + 200y$(目的関数) を最大化すること」となる。

仮にメモリの大きさ x とディスクの大きさ y を勝手に設定できるなら, A さんはいくらでもお金を儲けられることになる。だが実際にはいくつか制約条件がある。それは,

- メモリの大きさは正
- メモリの大きさは 100 MB 以上 (顧客の要望)
- メモリの大きさは 800 MB 以下 (計算機の仕様)
- ディスクの大きさは正
- ディスクの大きさは 5 GB 以上 (顧客の要望)
- メモリとディスクの合計金額は 10 万円以下 (顧客の懐具合)

というものである。これらを書き直すと,

$$\begin{array}{rcl} x & \geq & 0 \\ x & \geq & 100 \\ x & \leq & 800 \\ y & \geq & 0 \\ y & \geq & 5 \\ 100x + 2500y & \leq & 100000 \end{array} \tag{2}$$

となる。

図 1 は, 制約条件を満たす領域の形と目的関数のようすをあらわしたものである。制約条件を満たす領域は図中の網のかかった台形の内部である。目的関数は図中に引かれた傾き -1 の一点鎖線上で一定値を取る。この直線が図中の右上に移動するほど目的関数の値は大きくなる。この図から, 台形の右上の頂点の部分で目的関数が最大値を取ることがわかる。

例 1 は, 結局

> 変数 (x, y), 目的関数 $10x + 200y$ および制約条件 (2) が与えられたとき, 与えられた制約条件のもとで目的関数を最大化するような x, y を求めよ

*この文書は琉球大学工学部電気電子工学科で開講されている 4 年次学生向けの専門実験 (電力工学実験, 電子工学実験, システム工学実験) 向けに執筆されたテキストに若干手を加えたものである。内容を改変しない限り自由に再配布して頂いて構わない。

†琉球大学工学部電気電子工学科 (hanba@eee.u-ryukyu.ac.jp)

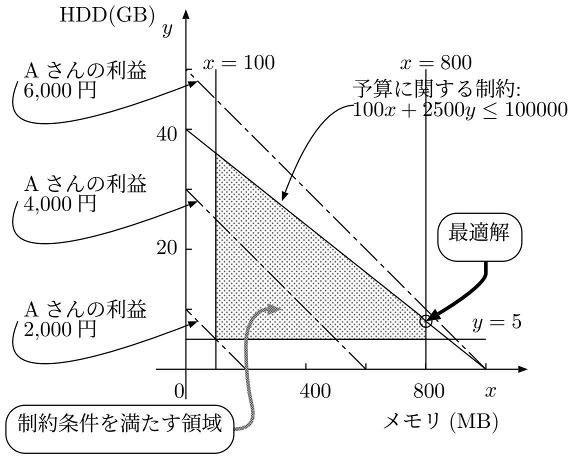


図 1 制約条件を満たす領域と目的関数の挙動

という問題であった。これを一般化したのが線形計画問題である。

線形計画問題 (一般形) n 個の変数 $x_1, \ldots, x_n$ に対し, m 個の等式あるいは不等式の制約条件

$$\begin{array}{rcl} a_{11}x_1 + \ldots + a_{1n}x_n & \geq & b_1 \\ \cdots\cdots & & \\ a_{k1}x_1 + \ldots + a_{kn}x_n & = & b_k \\ \cdots\cdots & & \\ a_{l1}x_1 + \ldots + a_{ln}x_n & \leq & b_l \\ \cdots\cdots & & \end{array} \tag{3}$$

が与えられ, かついくつかの変数に符号に関する制約条件 $x_{i_1} \geq 0, \ldots, x_{i_r} \geq 0$, $x_{j_1} \leq 0, \ldots, x_{j_s} \leq 0$ が課されているものとする。このとき, 制約条件 (3) を満たす範囲内で, 目的関数

$$c_1x_1 + \ldots + c_nx_n \tag{4}$$

を最小化 (あるいは最大化) せよ。

ここに, 最小化 (最大化) とは, 目的関数を最小 (最大) にする x_1 から x_n の値とその最小値 (最大値) を求めることを言う。

線形計画問題の例として, 工場において一定の原料やエネルギー消費量の制約のもとで利益を最大化するための生産計画を立てる問題などが挙げられる。実用的な線形計画問題では, 変数および制約条件の数は極めて大きくなる。

線形計画問題の解法としてよく知られているものに, シンプレックス法と内点法がある。シンプレックス法は 1947 年に G. B. Dantzig によって提案された線形計画問題の解法である。この方法は, 幾何学的には, 凸多面体の形をした制約領域の隣接する頂点をたどりながら解を改善する方法である。シンプレックス法は, 提案されて以降 40 年ほどの間, 最も効率的な線形計画問題の解法とされてきた。これに対し, 1984 年に N. Karmarkar によって提案された内点法は, 理論的な収束特性がシンプレックス法より良いばかりでなく, 極めて大規模な問題に対しては実用上の計算効率もシンプレックス法に勝るため, 提案されて以来, 爆発的に研究が進んできた。(ただし, 規模の小さい問題ではシンプレックス法の方が圧倒的に速い)。

本実験ではシンプレックス法と内点法の双方を取り扱う。

以下ではシンプレックス法と内点法の基本について説明してゆくのだが, 説明がかなり長いので, 読者の負担を減らすために, 読まなくても実験に支障がない部分には印がつけてある。記号 ▶ が欄の左端に記載されているところの記号 ◀ が欄の右端に記載されているところあいだは読み飛ばして構わない。具体的には, ページ 5 からページ 7 までとページ 10 からページ 14 の該当部分である。

2.2 標準形と基準形

線形計画問題の一般形は, 異なる向きの不等号や制約条件が付いた変数と制約条件のない変数が混在しているために, 数学的に取り扱いにくい。そこで, ふつうは, 線形計画問題の一般形を特定の標準的な形に変換してから解く。

標準的な形としてよく用いられるのは, 標準形と呼ばれるものと, 基準形と呼ばれるものである。以下でこれらについて説明する。

以下では, 特に断らない限り, $\boldsymbol{A}$ は m 行 n 列の実行列, $\boldsymbol{b}$ は m 次元のベクトル, $\boldsymbol{c}$, $\boldsymbol{x}$ は n 次元のベクトルとする。また, $\boldsymbol{A}$ が m 行 n 列の実行列であることを $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ という記法であらわし, $\boldsymbol{x}$ が n 次元の実ベクトルであるということを $\boldsymbol{x} \in \mathbb{R}^n$ という記法であらわす[i]。さらに, 行列 $\boldsymbol{A}$ の第 i 番目の列ベクトルを $\boldsymbol{a}_i$ などと書き, ベクトル $\boldsymbol{b}$, $\boldsymbol{c}$ などの第 i 要素をそれぞれ b_i, c_i などと書く。また, $\boldsymbol{x}$ と $\boldsymbol{w}$ がともに n 次のベクトルであるとき, $\boldsymbol{x} \geq \boldsymbol{w}$ という記法によって, ベクトル $\boldsymbol{x}$ の各成分がベクトル $\boldsymbol{w}$ の対応する成分以上である, すなわち $x_i \geq w_i$ $(i = 1, \ldots, n)$ となっているということをあらわす。零ベクトルあるいは零行列をあらわすのには記号 $\boldsymbol{0}$ を使う。次元を明示する必要がある

[i] 記号 $\mathbb{R}$ は実数体をあらわす。

ときには $\mathbf{0}_n$, $\mathbf{0}_{m\times n}$ などのように書く[ii)]。

線形計画問題の標準形とは, 以下のような形式の問題である。

$$\boxed{\textbf{標準形:}\quad \min\{z = \boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \mathbf{0}\}}$$

この式の読み方は,「制約条件 $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{x} \geq \mathbf{0}$ のもとで目的関数 $z = \boldsymbol{c}^T\boldsymbol{x}$ を最小化せよ」である。なお, min は minimize(最小化する) の略である。

続いて, 線形計画問題の一般形を標準形に変換する手順について説明する。このためには, 以下に述べるような一連の手順を実行すればよい。

1. 不等式制約 $a_{i,1}x_1 + \cdots + a_{i,n}x_n \geq b_i$: 新たな変数 $y_i \in \mathbb{R}$ を導入し, 制約条件を $a_{i,1}x_1 + \cdots + a_{i,n}x_n - y_i \geq b_i, y_i \geq 0$ と書き直す。目的関数は変更しない。

2. 等式制約 $a_{i,1}x_1 + \cdots + a_{i,n}x_n = b_i$: 変更しない。

3. 不等式制約 $a_{i,1}x_1 + \cdots + a_{i,n}x_n \leq b_i$: 新たな変数 $y_i \in \mathbb{R}$ を導入し, 制約条件を $a_{i,1}x_1 + \cdots + a_{i,n}x_n + y_i \geq b_i, y_i \geq 0$ と書き直す。目的関数は変更しない。

4. 符号制約なしの変数 x_i: 新たに 2 個の正の変数 x_{i-}, x_{i+} を導入し, $x_i = x_{i+} - x_{i-}$, $x_{i+} \geq 0$, $x_{i-} \geq 0$ と書き直す。このとき, 各制約条件にあらわれる項 $a_{p,i}x_i$ は $a_{p,i}x_{i+} - a_{p,i}x_{i-}$ で置き換えられる。また, 目的関数にあらわれる c_ix_i という項も $c_ix_{i+} - c_ix_{i-}$ で置き換えられる。

5. 負でない変数 $x_i \geq 0$: 変更しない。

6. 正でない変数 $x_i \leq 0$: $x_i' = -x_i$ と変数を置き直す。このとき, 制約条件にあらわれる項 $a_{p,i}x_i$ は $-a_{p,i}x_i'$ で置き換えられる。また, 目的関数にあらわれる c_ix_i という項も $-c_ix_i'$ で置き換えられる。

7. $\boldsymbol{c}^T\boldsymbol{x}$ を最大化する問題 (最大化問題): $\boldsymbol{c}' = -\boldsymbol{c}$ と定義して, 新たな目的関数 $(\boldsymbol{c}')^T\boldsymbol{x}$ を最小化する。

上記の手続きのうち, 1, 3, 4 では変数の数がそれぞれ 1 個ずつ増えることを注意しておく。

これに対し, 線形計画問題の基準形とは, 次のような問題である。

$$\boxed{\textbf{基準形:}\quad \min\{z = \boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}, \boldsymbol{x} \geq \mathbf{0}\}}$$

ii) 零ベクトルと零行列に同じ記号が用いられるので, 混乱しないよう注意すること。

標準形の場合と同様の手順によって, 線形計画問題の一般形を基準形に変換することができる。この場合は, 制約条件や変数の数は増加する。基準形への変換が標準形への変換と異なる点は, 上記とは異なる方法を使うと, 等式制約条件や制約条件なしの変数を除去する際に, 制約条件や変数の数を減らすこともできる, ということである。この方法はやや繁雑なのでここでは述べない。興味のある読者は文献 [5] を参照すること。

2.3 シンプレックス法

本節では, 線形計画問題を解くためのシンプレックス法というアルゴリズムについて簡単に説明する。この部分の説明はおもに文献 [3],[4] による。

標準形で記述された線形計画問題:

$$\text{(STD)}\quad \min\left\{z = \boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \mathbf{0}\right\} \tag{5}$$

が与えられているものとする。先に注意したように, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ である。また, $n \geq m$ で, 行列 $\boldsymbol{A}$ はフルランクであると仮定しておく。また, 説明の便宜上, (STD) の制約条件を満たす領域は空でないと仮定する。

シンプレックス法は, まず制約条件を満たす (最適とは限らない) 基底解と呼ばれる解をひとつ見付け, 続いて最適解が得られるまで基底解を改善していく方法である。標準形の問題が一定の条件を満たすときは, シンプレックス法を用いると有限回の計算で最適解が得られることが保証される。

ではシンプレックス法の説明に入ろう。

行列 $\boldsymbol{A}$ の列ベクトル $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ の中から適当な m 個の線形独立なベクトルを選び出し, これらを並べて

$$\boldsymbol{B} = [\boldsymbol{a}_{i_1}, \ldots, \boldsymbol{a}_{i_m}] \tag{6}$$

という行列を作る。また, 行列 $\boldsymbol{A}$ の列ベクトルから行列 $\boldsymbol{B}$ の列ベクトルを除いた残りを並べて,

$$\boldsymbol{N} = [\boldsymbol{a}_{j_1}, \ldots, \boldsymbol{a}_{j_{n-m}}] \tag{7}$$

という行列を作る。行列 $\boldsymbol{B}$, $\boldsymbol{N}$ はそれぞれ基底行列, 非基底行列と呼ばれる。さらに, 行列 $\boldsymbol{B}$ に対応する変数の組 $x_{i_1}, \ldots, x_{i_m}$ を集めて

$$\boldsymbol{x}_B = [x_{i_1}, \ldots, x_{i_m}]^T \tag{8}$$

なるベクトルを作る。$\boldsymbol{x}_B$ を基底変数ベクトルと呼ぶ。続いて, 行列 $\boldsymbol{N}$ に対応する変数の組 $x_{j_1}, \ldots, x_{j_{n-m}}$ を集めて

$$\boldsymbol{x}_N = [x_{j_1}, \ldots, x_{j_{n-m}}]^T \tag{9}$$

なるベクトルを作る。$\boldsymbol{x}_N$ を非基底変数ベクトルと呼ぶ。

ここで, 目的関数に対応するベクトル $\boldsymbol{c}$ を基底行列に対応した形に分割し,

$$\boldsymbol{c}_B = [c_{i_1}, \ldots, c_{i_m}], \quad \boldsymbol{c}_N = [c_{j_1}, \ldots, c_{j_{n-m}}] \tag{10}$$

とおくと, 目的関数 $z = \boldsymbol{c}^T\boldsymbol{x}$ は

$$z = \boldsymbol{c}_B^T\boldsymbol{x}_B + \boldsymbol{c}_N^T\boldsymbol{x}_N \tag{11}$$

と書き直される。

さて, 変数ベクトル $\boldsymbol{x}$ を

$$\boldsymbol{x}_B = \boldsymbol{B}^{-1}\boldsymbol{b}, \quad \boldsymbol{x}_N = \boldsymbol{0} \tag{12}$$

となるように選んでみよう。等式制約条件 $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ は

$$\boldsymbol{B}\boldsymbol{x}_B + \boldsymbol{N}\boldsymbol{x}_N = \boldsymbol{b} \tag{13}$$

と書き直せるから, (12) から決まる $\boldsymbol{x}$ は制約条件 $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ を満たしている。このようにして (12) から定められた解 $\boldsymbol{x}$ を基底解と呼ぶ。ただし, この段階では, 符号制約 $\boldsymbol{x} \geq \boldsymbol{0}$ が満たされているかどうかは明らかでない。

ところで, $\boldsymbol{B}^{-1}\boldsymbol{b} \geq \boldsymbol{0}$ という不等式が成り立つ場合には, $\boldsymbol{x}_B \geq 0$, $\boldsymbol{x}_N \geq 0$ より, 基底解 $\boldsymbol{x}$ は符号制約 $\boldsymbol{x} \geq \boldsymbol{0}$ を満たす。よって, この場合には, 基底解 $\boldsymbol{x}$ は問題 (STD) のひとつの解になっていることがわかる。(12) から定まる基底解 $\boldsymbol{x}$ が問題 (STD) の解となるとき, この解を実行可能基底解と呼び, 対応する行列 $\boldsymbol{B}$ を実行可能基底行列と呼ぶ[iii)] 。

以下では, まず実行可能基底行列がひとつ得られている場合に最適解を構成するためのアルゴリズムについて説明し, 続いて実行可能基底行列が未知の場合への拡張について述べる。

2.3.1 実行可能基底行列が得られている場合

実行可能基底行列 $\boldsymbol{B}$ がすでに得られ, 制約条件 $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ が (13) の形に書き直されているものとする。ここで,

$$\bar{\boldsymbol{b}} = \boldsymbol{B}^{-1}\boldsymbol{b}, \quad \bar{\boldsymbol{A}}_N = \boldsymbol{B}^{-1}\boldsymbol{N} \tag{14}$$

と定義し, (13) を

$$\boldsymbol{x}_B = \bar{\boldsymbol{b}} - \bar{\boldsymbol{A}}_N\boldsymbol{x}_N \tag{15}$$

のように書き直す。(15) を (11) に代入し,

$$z_B = \boldsymbol{c}_B^T\bar{\boldsymbol{b}}, \quad \bar{\boldsymbol{c}}_N^T = \boldsymbol{c}_N^T - \boldsymbol{c}_B^T\bar{\boldsymbol{A}}_N \tag{16}$$

とおくと,

$$z = z_B + \bar{\boldsymbol{c}}_N^T\boldsymbol{x}_N \tag{17}$$

なる表現が得られる。

ここで, 行列 $\bar{\boldsymbol{A}}_N$ の第 i 列ベクトルを $\bar{\boldsymbol{a}}_i$ と書くことにする。すなわち,

$$\bar{\boldsymbol{A}}_N = [\bar{\boldsymbol{a}}_1, \ldots, \bar{\boldsymbol{a}}_{n-m}] \tag{18}$$

と書く。(15) と (17) をまとめると, 問題 (SP) と等価な

$$\text{(BF)} \quad \min\{z = z_B + \bar{\boldsymbol{c}}_N^T\boldsymbol{x}_N : \\ \boldsymbol{x}_B = \bar{\boldsymbol{b}} - \bar{\boldsymbol{A}}_N\boldsymbol{x}_N, \boldsymbol{x}_B \geq \boldsymbol{0}, \boldsymbol{x}_N \geq \boldsymbol{0}\} \tag{19}$$

という問題が得られる。問題 (BF) を実行可能基底行列 $\boldsymbol{B}$ に対する基底形式表現と呼ぶ。

さて, 基底解 (12) は (BF) の制約条件を満たしているので, 最適とは限らない解である。そこで, 基底解 (12) を変更して目的関数 z をより小さくすることを考える。このために, (15) と (17) を見比べると, 以下のことがわかる。

- (16) のベクトル $\bar{\boldsymbol{c}}_N$ の要素の中から値が最小のものを探す。ここで, 添字 s を持つ要素 $\bar{c}_{N,s}$ が最小値であったものとしよう。$\bar{c}_{N,s} \geq 0$ であれば, 非基底変数をどのように変更しても目的関数は減少しないから, 現在の z がすでに最小値である。逆に $\bar{c}_{N,s} < 0$ のときには, (17) で $\bar{c}_{N,s}$ に対応する非基底変数 x_{j_s} の値を零から増加させれば z は減少するから, 現在の解を改善できる可能性がある。

- 非基底変数 x_{j_s} は (19) の制約条件を崩さない範囲で選ばれなければならない。x_{j_s} の値を零から θ に変更したとき, $\boldsymbol{x}_B = \bar{\boldsymbol{b}} - \bar{\boldsymbol{a}}_s\theta$ となる。ベクトル $\bar{\boldsymbol{b}}$ の各成分を $\bar{b}_1, \ldots, \bar{b}_m$, ベクトル $\bar{\boldsymbol{a}}_s$ の各成分を $\bar{a}_{s;1}, \ldots, \bar{a}_{s;m}$ と書く。$\boldsymbol{x}_B \geq \boldsymbol{0}$ という制約から, $\bar{a}_{s,i}$ が正のときには, θ を $\bar{b}_i/\bar{a}_{s;i}$ 以下に取る必要がある。これに対し, $\bar{a}_{s,i} = 0$ のときは, θ をいくら大きく取っても制約条件はつねに満たされる。よって, 添字集合 $\{1, \ldots, m\}$ から $\bar{a}_{s;i} > 0$ となるものを抜き出し,

  $$\bar{\theta} = \min\{\bar{b}_i/\bar{a}_{s;i} : \bar{a}_{s;i} > 0, 1 \leq i \leq m\}$$

  とすると, この $\bar{\theta}$ が θ の取りうる上限となる。一方, $a_s <= 0$ のときには, θ をいくらでも大きく取ることができるため, z の値をいくらでも小さくすることができる。

[iii)] ここで言う「解」とは制約条件をすべて満たしているもののことである。与えられた最適化問題の解 (制約条件をすべて満たし, かつ目的関数を最小化するもの) とは異なるので, 混乱しないよう注意すること。

以上のような考え方からシンプレックス法のアルゴリズムが得られる[iv]。

シンプレックス法

(初期化) 実行可能基底行列 $\boldsymbol{B}$ が与えられているものとする。$\boldsymbol{B}$ を使って基底形式表現 (19) を求める。

(ステップ 1) $\{\bar{c}_{N,1},\ldots,\bar{c}_{N,n-m}\}$ の中で最小のものを $\bar{c}_{N,s}$ とする。条件を満たすものが複数あるときには, どれを選んでもよい。$\bar{c}_{N,s}\geq 0$ であれば終了。この場合には,

$$\boldsymbol{x}_B^*=\boldsymbol{B}^{-1}\boldsymbol{b},\quad \boldsymbol{x}_N^*=\boldsymbol{0} \tag{20}$$

が最適解である。$\bar{c}_{N,s}<0$ であればステップ 2 に進む。

(ステップ 2) 非基底行列 $\boldsymbol{N}$ の第 s 列 $\boldsymbol{a}_{j_s}$ を取り, 線形方程式

$$\boldsymbol{B}\bar{\boldsymbol{a}}_s=\boldsymbol{a}_{j_s} \tag{21}$$

を解いてベクトル $\bar{\boldsymbol{a}}_s$ を求める。$\bar{\boldsymbol{a}}_s\leq\boldsymbol{0}$ なら終了 (無限解が存在する)。そうでないときは, $\bar{\theta}=\min\{\bar{b}_i/\bar{a}_{s;i}:\bar{a}_{s;i}>0,1\leq i\leq m\}$ とする。ある r に対して $\bar{\theta}=\bar{b}_r/\bar{a}_{s;r}$ となっているので, このような添字 r を選ぶ (条件を満たす添字が複数ある場合は, どれを選んでもよい)。

(ステップ 3) ステップ 1 とステップ 2 で得られた添字 r と s を用いて基底行列を

$$\boldsymbol{B}=[\boldsymbol{a}_{i_1},\ldots,\boldsymbol{a}_{i_{r-1}},\boldsymbol{a}_{i_r},\boldsymbol{a}_{i_{r+1}},\ldots,\boldsymbol{a}_{i_m}] \tag{22}$$

から

$$\boldsymbol{B}'=[\boldsymbol{a}_{i_1},\ldots,\boldsymbol{a}_{i_{r-1}},\boldsymbol{a}_{j_s},\boldsymbol{a}_{i_{r+1}},\ldots,\boldsymbol{a}_{i_m}] \tag{23}$$

に変更し, 新しい基底変数と非基底変数から対応する $\bar{\boldsymbol{c}}_N$ を求めてステップ 1 に戻る。

なお, (14) の第 1 式の $\bar{\boldsymbol{b}}$ を求めるときには, 逆行列は使わず, 線形方程式

$$\boldsymbol{B}\bar{\boldsymbol{b}}=\boldsymbol{b} \tag{24}$$

を解く。一般に, 逆行列の数値計算は計算量が多くかつ誤差の影響を受けやすいので, 数値解析の分野では逆行列の計算は可能な限り回避される。

また, 各ステップでベクトル $\bar{\boldsymbol{c}}_N$ を求めるためには $\boldsymbol{c}_B^T\boldsymbol{B}^{-1}\boldsymbol{N}$ を計算する必要があるが, これを求めるためには

- 線形方程式 $\boldsymbol{B}^T\boldsymbol{c}_{N0}=\boldsymbol{c}_B$ を解いてベクトル $\boldsymbol{c}_{N0}$ を求める
- $\boldsymbol{c}_{N0}$ に行列 $\boldsymbol{N}$ の転置を左から掛ける

という手順で計算をおこなう。

$$(\boldsymbol{c}_B^T\boldsymbol{B}^{-1}\boldsymbol{N})^T=\boldsymbol{N}^T(\boldsymbol{B}^T)^{-1}\boldsymbol{c}_B$$

だから, このようにしても正しい結果が得られることがわかる。$\bar{\boldsymbol{A}}_N=\boldsymbol{B}^{-1}\boldsymbol{N}$ のすべての成分を計算しても正しい結果は得られるが, これは計算量の観点からは無駄である。

2.3.2　もとの問題の最適解の抽出

シンプレックス法終了時に得られている実行可能基底行列を $\boldsymbol{B}^*$ とすると, 線形方程式

$$\boldsymbol{B}^*\boldsymbol{x}_B^*=\boldsymbol{b} \tag{25}$$

を解くことにより最適解の一部 $\boldsymbol{x}_B^*$ が得られる。残りの変数 $\boldsymbol{x}_N^*$ は零である。$\boldsymbol{x}_B^*$ と $\boldsymbol{x}_N^*$ からもとの問題の最適解 $\boldsymbol{x}^*$ を構成するためには, どの変数が基底変数に入っていたかを確認する必要がある。

たとえば, 基底変数が x_1, x_3, x_5 であり, $\boldsymbol{x}_B^*=[10,20,30]$ となっている場合には, もとの問題の最適解は

$$[10,0,20,0,30,0,...]$$

である (非基底変数はすべて零である)。
▶

例 2 シンプレックス法の挙動を調べるため, 以下のような 2 変数の簡単な線形計画問題を考える。

$$\begin{aligned}&\min\{x_1+x_2:\\&\qquad 1\leq x_1\leq 2,1\leq x_2\leq 2,x_1\geq 0,x_2\geq 0\}\end{aligned} \tag{26}$$

問題 (26) の制約条件を満たす領域を図 2 に示す。

目的関数は x_1+x_2 であるが, これは傾き -1 の直線上で一定値となる。そして, この直線が左下にずれるほど目的関数の値は小さくなる。よって, この問題の最適解は $x_1=1,x_2=1$ であることがわかる。

[iv] ここで挙げたアルゴリズムは「改訂シンプレックス法」と呼ばれるものである。ところで, 「改訂」シンプレックス法が存在する以上, ただのシンプレックス法も存在する。シンプレックス法と改訂シンプレックスの違いは, シンプレックス法が基底形式表現そのものを繰り返しの各ステップで直接行列の基本変形によって更新してゆくのに対し, 改訂シンプレックス法は各ステップで基底行列 (に対応する添字) を更新してから対応する線形方程式を解く, という点にある。高速な線形方程式のソルバが利用可能である場合には, ふつうは改訂シンプレックス法の方が速い。

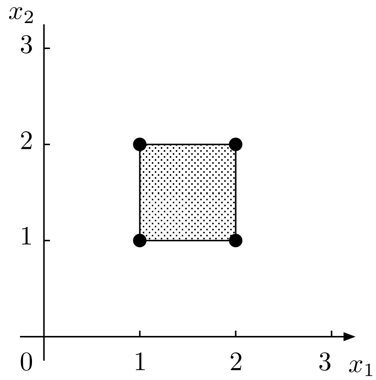


図 2 (26) の制約条件を満たす領域

では, この問題を標準形に書き直そう。不等式 $1 \leq x_1$ は, 新しい変数 x_3 を導入して,

$$x_1 - x_3 = 1, x_3 \geq 0$$

と書ける。不等式 $x_1 \leq 2$ は, 新しい変数 x_4 を導入することで,

$$x_1 + x_4 = 2, x_4 \geq 0$$

のようになる。x_2 に関する不等式も同様にして

$$x_2 - x_5 = 1, x_5 \geq 0$$
$$x_2 + x_6 = 2, x_6 \geq 0$$

という等式に変わる。さて,

$$\boldsymbol{x} = [x_1, x_2, x_3, x_4, x_5, x_6]^T$$

とおくと, この問題は

$$\min\{\boldsymbol{c}^T\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\},$$

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} 1 \\ 2 \\ 1 \\ 2 \end{bmatrix}, \boldsymbol{c} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{27}$$

という線形計画問題に書き直される。これは標準形である。

次に, (27) の実行可能基底行列と実行可能基底解を求めてみよう。基底変数を決めるときには 1 から 6 までの添字の中から 4 個の添字を選ぶことになるので, 可能な組み合わせの数は $\binom{6}{4} = 15$ 通りである[v]。一方, 実行可能基底解は図 2 の制約条件を満たす領域 (正方形) の頂点であり, 頂点の数は 4 個しかないから, 上記の 15 種類の添字の選び方のうち実行可能なものは 4 個しかないと推察される。この推察は正しく, 実際に計算してみると, 特定の 4 通りの添字の選び方に対してのみ実行可能基底解が得られ, それ以外の選び方では, $\boldsymbol{B}$ が正則でなくなるか, $\boldsymbol{B}^{-1}\boldsymbol{b}$ の成分の中に負のものがあらわれるということがわかる。表 1 に実行可能な基底変数ベクトルと, 実行可能基底解に対応する x_1 と x_2 の値を示す。表 1 の右の欄の値から, これらが図 2 の 4 個の頂点に対応していることがわかる。

表 1 実行可能な基底変数ベクトルと実行可能基底解

$\boldsymbol{x}_B$	(x_1, x_2) の値
$[x_1, x_2, x_3, x_5]^T$	(2, 2)
$[x_1, x_2, x_3, x_6]^T$	(2, 1)
$[x_1, x_2, x_4, x_5]^T$	(1, 2)
$[x_1, x_2, x_5, x_6]^T$	(1, 1)

続いて, 基底変数ベクトルを $\boldsymbol{x}_B = [x_1, x_2, x_3, x_5]^T$ とした場合のシンプレックス法の動作を見よう。シンプレックス法は制約条件を満たす領域の頂点をたどって最適解を探す方法なのだから, 点 $(x_1, x_2) = (2, 2)$ から出発した場合, 2 回の繰り返し計算で最適点 $(x_1, x_2) = (1, 1)$ に到達できるはずである。

ループ第 1 回目 $\boldsymbol{x}_B = [x_1, x_2, x_3, x_5]^T$, $\boldsymbol{x}_N = [x_4, x_6]^T$ から出発する (点 $(x_1, x_2) = (2, 2)$ から始めることに対応)。これに対応する基底行列は

$$\boldsymbol{B} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3, \boldsymbol{a}_5] = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

であり, 非基底行列は

$$\boldsymbol{N} = [\boldsymbol{a}_4, \boldsymbol{a}_6] = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

である。また,

$$\boldsymbol{c}_B = [1, 1, 0, 0]^T, \quad \boldsymbol{c}_N = [0, 0]^T$$

となる。ここから, $z_B = 4$,

$$\bar{\boldsymbol{b}} = \begin{bmatrix} 2 \\ 2 \\ 1 \\ 1 \end{bmatrix}, \bar{\boldsymbol{A}}_N = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \bar{\boldsymbol{c}}_N = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

v) $\binom{n}{r} = \frac{n!}{r!(n-r)!}$ である。

が得られる。

(ステップ 1-1) ベクトル $\boldsymbol{c}_N$ の第 1 要素, 第 2 要素はともに -1 で負なので, このうち番号が若い $s=1$ を選択して次に進む。

(ステップ 1-2) $\bar{\boldsymbol{a}}_1 = [1,0,1,0]^T$ で, このうち第 1 成分と第 3 成分が正である。$\bar{b}_1/\bar{a}_{1;1} = 2$ $\bar{b}_3/\bar{a}_{1;3} = 1$ であり, $\bar{b}_3/\bar{a}_{1;3}$ の方が小さいので, 添字 $r=3$ を選択して次に進む。

(ステップ 1-3) 基底行列は

$$\boldsymbol{B} = [\boldsymbol{a}_{i_1}, \boldsymbol{a}_{i_2}, \overset{\Downarrow}{\boldsymbol{a}}_{i_3}, \boldsymbol{a}_{i_4}]$$

から

$$\boldsymbol{B}' = [\boldsymbol{a}_{i_1}, \boldsymbol{a}_{i_2}, \overset{\Downarrow}{\boldsymbol{a}}_{j_1}, \boldsymbol{a}_{i_4}]$$

に変わる。ここで $\boldsymbol{x}_B = [x_1,x_2,x_3,x_5]^T$, $\boldsymbol{x}_B = [x_4,x_6]^T$ であったことを思い出すと, $i_1=1$, $i_2=2$, $i_3=3$, $i_4=5$, $j_1=4$, $j_2=6$ であることがわかる。ゆえに, 新しい基底行列は

$$\boldsymbol{B}' = [\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_4, \boldsymbol{a}_5]$$

となる。対応する基底変数ベクトルは $\boldsymbol{x}_B = [x_1,x_2,x_4,x_5]^T$ である。

ループ第 2 回目 ループ第 1 回目の終了時の状態を継承して, $\boldsymbol{x}_B = [x_1,x_2,x_4,x_5]^T$, $\boldsymbol{x}_N = [x_3,x_6]^T$ から出発する。表 1 から, これが点 $(x_1,x_2)=(1,2)$ に対応することがわかる。先と同様の計算により, $z_B = 3$,

$$\bar{\boldsymbol{b}} = \begin{bmatrix}1\\2\\1\\1\end{bmatrix}, \bar{\boldsymbol{A}}_N = \begin{bmatrix}-1 & 0\\0 & 1\\1 & 0\\0 & 1\end{bmatrix}, \bar{\boldsymbol{c}}_N = \begin{bmatrix}1\\-1\end{bmatrix}$$

が得られる。

(ステップ 2-1) ベクトル $\boldsymbol{c}_N$ の第 2 要素のみ負なので, $s=2$ を選択して次に進む。

(ステップ 2-2) $\bar{\boldsymbol{a}}_1 = [0,1,0,1]^T$ で, このうち第 2 成分と第 4 成分が正である。$\bar{b}_2/\bar{a}_{2;2} = 2$, $\bar{b}_4/\bar{a}_{2;2} = 1$ であり, $\bar{b}_4/\bar{a}_{2;4}$ の方が小さいので, 添字 $r=4$ を選択して次に進む。

(ステップ 2-3) 基底行列は

$$\boldsymbol{B} = [\boldsymbol{a}_{i_1}, \boldsymbol{a}_{i_2}, \boldsymbol{a}_{i_3}, \overset{\Downarrow}{\boldsymbol{a}}_{i_4}]$$

から

$$\boldsymbol{B}' = [\boldsymbol{a}_{i_1}, \boldsymbol{a}_{i_2}, \boldsymbol{a}_{i_3}, \overset{\Downarrow}{\boldsymbol{a}}_{j_2}]$$

に変わる。ここで $\boldsymbol{x}_B = [x_1,x_2,x_4,x_5]^T$, $\boldsymbol{x}_B = [x_3,x_6]^T$ であったことを思い出すと, $i_1=1$, $i_2=2$, $i_3=4$, $i_4=5$, $j_1=3$, $j_2=6$ であることがわかる。ゆえに, 新しい基底行列は

$$\boldsymbol{B}' = [\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_4, \boldsymbol{a}_6]$$

となる。対応する基底変数ベクトルは $\boldsymbol{x}_B = [x_1,x_2,x_4,x_6]^T$ である。

ループ第 3 回目 ループ第 2 回目の終了時の状態を継承して, $\boldsymbol{x}_B = [x_1,x_2,x_4,x_6]^T$, $\boldsymbol{x}_N = [x_3,x_5]^T$ から出発する。表 1 から, これが点 $(x_1,x_2)=(1,1)$ に対応することがわかる。この点で目的関数は最小となるから, アルゴリズムはこの段階で終了するはずである。これを確認しよう。先と同様の計算により, $z_B = 2$,

$$\bar{\boldsymbol{b}} = \begin{bmatrix}1\\1\\1\\1\end{bmatrix}, \bar{\boldsymbol{A}}_N = \begin{bmatrix}-1 & 0\\0 & -1\\1 & 0\\0 & 1\end{bmatrix}, \bar{\boldsymbol{c}}_N = \begin{bmatrix}0\\0\end{bmatrix} \tag{28}$$

が得られる。

(ステップ 3-1) ベクトル $\boldsymbol{c}_N$ のすべての要素は零である。よって, すでに最適解が得られているので, 計算を終了する。

もとの問題の解の復元 計算終了時の基底変数ベクトルが $\boldsymbol{x}_B = [x_1,x_2,x_4,x_6]^T$ であったことと最適解が $\boldsymbol{x}_B = \bar{\boldsymbol{b}}$, $\boldsymbol{x}_N = \boldsymbol{0}$ とあらわされることを思い出すと, (28) から, $x_1=1$, $x_2=1$, $x_3=0$, $x_4=1$, $x_5=0$, $x_6=1$ が最適解となる。よって, $x_1=1$, $x_2=1$ が目的関数を最小とする点であることがわかる。

以上の説明から, シンプレックス法のアルゴリズムが, 点 $(x_1,x_2)=(2,2)$ から出発し, $(x_1,x_2)=(1,2)$ という頂点を経由して最適点 $(x_1,x_2)=(1,1)$ に到達していることがわかる。

◀

2.3.3 ビッグ M 法

第 2.3.1 節で述べたアルゴリズムを利用して与えられた線形計画問題を解くためには実行可能基底行列 $\boldsymbol{B}$ が必要であった。ところで, 問題の規模が大きいときには実行可能基底行列をひとつ求めること自体も容易でないことが多い。そのような状況に対処するためには, 第 2.3.1 節のアルゴリズムを実行可能基底行列が未知の場合にも適用できるように拡張しておく必要がある。

このような拡張としてよく知られているものに, 2段階法とビッグM法がある。本節では, これらのうちビッグM法について述べる。

(STD) にビッグ M 法が適用可能であるためには (STD) の制約条件の右辺にあらわれるベクトル $\boldsymbol{b}$ の各成分が非負でなければならない。そこで, $\boldsymbol{b}$ の第 i 成分 b_i が負で, 対応する行列 $\boldsymbol{A}$ の行ベクトルが $\boldsymbol{\alpha}_i^T$ であるときには, b_i を $-b_i$ で置き換え, $\boldsymbol{\alpha}_i^T$ を $-\boldsymbol{\alpha}_i^T$ で置き換える。これにより, もとの制約条件と等価で $\boldsymbol{b} \geq \boldsymbol{0}$ を満たす制約条件が得られる。

次に, 人為変数と呼ばれる新たな変数

$$\boldsymbol{\xi} = [\xi_1, \ldots, \xi_m]^T \tag{29}$$

を導入し,

$$\boldsymbol{c}_2 = [\boldsymbol{c}^T, M \cdot \boldsymbol{1}_m^T]^T, \quad \boldsymbol{A}_2 = [\boldsymbol{A}, \boldsymbol{I}_m] \tag{30}$$

と定義して (ただし $\boldsymbol{1}_m$ は m 次の要素がすべて 1 のベクトル, M は十分大きい正の定数, $\boldsymbol{I}_m$ は m 次の単位行列), (5) から次のような問題を作る。

$$\min \left\{ \boldsymbol{c}_2^T \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\xi} \end{bmatrix} : \boldsymbol{A}_2 \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\xi} \end{bmatrix} = \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}, \boldsymbol{\xi} \geq \boldsymbol{0} \right\} \tag{31}$$

もとの問題は $\boldsymbol{b} \geq \boldsymbol{0}$ となるように書き換えられていたから, $\boldsymbol{x} = \boldsymbol{0}, \boldsymbol{\xi} = \boldsymbol{b}$ は (31) の制約条件を満たす。よって, これは実行可能基底解である。ゆえに, 問題 (31) において基底変数ベクトルとして $\boldsymbol{\xi}$ を取ると, 実行可能基底行列が得られるから, ここから出発してシンプレックス法を実行し, 最適解を求めることが可能である。

シンプレックス法の終了条件が満たされ, 繰り返し計算が終了したものとしよう。このとき問題となるのは, もとの問題の最適解をどのようにして復元するか, ということである。

パラメータ M が十分大きく, かつ計算終了時に有限の最小値が得られた場合を考える。この最小値に対応する $\boldsymbol{x}$ と $\boldsymbol{\xi}$ をそれぞれ $\boldsymbol{x}^*$ と $\boldsymbol{\xi}^*$ とする。このとき, $\boldsymbol{\xi}^* = \boldsymbol{0}$ であれば $\boldsymbol{x} = \boldsymbol{x}^*$ が問題 (5) の最適解であることが言え, $\boldsymbol{\xi}^* \neq \boldsymbol{0}$ であれば問題 (5) は解を持たないということが言える。証明は繁雑なので略す。ただし, パラメータ M をどの程度取れば十分であるかについての基準はない。よって, 計算終了時に $\boldsymbol{\xi}^* \neq \boldsymbol{0}$ となったときには, M を大きく取り直してシンプレックス法を再度実行する必要がある。

なお, 計算終了時の結果からもとの問題の最適解 $\boldsymbol{x}^*$ を構成する手順は第 2.3.2 節と同一である。

2.4 内点法

本節では内点法について述べる。

内点法は大きく分けると主内点法と主-双対法の 2 種類に分類されるのだが, 本節では主-双対法と呼ばれるアルゴリズムのうち比較的簡単なもの 2 種類を文献 [5, 6] にしたがって紹介する。

まず, いくつか記号の定義をしておく。ベクトルのノルムは, $\|\boldsymbol{x}\| = \sqrt{\sum_{i=1}^n x_i^2}$ によって定義される。ベクトル同士の Hadamard 積を, 以下によって定義する。

$$\boldsymbol{x}\boldsymbol{s} = [x_1 s_1, \ldots, x_n s_n]^T$$

また, 記法の簡単のため, ベクトル $\boldsymbol{x}$ に対し,

$$\begin{aligned} 1/\boldsymbol{x} &= [1/x_1, \ldots, 1/x_n]^T, \\ \boldsymbol{x}^2 &= [x_1^2, \ldots, x_n^2]^T \end{aligned}$$

などと書く。

2.4.1 内点法のアルゴリズム

基準形で記述された最適化問題

$$\text{(P)} \quad \min \left\{ z = \boldsymbol{c}^T \boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0} \right\} \tag{32}$$

を考える。(32) を主問題と呼ぶ。(32) に対し, 双対問題と呼ばれる問題が,

$$\text{(D)} \quad \max \left\{ w = \boldsymbol{b}^T \boldsymbol{y} : \boldsymbol{A}^T \boldsymbol{y} \leq \boldsymbol{c}, \boldsymbol{y} \geq \boldsymbol{0} \right\} \tag{33}$$

で定義される。ここに, max は maximize の略であり, 目的関数を最大化する問題をあらわす。なお, $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{y} \in \mathbb{R}^m$ である[vi] 。

主-双対法は, 簡単に言うと, (P) と (D) をまとめて解く方法である。以下では, まず繰り返し計算のための初期値を定め, 続いて (P) と (D) を同時に解くために新たな問題 (SP) を定義し, さいごに繰り返し計算のアルゴリズムを述べる。

まず, $\boldsymbol{x}$ と $\boldsymbol{y}$ の適当な初期値 $\boldsymbol{x}^0 > \boldsymbol{0}$, $\boldsymbol{y}^0 > \boldsymbol{0}$ を決める。これは正であればどのように取っても構わない。次に, $\boldsymbol{p}^0 = 1/\boldsymbol{x}^0$, $\boldsymbol{t}^0 = 1/\boldsymbol{y}^0$ とおく。パラメータ $\bar{\boldsymbol{b}}$, $\bar{\boldsymbol{c}}$, β を

$$\begin{aligned} \bar{\boldsymbol{b}} &= \boldsymbol{t}^0 + \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^0 \\ \bar{\boldsymbol{c}} &= \boldsymbol{p}^0 - \boldsymbol{c} + \boldsymbol{A}^T \boldsymbol{y}^0 \\ \beta &= 1 - \boldsymbol{b}^T \boldsymbol{y}^0 + \boldsymbol{c}^T \boldsymbol{x}^0 \end{aligned} \tag{34}$$

[vi] 主問題が標準形 (STD) で書き表されているときには, 対応する双対問題は

$$\text{(D}'\text{)} \quad \max \left\{ w = \boldsymbol{b}^T \boldsymbol{y} : \boldsymbol{A}^T \boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c}, \boldsymbol{s} \geq \boldsymbol{0} \right\}$$

という形になる。標準形では, 主問題と双対問題は対称な形にならない。また, 標準形に対する双対問題では制約条件が付かない変数があらわれる。とはいっても, これらの相異はどちらかというと形式的なものである。内点法は標準形の問題にも適用可能である。アルゴリズム自体は標準形の場合も基準形の場合もほとんど同じである。なお, 歴史的経緯から, 多くの内点法の解説書では標準形が採用されている。

と定義する。続いて, 行列 $\boldsymbol{M}$ およびベクトル $\boldsymbol{q}$ を

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{0}_{m\times m} & \boldsymbol{A} & -\boldsymbol{b} & \bar{\boldsymbol{b}} \\ -\boldsymbol{A}^T & \boldsymbol{0}_{n\times n} & \boldsymbol{c} & \bar{\boldsymbol{c}} \\ \boldsymbol{b}^T & -\boldsymbol{c}^T & 0 & \beta \\ -\bar{\boldsymbol{b}}^T & -\bar{\boldsymbol{c}}^T & -\beta & 0 \end{bmatrix}$$
$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{0}_m \\ \boldsymbol{0}_n \\ 0 \\ n+m+2 \end{bmatrix} \tag{35}$$

と定義し, 次の問題を考える。

$$\text{(SP)} \quad \min\left\{\boldsymbol{q}^T\boldsymbol{\xi} : \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q} \geq \boldsymbol{0}, \boldsymbol{\xi} \geq \boldsymbol{0}\right\} \tag{36}$$

ここに, $\boldsymbol{\xi} = [\boldsymbol{y}^T, \boldsymbol{x}^T, \kappa, \theta]^T$ は $n+m+2$ 次元のベクトルである。問題 (SP) のように, 制約条件に対応する行列 $\boldsymbol{M}$ が歪対称行列[vii] であり, 非負のベクトル $\boldsymbol{q}$ が制約条件と目的関数の双方にあらわれる問題を, 歪対称問題と呼ぶ。

与えられた $\boldsymbol{\xi}$ に対し, (SP) の制約条件であらわれた式 $\boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q}$ が取る値を $\boldsymbol{s}(\boldsymbol{\xi})$ とおく。すなわち,

$$\boldsymbol{s}(\boldsymbol{\xi}) = \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q} \tag{37}$$

と定義する。また, 行列 $\boldsymbol{S}$, $\boldsymbol{\Xi}$ を

$$\boldsymbol{S} = \mathrm{diag}[s_1, \ldots, s_{m+n+2}], \quad \boldsymbol{\Xi} = \mathrm{diag}[\xi_1, \ldots, \xi_{m+n+2}] \tag{38}$$

と定義する。ただし, $\mathrm{diag}[x_, \ldots, x_k]$ は対角要素が $x_1, \ldots, x_k$ でそれ以外の要素が零の正方行列をあらわす。

以上の準備のもとで, 2 種類の内点法のアルゴリズムについて述べる。最初のアルゴリズムは, Dikin ステップ法と呼ばれるものである。

内点法 (Dikin ステップ法)

(初期化) 精度パラメータ ε を適当に決める。また, $\boldsymbol{\xi} := [(\boldsymbol{y}^0)^T, (\boldsymbol{x}^0)^T, 1, 1]^T$, $\boldsymbol{s} := \boldsymbol{s}(\boldsymbol{\xi})$ とする。さらに, ステップ幅 α を決める。ただし $0 < \alpha \leq 1$ である。

(ループ) 以下のプログラムを実行する。

while $\boldsymbol{\xi}^T\boldsymbol{s} \geq \varepsilon$ **do**
  begin
    $\boldsymbol{\xi} := \boldsymbol{\xi} + \alpha\boldsymbol{\Delta\xi}$
    $\boldsymbol{s} := \boldsymbol{s}(\boldsymbol{\xi})$
  end

ここに, $\boldsymbol{\Delta\xi}$ は

$$(\boldsymbol{S} + \Xi\boldsymbol{M})\boldsymbol{\Delta\xi} = -\frac{\boldsymbol{\xi}^2\boldsymbol{s}^2}{\|\boldsymbol{\xi}\boldsymbol{s}\|} \tag{39}$$

を解くことで得られる。

このアルゴリズムによって最適解が得られるための十分条件は, ステップ幅 α が不等式

$$0 < \alpha \leq \frac{1}{2\sqrt{n+m+2}} \tag{40}$$

を満たすことである[viii]。とくに,

$$\alpha = 1/(2\sqrt{n+m+2})$$

としたときには, 最適解を得るために必要な繰り返し計算の回数は高々

$$2(n+m+1)\log\frac{n+m+2}{\varepsilon} \tag{41}$$

回であることが示せる [5]。なお, (41) において, 小数点以下は切り上げて考えるものとする。

問題 (36) を初期値 $\boldsymbol{\xi}^0$ のもとで解き, $\boldsymbol{q}^T\boldsymbol{\xi}^* = 0$ なる最適解 $\boldsymbol{\xi}^* = [\boldsymbol{y}^*, \boldsymbol{x}^*, \kappa^*, \theta^*]^T$ が得られたとしよう。このとき $\kappa^* \neq 0$ であれば, もとの問題の最適解は

$$\bar{\boldsymbol{x}} = \frac{\boldsymbol{x}^*}{\kappa^*}, \quad \bar{\boldsymbol{y}} = \frac{\boldsymbol{y}^*}{\kappa^*} \tag{42}$$

となる。$\kappa^* = 0$ のときには, 最適解は存在しない。すなわち, 制約条件を満たす領域が存在しないか, 目的関数の値は有限にならない。これについては, 第 2.4.2 節においてもう少し詳しく説明する。

続いて, ショートステップパス追跡法と呼ばれるアルゴリズムについて述べる。これは, 次のようなアルゴリズムである。

内点法 (ショートステップパス追跡法)

(初期化) 精度パラメータ ε を適当に決める。また, $\boldsymbol{\xi} := \boldsymbol{\xi}^0$, $\boldsymbol{s} := \boldsymbol{s}(\boldsymbol{\xi})$ とする。さらに,

$$\sigma = 1 - 0.4/\sqrt{n+m+2} \tag{43}$$

とおく。

[vii] $\boldsymbol{M}$ が正方行列で, $\boldsymbol{M}^T = -\boldsymbol{M}$ となるとき, $\boldsymbol{M}$ を歪対称行列という。

[viii] $n+m+2$ はベクトル $\boldsymbol{\xi}$ の次元である。平方根の中で数 2 が加算されていることに特別な意味があるわけではない。この点に関する誤解を避けるためには, ベクトル $\boldsymbol{\xi}$ の次元を $\dim\boldsymbol{\xi}$ と書き, α に関する制約条件 (40) を

$$0 < \alpha \leq \frac{1}{2\sqrt{\dim\boldsymbol{\xi}}}$$

と書き直した方が良いであろう。

(ループ) 以下のプログラムを実行する。

$$
\begin{array}{l}
\textbf{while } \boldsymbol{\xi}^T \boldsymbol{s} \geq \varepsilon \textbf{ do} \\
\qquad \textbf{begin} \\
\qquad\qquad \mu := \boldsymbol{s}^T \boldsymbol{\xi}/(n+m+2) \\
\qquad\qquad \boldsymbol{\xi} := \boldsymbol{\xi} + \alpha \boldsymbol{\Delta\xi} \\
\qquad\qquad \boldsymbol{s} := \boldsymbol{s}(\boldsymbol{\xi}) \\
\qquad \textbf{end}
\end{array}
$$

ここに, $\boldsymbol{\Delta\xi}$ は線形方程式

$$(\boldsymbol{S} + \boldsymbol{\Xi M})\boldsymbol{\Delta\xi} = -\boldsymbol{s\xi} + \sigma\mu\mathbf{1} \tag{44}$$

の解である。各々のステップで (44) を解き直す必要があることに注意せよ。

Dikin ステップ法は最急降下法の一種である。これに対して, ショートステップパス追跡法は Newton 法の一種である。一般にショートステップパス追跡法は Dikin ステップ法よりは高速である。Dikin ステップ法, ショートステップパス追跡法のいずれも, 理論的な特性は良いが, 実際の問題に適用すると実行が極めて遅いという特徴がある。より実用的なアルゴリズムとしては, ロングステップパス追跡法, Mehrotra の予測子・修正子法などがある.

以下では, まず第 2.4.2 節において, 本節で利用された行列 $\boldsymbol{M}$ などの意味を説明する。続いて, 第 2.4.3 節において Dikin ステップ法の意味を説明する。第 2.4.4 節では, Newton 法に基づく内点法の一般的なアルゴリズムについて説明する。これらは実験とは直接関係ないので, 興味のない読者は読み飛ばしてよい。

▶

2.4.2 線形計画問題 (SP) の意味

本節では, 第 2.4 節の行列 $\boldsymbol{M}$ などの意味についていくつかの証明を省いた上で述べる。この節は実験そのものには関係ないので, 興味のない読者は読み飛ばしてよい。

主問題 (P) とその双対問題 (D) が与えられているものとする。主問題の制約条件 ($\boldsymbol{Ax} \geq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}$) を満たす点の集合を $\mathcal{P}$ と書き, 双対問題の制約条件 ($\boldsymbol{A}^T\boldsymbol{y} \leq \boldsymbol{c}, \boldsymbol{y} \geq \boldsymbol{0}$) を満たす点の集合を $\mathcal{D}$ と書く。$\mathcal{P}$ や $\mathcal{D}$ が空集合となることもある。このような場合には, 主問題や双対問題には解が存在しない。

さて, 適当に取られた $\boldsymbol{x} \in \mathcal{P}$ と $\boldsymbol{y} \in \mathcal{D}$ に対し, $\boldsymbol{b}^T\boldsymbol{y} \leq (\boldsymbol{x}^T\boldsymbol{A}^T)\boldsymbol{y} = \boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{y}) \leq \boldsymbol{x}^T\boldsymbol{c}$ より, $\boldsymbol{b}^T\boldsymbol{y} \leq \boldsymbol{c}^T\boldsymbol{x}$ という不等式が成り立つことがわかる。すなわち, 双対問題の目的関数値は必ず主問題の目的関数値以下となる。このことから, ある $\boldsymbol{x}^* \in \mathcal{P}$ と $\boldsymbol{y}^* \in \mathcal{D}$ が等式 $\boldsymbol{b}^T\boldsymbol{y}^* = \boldsymbol{c}^T\boldsymbol{x}^*$ を満たせば, $\boldsymbol{x}^*$ は主問題の最適解であり, $\boldsymbol{y}^*$ は双対問題の最適解であるということがわかる。実は, この逆も成り立つ。すなわち, 主問題と双対問題がそれぞれ有限の最適解 $\boldsymbol{x}^*$ と $\boldsymbol{y}^*$ を持つとき, 等式 $\boldsymbol{b}^T\boldsymbol{y}^* = \boldsymbol{c}^T\boldsymbol{x}^*$ が成り立つということが言える。後半の結果を双対定理という。双対定理の証明はやや繁雑なのでここでは述べない。

さて, 最適解のための等式 $\boldsymbol{b}^T\boldsymbol{y} = \boldsymbol{c}^T\boldsymbol{x}$ に不等号を追加した不等式

$$\boldsymbol{b}^T\boldsymbol{y} \geq \boldsymbol{c}^T\boldsymbol{x} \tag{45}$$

を考えよう。不等式 $\boldsymbol{b}^T\boldsymbol{y} \leq \boldsymbol{c}^T\boldsymbol{x}$ はつねに成り立つのだから, これは実質的にもとの等式と変わらない。(45) の右辺を左辺に移項し, 主問題の制約条件, 双対問題の制約条件の両辺に -1 を乗じたものと合わせて整理すると,

$$
\begin{aligned}
\boldsymbol{Ax} - \boldsymbol{b} &\geq \boldsymbol{0} \\
-\boldsymbol{A}^T\boldsymbol{y} + \boldsymbol{c} &\geq \boldsymbol{0} \\
\boldsymbol{b}^T\boldsymbol{y} - \boldsymbol{c}^T\boldsymbol{x} &\geq 0
\end{aligned}
$$

という不等式が得られる。これを行列の形でまとめると,

$$
\begin{bmatrix} \boldsymbol{0} & \boldsymbol{A} & -\boldsymbol{b} \\ -\boldsymbol{A}^T & \boldsymbol{0} & \boldsymbol{c} \\ \boldsymbol{b}^T & -\boldsymbol{c}^T & 0 \end{bmatrix}
\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ 1 \end{bmatrix}
\geq
\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ 0 \end{bmatrix}, \boldsymbol{x} \geq \boldsymbol{0}, \boldsymbol{y} \geq \boldsymbol{0} \tag{46}
$$

という形になる。この段階ではまだ (46) が解を持つかどうかは明らかでないことに注意しよう。

ここで, (46) の左辺のベクトルの最下段にあらわれた数 1 を変数 κ で置き変えた不等式

$$
\begin{bmatrix} \boldsymbol{0} & \boldsymbol{A} & -\boldsymbol{b} \\ -\boldsymbol{A}^T & \boldsymbol{0} & \boldsymbol{c} \\ \boldsymbol{b}^T & -\boldsymbol{c}^T & 0 \end{bmatrix}
\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \kappa \end{bmatrix}
\geq
\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ 0 \end{bmatrix},
\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \\ \kappa \end{bmatrix}
\geq
\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ 0 \end{bmatrix} \tag{47}
$$

を考える。(47) を満たす $(\boldsymbol{y}^T, \boldsymbol{x}^T, \kappa)$ が求められていて, かつ $\kappa > 0$ であれば, $(\boldsymbol{y}^T/\kappa, \boldsymbol{x}^T/\kappa, 1)$ は (46) を満たす。すなわち, 新しい変数 κ を導入することはもとの問題 (46) の解を求めることの障害とはならない。では, 新しい変数 κ を導入することには何か利点はあるのだろうか。まず (47) が自明な解 $\boldsymbol{x} = \boldsymbol{0}$, $\boldsymbol{y} = \boldsymbol{0}$, $\kappa = 0$ を持つことに注意する。ところで, 内点法を用いて問題 (47) の解を求めると, 問題 (46) に解があるときには κ は零以外の値に収束し, 問題 (46) に解がないときには κ は零に収束することが示せる (証明は略す)。すなわち, κ を (46) の解の存在判定に使うことができる。

さて, (47) の左端の行列は先に内点法のアルゴリズムで使った (35) と似てはいるが, 同一ではない。(35) の行列 $\boldsymbol{M}$ の右端および下段の要素は (47) にはないものである。これはどのような経緯で出て来たのだろうか?

実は, 線形計画問題を内点法で解くためには初期値として制約領域の内点 (制約条件から等号を取り除いた不等式を満たす点) が必要になるのだが, 内点を 1 個定めることはそれほど容易でない。そのため, 代替策として, 新たに変数を 1 個追加して, (47) から内点が既知の拡大された問題に変換する。(35) は (47) にそのような手法を適用した結果なのである。以下では, この方法について見てゆこう。

初期値 $\boldsymbol{x}^0 > \boldsymbol{0}$, $\boldsymbol{y}^0 > \boldsymbol{0}$ を適当に決め (各成分が 0 でなければ何でもよい), $\boldsymbol{p}^0 = 1/\boldsymbol{x}^0$, $\boldsymbol{t}^0 = 1/\boldsymbol{y}^0$ とおく。パラメータ $\bar{\boldsymbol{b}}, \bar{\boldsymbol{c}}, \beta$ を (34) によって定義する。また, 行列 $\boldsymbol{M}$ およびベクトル $\boldsymbol{q}$ を (35) によって決め, (36) によって定められた問題 (SP) を考える。すると, 初期値

$$\boldsymbol{\xi}^0 = \begin{bmatrix} \boldsymbol{y}^0 \\ \boldsymbol{x}^0 \\ 1 \\ 1 \end{bmatrix} \tag{48}$$

が問題 (36) の内点となるのである。これを見ておこう。$\boldsymbol{\xi}^0 > \boldsymbol{0}$ は定義より明らかだから, $\boldsymbol{M}\boldsymbol{\xi}^0 + \boldsymbol{q} > \boldsymbol{0}$ となることを見ればよい。これは, (34) を使えば,

$$\begin{aligned} \boldsymbol{M}\boldsymbol{\xi}^0 + \boldsymbol{q} &= \begin{bmatrix} \boldsymbol{A}x^0 - \boldsymbol{b} + \bar{\boldsymbol{b}} \\ -\boldsymbol{A}^T\boldsymbol{y}^0 + \boldsymbol{c} + \bar{\boldsymbol{c}} \\ \boldsymbol{b}^T\boldsymbol{y}^0 - \boldsymbol{c}^T\boldsymbol{x}^0 + \beta \\ -\bar{\boldsymbol{b}}^T\boldsymbol{y}^0 - \bar{\boldsymbol{c}}^T\boldsymbol{x}^0 - \beta + (n+m+1) \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{t}^0 \\ \boldsymbol{p}^0 \\ 1 \\ 1 \end{bmatrix} \end{aligned} \tag{49}$$

となることからわかる。

2.4.3 Dikin ステップ法の意味

続いて, 繰り返し計算で用いられる探索ベクトル $\boldsymbol{\Delta\xi}$ を求める式 (39) の意味を簡単に説明しておく。

$\boldsymbol{s}(\boldsymbol{\xi}) = \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q} > \boldsymbol{0}$, $\boldsymbol{\xi} > \boldsymbol{0}$ を満たす $\boldsymbol{\xi}$ が与えられているものとする。記法の簡単のために, $\boldsymbol{s}(\boldsymbol{\xi})$ を $\boldsymbol{s}$ と略記する。さて, $\boldsymbol{M}$ は歪対象行列だから, $\boldsymbol{\xi}^T\boldsymbol{M}\boldsymbol{\xi} = 0$ となる[ix] 。ゆえに, 目的関数の値 $\boldsymbol{q}^T\boldsymbol{\xi}$ は

$$\boldsymbol{q}^T\boldsymbol{\xi} = \boldsymbol{\xi}^T\boldsymbol{s}$$

とも書ける[x] 。さて, $\boldsymbol{\xi}$ を $\boldsymbol{\xi} + \boldsymbol{\Delta\xi}$ に変更し,

$$\boldsymbol{\Delta s} = \boldsymbol{s}(\boldsymbol{\xi} + \boldsymbol{\Delta\xi}) - \boldsymbol{s}(\boldsymbol{\xi})$$

とおくと, 目的関数の値は

$$(\boldsymbol{\xi} + \boldsymbol{\Delta\xi})^T(\boldsymbol{s} + \boldsymbol{\Delta s}) - \boldsymbol{\xi}^T\boldsymbol{s} = \boldsymbol{\xi}^T\boldsymbol{\Delta s} + \boldsymbol{\Delta\xi}^T\boldsymbol{s}$$

だけ変動する[xi] 。そこで, 単純に考えると, $\boldsymbol{\xi}^T\boldsymbol{\Delta s} + \boldsymbol{\Delta\xi}^T\boldsymbol{s}$ が最も減少する方向に $\boldsymbol{\Delta\xi}$ を取れば良さそうに思える。しかし, これは必ずしも望ましくない。解が制約条件を満たす領域の境界に近付きすぎると計算効率が落ちることがあるのだが, 上述の $\boldsymbol{\Delta\xi}$ の取り方ではこの問題が考慮されていないからである。そこで, 新たに

$$\left\| \frac{\boldsymbol{\Delta\xi}}{\boldsymbol{\xi}} + \frac{\boldsymbol{\Delta s}}{\boldsymbol{s}} \right\| \le 1 \tag{50}$$

という制約条件を課し, (50) を満たす領域 (この領域を Dikin 楕円体と呼ぶ) の中から $\boldsymbol{\xi}^T\boldsymbol{\Delta s} + \boldsymbol{\Delta\xi}^T\boldsymbol{s}$ を最小化する $\boldsymbol{\Delta\xi}$ を求めたものが (39) である。(50) の制約条件の形から, $\boldsymbol{\xi}$ あるいは $\boldsymbol{s}$ の成分のうち値が 0 に近いものについては, 対応する $\boldsymbol{\Delta\xi}$ あるいは $\boldsymbol{\Delta s}$ の成分の値はあまり大きく取れず, 結果として $\boldsymbol{\xi}$ あるいは $\boldsymbol{s}$ が 0 に近付くことが抑制される。この方法は, $\boldsymbol{\xi}$ や $\boldsymbol{s}$ の大きさに応じて重みを付けた, 最急降下法の一種と考えられる。

ここで, ごく簡単な歪対称問題について, 制約領域の各点で Dikin 楕円体がどのような形になるかを見ておこう。ただし, ここで取り扱うのは 2 変数の人工的な歪対称問題であり, 主問題 (P) と双対問題 (D) から決まる歪対称問題ではない。これは, 主問題 (P) および双対問題 (D) が決まる歪対称問題は最低でも 4 変数となり, 2 次元のグラフでは Dikin 楕円体をうまく書き表せないからである。

例 3 2 変数の歪対称問題

$$\begin{aligned} \min \Bigg\{ &\begin{bmatrix} 1 \\ 0 \end{bmatrix}^T \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} : \\ &\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \ge \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \ge \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Bigg\} \end{aligned} \tag{51}$$

[ix] $\boldsymbol{\xi}^T\boldsymbol{M}\boldsymbol{\xi}$ がスカラーであることと $\boldsymbol{M}^T = -\boldsymbol{M}$ から,

$$\boldsymbol{\xi}^T\boldsymbol{M}\boldsymbol{\xi} = (\boldsymbol{\xi}^T\boldsymbol{M}\boldsymbol{\xi})^T = \boldsymbol{\xi}^T\boldsymbol{M}^T\boldsymbol{\xi} = -\boldsymbol{\xi}^T\boldsymbol{M}^T\boldsymbol{\xi},$$

よって $2\boldsymbol{\xi}^T\boldsymbol{M}\boldsymbol{\xi} = 0$。

[x] $\boldsymbol{\xi}^T\boldsymbol{M}\boldsymbol{\xi} = 0$ となることに注意すれば, $\boldsymbol{\xi}^T\boldsymbol{s} = \boldsymbol{\xi}^T(\boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q}) = \boldsymbol{\xi}^T\boldsymbol{q}$ がいえる。

[xi] $(\boldsymbol{\Delta\xi})^T\boldsymbol{\Delta s} = (\boldsymbol{\Delta\xi})^T\boldsymbol{M}\boldsymbol{\Delta\xi} = 0$

を考える。

(51) の制約条件は, $0 \le \xi_1$, $0 \le \xi_2 \le 1$ と書き直される。目的関数は ξ_1 なので, $\xi_1 = 0$ を満たす任意の点で目的関数は最小となる。

図 3 に (51) の制約条件を満たす領域の各点における Dikin 楕円体の形を示す。制約条件を満たす領域は図

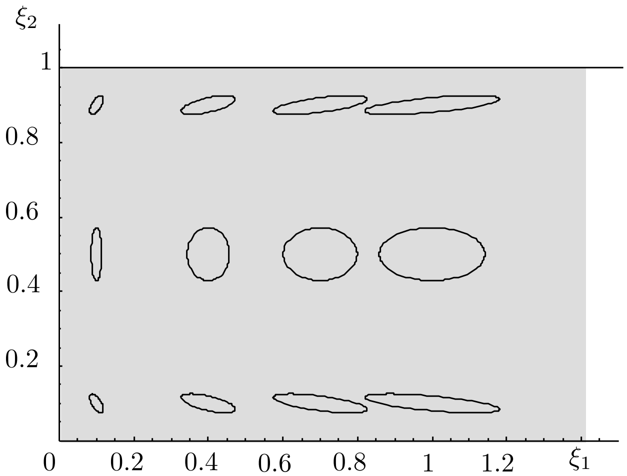


図 3 2 変数の歪対称問題に対する Dikin 楕円体

中の網のかかった領域である。この図から, 境界に近付くにつれ Dikin 楕円体が潰れてゆくことがわかる。

2.4.4 Newton 法に基づく内点法の一般的なアルゴリズム

本節では, Newton 法に基づく内点法のアルゴリズムについて簡単に述べる。

第 2.4.2 節の結果から, 主問題と双対問題を解くことは, 問題

$$(\mathrm{SP}) \quad \min\left\{\boldsymbol{q}^T\boldsymbol{\xi} : \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q} \ge \boldsymbol{0}, \boldsymbol{\xi} \ge \boldsymbol{0}\right\}$$

を解くことに帰着されることがわかっている。ここに, $\boldsymbol{M}$ は歪対称行列であり, $\boldsymbol{q}$ は各成分が非負のベクトルである。そこで, 本節では (SP) を出発点とする。

2.4.4.1 強相補性条件 行列 $\boldsymbol{M}$ が歪対称であることから, 目的関数値は $\boldsymbol{q}^T\boldsymbol{\xi} = \boldsymbol{s}^T\boldsymbol{\xi}$ を満たす[xii)]。

さて, 問題 (SP) は自明な最適解 $\boldsymbol{\xi} = \boldsymbol{0}$ を持つ。しかし, 第 2.4.2 で説明したように, 我々は $\boldsymbol{\xi}$ のある成分が収束する値が 0 であるか否かに応じて主問題 (P) と双対問題 (D) の解の存在性の判定をするので, 自明な解は我々の望むものではない。ところで, $\boldsymbol{\xi}$ が最適であり ($\boldsymbol{s}^T\boldsymbol{x} = 0$), かつ $\boldsymbol{\xi}$ の要素の中の零でないものがあったとしよう。$\boldsymbol{\xi} \ge \boldsymbol{0}$, $\boldsymbol{s} \ge \boldsymbol{0}$ であることを思いだすと, $\boldsymbol{\xi}$ と $\boldsymbol{s}$ の成分は,

- $\xi_i = 0$ なら $s_i > 0$,
- $\xi_i > 0$ なら $s_i = 0$

という条件を満たすことがわかる。この条件を強相補性条件という。強相補性条件は, より簡単に,

$$\boldsymbol{s} + \boldsymbol{\xi} > \boldsymbol{0} \tag{52}$$

とも書ける。以上の議論により, 我々は問題 (SP) の強相補性条件を満たす最適解を (もしあれば) 見付けなければならないことがわかる。

2.4.4.2 中心パス ここで, 中心パスという概念を導入しておく。$\boldsymbol{1}$ をすべての要素が 1 のベクトルとし, μ を正の定数として,

$$\boldsymbol{s}\boldsymbol{\xi} = \mu\boldsymbol{1} \tag{53}$$

なる等式を考える。ここでは証明を省くが, (SP) の制約条件を満たす領域が内点を持つ (開集合を含む) とき, (53) と (SP) の制約条件を連立させた非線形連立方程式

$$\begin{aligned} \boldsymbol{s} &= \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q}, \\ \boldsymbol{s}\boldsymbol{\xi} &= \mu\boldsymbol{1} \end{aligned} \tag{54}$$

を $\boldsymbol{\xi}$ について解くと, 与えられた μ に対して制約領域の内点にある $\boldsymbol{\xi}(\mu)$ が一意的に定まることが言える。さらに, 次の事実が成り立つ。

- $\boldsymbol{\xi}(\mu)$ は μ の連続関数である。
- $\mu \to 0$ とすると, $\boldsymbol{\xi}(\mu)$ は強相補性条件を満たす点に収束する。

これらの事実から, 軌道 $\boldsymbol{\xi}(\mu)$(これを中心パスと呼ぶ) を追跡することにより強相補性条件を満たす解が求められることがわかる。中心パスは, 制約条件を満たす領域の内部を通って最適解に到達する, 幾何学的に性質の良い軌道である。

2.4.4.3 アフィンスケーリング方向と中心化方向 以上の議論を踏まえた上で, 2 種類の異なる探索ベクトルの決め方について述べよう。

まず最初に, (SP) を変数 $\boldsymbol{x}$ と $\boldsymbol{s}$ に関する非線形関数

$$\boldsymbol{F}(\boldsymbol{\xi}, \boldsymbol{s}) = \begin{bmatrix} \boldsymbol{M}\boldsymbol{\xi} - \boldsymbol{s} + \boldsymbol{q} \\ \boldsymbol{s}\boldsymbol{\xi} \end{bmatrix} \tag{55}$$

xii) 先と同様に, $\boldsymbol{s} = \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{q}$ とする

の零点を求める問題とみなす。点 $\boldsymbol{\xi}$ と $\boldsymbol{s} = \boldsymbol{M\xi}+\boldsymbol{q}$ が与えられているとき, $\boldsymbol{\xi}' = \boldsymbol{\xi} + \boldsymbol{\Delta\xi}_a$ および $\boldsymbol{s}' = \boldsymbol{s} + \boldsymbol{\Delta s}_a$ として $\boldsymbol{F}(\boldsymbol{\xi}', \boldsymbol{s}') = \boldsymbol{0}$ を $\boldsymbol{\Delta\xi}_a$ と $\boldsymbol{\Delta s}_a$ について Tailor 展開によって近似的に解くと, $\boldsymbol{\Delta\xi}_a$ と $\boldsymbol{\Delta s}_a$ は方程式

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{I} \\ \boldsymbol{S} & \boldsymbol{\Xi} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta\xi}_a \\ \boldsymbol{\Delta s}_a \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ -\boldsymbol{s\xi} \end{bmatrix} \tag{56}$$

の解となる。なお, 適当な正の定数 α を取り, $\boldsymbol{\xi}' = \boldsymbol{\xi} + \alpha\boldsymbol{\Delta\xi}_a$ および $\boldsymbol{s}' = \boldsymbol{s} + \alpha\boldsymbol{\Delta s}_a$ とすると,

$$\boldsymbol{M}(\boldsymbol{\xi}') + \boldsymbol{q} = \boldsymbol{s} + \alpha\boldsymbol{M\Delta\xi}_a = \boldsymbol{s}'$$

となる。よって, 任意の α に対し, $\boldsymbol{s}'$ と $\boldsymbol{\xi}'$ は等式 $\boldsymbol{s}' = \boldsymbol{M\xi}' + \boldsymbol{q}$ を満たす。(56) を解くことによって得られる探索ベクトル $(\boldsymbol{\Delta\xi}_a^T, \boldsymbol{\Delta s}_a^T)^T$ をアフィンスケーリング方向と呼ぶ。

アフィンスケーリング方向は目的関数を減少させるには都合が良いが, 制約条件を見たす領域の境界に解を誘導する可能性があるという点では不都合である。ところで, 解を内点に誘導するには, 中心パス上の点を利用するのが良い。そこで, パラメータ μ を適当に決め, それ対応する中心パス上の点を求める問題を考える。パラメータ μ の選び方にはいろいろなものが考えられるが, たとえば目的関数値を現在の値の $1/\dim\boldsymbol{\xi}$ にすることを目指す, すなわち $\mu = (1/\dim\boldsymbol{\xi})\boldsymbol{s}^T\boldsymbol{\xi}$ とするという方法が考えられる。この問題も, 先と同様に, 関数

$$\boldsymbol{G}(\boldsymbol{\xi}, \boldsymbol{s}) = \begin{bmatrix} \boldsymbol{M\xi} - \boldsymbol{s} + \boldsymbol{q} \\ \boldsymbol{s\xi} - \mu\boldsymbol{1} \end{bmatrix} \tag{57}$$

の零点を求める問題に帰着される。$\boldsymbol{\xi}' = \boldsymbol{\xi} + \boldsymbol{\Delta\xi}_c$, $\boldsymbol{s}' = \boldsymbol{s} + \boldsymbol{\Delta s}_c$ とし, (57) を Tailor 展開によって近似的に解くと, $\boldsymbol{\Delta\xi}_c$ と $\boldsymbol{\Delta s}_c$ は方程式

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{I} \\ \boldsymbol{S} & \boldsymbol{\Xi} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta\xi}_c \\ \boldsymbol{\Delta s}_c \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ -\boldsymbol{s\xi} + \mu\boldsymbol{1} \end{bmatrix} \tag{58}$$

の解となる。(58) を解くことによって得られる探索ベクトル $(\boldsymbol{\Delta\xi}_c^T, \boldsymbol{\Delta s}_c^T)^T$ を中心化方向と呼ぶ。

これらの探索ベクトルを用いた解法は, Newton 法による非線形方程式の求解の一種となる。

2.4.4.4 アルゴリズムの一般形 アフィンスケーリング方向は目的関数を減少させる方向, 中心化方向は中心パスに向かう方向であった。Newton 法型の内点法のアルゴリズムでは, アフィンスケーリング方向と中心化方向の線形結合を使って解を更新してゆく。

σ を 0 以上 1 以下の定数とし, 探索ベクトルを

$$\begin{bmatrix} \boldsymbol{\Delta\xi} \\ \boldsymbol{\Delta s} \end{bmatrix} = (1-\sigma)\begin{bmatrix} \boldsymbol{\Delta\xi}_a \\ \boldsymbol{\Delta s}_a \end{bmatrix} + \sigma\begin{bmatrix} \boldsymbol{\Delta\xi}_c \\ \boldsymbol{\Delta s}_c \end{bmatrix} \tag{59}$$

とすると, 探索ベクトル $(\boldsymbol{\Delta\xi}^T, \boldsymbol{\Delta s}^T)^T$ は

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{I} \\ \boldsymbol{S} & \boldsymbol{\Xi} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta\xi} \\ \boldsymbol{\Delta s} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ -\boldsymbol{s\xi} + \sigma\mu\boldsymbol{1} \end{bmatrix} \tag{60}$$

の解となる。

なお, (60) は $\boldsymbol{\Delta\xi}$ と $\boldsymbol{\Delta s}$ に関する線形方程式であるが, (60) の上半分は $\boldsymbol{\Delta s}$ について解けて $\boldsymbol{\Delta s} = \boldsymbol{M\Delta\xi}$ となるので, 実際にはこれを (60) の下半分に代入した

$$\left(\boldsymbol{S} + \boldsymbol{\Xi M}\right)\boldsymbol{\Delta\xi} = -\boldsymbol{s\xi} + \sigma\mu\boldsymbol{1} \tag{61}$$

を $\boldsymbol{\Delta\xi}$ について解けばよい.

(60) の解 $(\boldsymbol{\Delta\xi}^T, \boldsymbol{\Delta s}^T)^T$ を探索ベクトルとして利用する内点法のアルゴリズムの一般形は, 以下に述べるようなものである。

内点法 (Newton 法による解法)

(初期化) 精度パラメータ ε を適当に決める。また, $\boldsymbol{\xi} := \boldsymbol{\xi}^0$, $\boldsymbol{s} := \boldsymbol{s}(\boldsymbol{\xi})$ とする。

(ループ) 以下のプログラムを実行する。

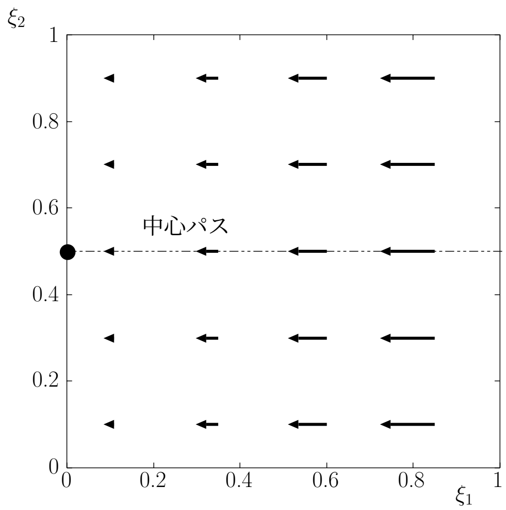
while $\boldsymbol{\xi}^T\boldsymbol{s} \geq \varepsilon$ **do**
　begin
　　σ を決める
　　$\mu := \boldsymbol{s}^T\boldsymbol{\xi}/(n)$ とする
　　(61) を解いて $\boldsymbol{\Delta\xi}$, $\boldsymbol{\Delta s}$ を決める
　　α を決める
　　$\boldsymbol{\xi} := \boldsymbol{\xi} + \alpha\boldsymbol{\Delta\xi}$
　　$\boldsymbol{s} := \boldsymbol{s} + \alpha\boldsymbol{\Delta s}$
　end

第 2.4.1 節で述べたショートステップパス追跡法は, 上記のアルゴリズムにおいて $\alpha = 1$, $\sigma = 1 - 0.4/\sqrt{\dim\boldsymbol{\xi}}$ と固定した特別な場合である。大規模な問題では, このように σ を決めると, 探索ベクトルが中心化方向とほぼ一致してしまうため, ショートステップパス追跡法は極めて低速になる。

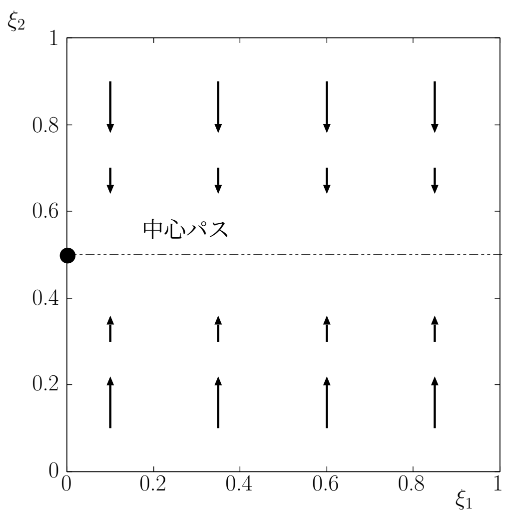
ループ内の各ステップでパラメータ α や σ の値を切り換えると, より効率の良いアルゴリズムが構成しうることが知られている。興味のある読者はたとえば文献 [2, 5, 6] を参照してほしい。

例 4 例 3 の歪対称問題を用いて, アフィンスケーリング方向と中心化方向の各点における変動を見よう。まず最初に, 例 3 の問題 (51) において, 中心パスは $\xi_2 = 1/2$

という直線であり, 強相補解は $(\xi_1, \xi_2) = (0, 1/2)$ という点である。図 4 に各点におけるアフィンスケーリング方向と中心化方向を示す。この例ではアフィンスケーリング方向は中心パスと平行になる。それゆえ, アフィンスケーリング方向のみを用いて解を更新すると強相補解に到達できない。また, $\mu = (1/2)\boldsymbol{s}^T\boldsymbol{\xi}$ の場合の中心化方向は中心パスと垂直となり, これを用いて解を更新すると目的関数は減少しない。



(a) アフィンスケーリング方向



(b) 中心化方向

図 4 各点における探索ベクトルの方向

◀

3 予習事項

- Scilab の基本的な使い方を復習しておくこと。
- ディレクトリ

  `/home/b/teacher/hanba/info-eng`

  に Scilab の使い方のサンプルが記載されたファイル sample.sci とシンプレックス法のプログラム作成時に使う関数が定義されたファイル functions.sci が準備されている。実験前にこれらのファイルに目を通しておくこと。

4 実験

4.1 実験課題

実験課題 1 シンプレックス法 (ビッグ M 法) を用いて与えられた線形計画問題を解くプログラムを作製せよ。

実験課題 2 内点法 (Dikin ステップ法) を用いて与えられた線形計画問題を解くプログラムを作製せよ。

実験課題 3 内点法 (ショートステップパス追跡法) を用いて与えられた線形計画問題を解くプログラムを作製せよ。

なお, 余力のない者は実験課題 3 はやらなくてよい。

4.2 実験方法

解くべき問題に対応する行列 $\boldsymbol{A}$ とベクトル $\boldsymbol{b}$ はディレクトリ

`/home/b/teacher/hanba/info-eng/`

にある practice.dat と problem.dat という 2 個のファイルに記載されている。このうち practice.dat は規模が小さい練習用のファイル, problem.dat は問題ファイルである。problem.dat に記載された問題を解くのには相当の時間を要するので, 作成中の未完成プログラムをテストするときの便宜のために短時間で解ける問題ファイル practice.dat が用意されている。

問題ファイルの行列 $\boldsymbol{A}$ は基準形である。行列 $\boldsymbol{A}$ の各成分を a_{ij}, ベクトル b の各成分を b_i とすると, このファイルには行列 $\boldsymbol{A}$ の各要素とベクトル $\boldsymbol{b}$ の要素が

$$\begin{array}{ccccc} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{array} \tag{62}$$

のような順で記載されている。すなわち, 行列 $\boldsymbol{A}$ の第 1 列ベクトル, 第 2 列ベクトル, ..., 第 n 列ベクトルおよびベクトル $\boldsymbol{b}$ が順に横に並べられて記載されている。

目的関数に対応するベクトル $\boldsymbol{c}$ の値については実験中に指示する。

[注意事項]

- Scilab のエラーメッセージは英語なので, 実験の際には必ず英和辞典を持参すること。
- 問題ファイル, LaTeX のサンプルファイル, Scilab で書かれたサンプルプログラムがディレクトリ

  `/home/b/teacher/hanba/info-eng/`

  にまとめられている。このディレクトリにあるファイルの一覧は以下の通りである。

  practice.dat 練習用の問題ファイル。

  problem.dat 問題ファイル。

  sample.tex レポート用の LaTeX のサンプルファイル。

  f1.eps サンプルの PostScript ファイル。

  sample.sci Scilab の使い方のサンプルファイル。

  functions.sci シンプレックス法のプログラム作成に必要と。なるいくつかの関数が定義されたファイル。

  実験に先立ってこれらのファイルを各自のディレクトリにコピーしておくこと。

- 作成中の未完成プログラムをテストするときには必ず practice.dat の方を用いること。practice.dat を用いてプログラムが正常に動作していることを確認してから使用する問題ファイルを problem.dat に切り換えるとよい。プログラムを問題の大きさに依存しないで動作するように作成しておけば, 問題ファイルの切り換えは容易である。
- Scilab の使い方の詳細については

  `~harada/Lab/scilab-intro.pdf`

  を参照すること。

- ファイル sample.sci にはシンプレックス法と Dikin ステップ法のほぼ完成したプログラムが記載されている。実験課題 1 と実験課題 2 を達成するには, これを若干加工すればよい。また, 実験課題 3 を

達成するには, 実験課題 2 のプログラムから出発して, 探索ベクトルを決めている部分を (44) にしたがって書き直せばよい。

- シンプレックス法のプログラム作成にはかなり手間がかかるので, プログラム作成のために必要となるいくつかの関数をディレクトリ functions.sci に用意してある。sample.sci のシンプレックス法の部分では, functions.sci を利用することが前提になっている。

- レポート提出は電子メールのみで受け付ける。提出物は,

  - 作成した LaTeX のソースコード
  - レポート本文の PostScript ファイル

  である。レポート作製のためのサンプル sample.tex を適宜利用すること。

- 各学生が提出したレポートは, 過去に他のグループなどの学生が提出したレポートと電子的に比較される。他の学生のレポートとの類似性が著しいレポートは他人のレポートがまる写ししたものとみなされる。この場合, レポートの得点は 0 点となる。

- レポート作成の際に実験指導書の原理の項をまる写ししてはならない。レポートには, 実験指導書の非常に簡単な要約や, あるいは (もしあれば) 自分で調べた追加事項を書くべきである。レポートにおける実験の原理の記載が実験指導書の内容以上のものを含まない場合は, 原理の部分のページ数の上限を 1 ページとし, それを越えるレポートは大幅に減点する。 実験指導書にない事項が調べられているレポートについてはこの限りではない。また, レポート中における実験の手順に関する記述も最低限にとどめること。また, 実験指導書に記載された例は諸君の理解を助けるためのものであるから, これをレポートに記載するのは不適切である (担当教官にとっては自明である)。例がまる写しされたレポートは大幅に減点する。この実験では長いレポートを書く必要はまったくない。レポートの分量そのものは評価の対象とはならないので注意すること。なるべく簡潔に要点のみ記述することを心掛けよ。

5 データ解析

- シンプレックス法と Dikin ステップ法の挙動を比較せよ。

- Dikin ステップ法とショートステップパス追跡法の挙動を比較せよ (実験課題 3 ができなかった者はやらなくてよい)。

- 実験用のディレクトリに

  `/home/b/teacher/hanba/info-eng`

  に, 制約条件の数が異なる 1.dat から 12.dat までのファイルが用意されている。これらを利用して, 問題の大きさが変わったときの各アルゴリズムの挙動を比較せよ (実験課題 3 ができなかった者はやらなくてよい)。

参考文献

[1] S. -C. Fang, S. Puthenpura: Linear Optimization and Extensions: Theoory and Algorithms, Prentice Hall (1993)

[2] 小島 政和, 土谷 隆, 水野 眞治, 矢部 弘: 「内点法」, 朝倉書店 (2001)

[3] 今野 浩: 「線形計画法」, 日科技連 (1987)

[4] 藤田 宏, 今野 浩, 田邉 國士: 「最適化法」, 岩波書店 (1994)

[5] C. Roos, T. Terlaky and J. -Ph. Vial: Theory and Algorithms for Linear Optimization, Wiley (1997)

[6] S. J. Wright: Primal-Dual Interior-Point Mehtods, SIAM (1997)