# システム最適化

工学システム学類対象

担当：宮本定明、遠藤靖典

2015 年度春 A,B

## まえがき

最適化の概念は，極めて一般的であり，すべてのシステムにおいて考察することが必要であるといっても過言ではない．これに比べて，最適化技法という場合，より限定された意味をもった具体的な理論とアルゴリズムを指している．ここで講義するのは，後者の諸方法であるが，通常の教科書・参考書等に比べて，内容を易しくして理論的に高度な部分は省略しつつ，幾何的・直観的な解釈によって主題が理解できるようにしている．一方で，分量に比べて広範な内容を扱い，現実の事例において生じる複雑な問題に対する技法も述べている．

システム最適化においては，数学（解析学と線形代数学）の上に立脚した議論と，計算機アルゴリズムに関連する部分が有機的に関連しつつ，現実のシステムにおける効率改善の問題を定式化し，解決する手段が与えられている．

受講者に望むことは次の 3 つである．まず，数学の用語・語法・記号の使用に習熟すること．また，用いられている数学とアルゴリズムについて復習すること．そのことによって，ここで述べる線形計画法，非線形計画法，組み合わせ最適化について，詳細にわたり記憶する必要はないとしても，その基礎・考え方を理解することである．

# 目 次

# 第1章　はじめに

この講義では，システム最適化について線形計画法，非線形計画法，組み合わせ最適化などを講述する．これらのテーマは基礎的な数学と次のように関係している．

**(1)** 線形計画法—線形代数が基礎

**(2)** 非線形計画法—解析学が基礎

**(3)** 組み合わせ最適化の技法—グラフ理論，アルゴリズム，計算論

まず，最適化という言葉の意味からはじめよう．

## 最適化という用語

**最適化**とは **optimization** の訳である．関連した言葉をあげると，最適 (optimal, optimum)，最適解 (optimal solution, optimum solution)，最適アルゴリズム (optimal algorithm)，最適にする (optimize).

Webster の辞書では，optimal, optimum という言葉は次のように説明されている．

> the amount of a degree of something that is most favorable to some end; esp: the most favorable condition (as of temperature, light, moisture, food) for the growth and reproduction of an organism.

工学における広い意味での最適化の概念は，上の一般的説明に従っている (たとえば，プログラムコンパイル時の最適化など)．これに対して，システム工学で最適化問題 (optimization problem) を扱うときの最適化はより狭い特定の意味で使われている．

すなわち，狭い意味での最適化とは，最小化と最大化をあわせた意味に他ならない．

$$\text{最適化 (optimization)} = \begin{cases} \text{最小化 (minimization)} \\ \text{最大化 (maximization)} \end{cases}$$

この講義では，主に後の意味での最適化を扱うが，先に，工学における広い意味での最適化についてふれておこう．

広い意味での最適化は，効率を良くするという意味で使われ，システム工学だけでなく，工学における基本的な概念の一つである．広義の最適化に関するいくつかの注意を行おう．

- 最適化は，必ずしも**定式化** (formulation, problem formulation) されるとは限らない．
  1. ある工場で，品質管理のためにいくつかの工夫を行ったところ，統計的に明らかな効率改善がみられた．これは，定式化されない最適化の一種である．
  2. 熟練したプログラマーは，処理系の性質を考慮して，計算効率が良いようにプログラミングを行う．
- 最適化の定式化には様々な種類がある．いわゆる最適化問題のカテゴリーに属さないものも多い．
  1. プログラムをコンパイル時に最適化（効率化）することができる．
  2. アルゴリズムにおける最適性: $n$ 個のデータをソーティングする際，ヒープソートの計算量は $O(n \log n)$ である．しかもこれよりオーダーを低くすることはできない．よって，$O(n \log n)$ は最適なオーダーである．
  3. ニューラルネットワークなどの学習では，最適性の概念が explicit(明示的) あるいは implicit に使われている場合がよくある．ホップフィールドネットワークは前者の例であり，バックプロパゲーションによる学習は後者の例である．
- 最適化問題には，時間を表す変数が明示的にはいらない静的 (static) な最適化問題と時間を表す変数が明示的に含まれる動的 (dynamic) な最適化問題がある．ここでは，主に前者をとりあげる．動的な最適化問題は力学に応用される変分法や最適制御に関連し，別にとりあげるべき主題である．

## 1.1 記号と最適化問題の記法

以下，狭義の最適化問題に関する記述を行う．

### 記号

- $\boldsymbol{R}^n$: $n$ 次元ユークリッド空間 (他に $\boldsymbol{R}^m$, $\boldsymbol{R}^p$, *etc.*).

- $\boldsymbol{x}$: $\boldsymbol{R}^n$ の要素で $\boldsymbol{x} \in \boldsymbol{R}^n$ と表す．成分は $^\top$ を転置記号として，次のように表す.

$$\boldsymbol{x} = (x_1, x_2, \cdots, x_n)^\top = \begin{pmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{pmatrix}$$

- $f(\boldsymbol{x})$, $g(\boldsymbol{x})$: $\boldsymbol{x}$ の実数値関数 ($f(\boldsymbol{x}) \in \boldsymbol{R}, g(\boldsymbol{x}) \in \boldsymbol{R}$).（一般に，$h$ が空間 $X$ で定義され，空間 $Y$ に値をとる写像のとき，$h\colon X \to Y$ とかく．従って，$f\colon \boldsymbol{R}^n \to \boldsymbol{R}$.）$f(\boldsymbol{x})$ は後に述べる目的関数あるいは評価関数，$g(\boldsymbol{x})$ は制約条件の意味で用いられることが多い.

- $F(\boldsymbol{x}), G(\boldsymbol{x})$: ベクトルの値あるいは行列の値をとる関数 (これに対して $f(\boldsymbol{x}), g(\boldsymbol{x})$ はスカラー値).

- $\nabla f$: $f$ のグラジエント (gradient)，($\nabla$: ナブラ) —行ベクトルであることに注意する.

$$\nabla f = \left( \frac{\partial f}{\partial x_1}, \cdots, \frac{\partial f}{\partial x_n} \right)$$

- $\nabla^2 f$: $f$ のヘッセ行列 (Hessian matrix)—ラプラシアンではないので注意.

$$\nabla^2 f = \left( \frac{\partial^2 f}{\partial x_i \partial x_j} \right) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdot\ \cdot\ \cdot & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdot\ \cdot\ \cdot & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

- $\|\boldsymbol{x}\|$: $\boldsymbol{x} \in \boldsymbol{R}^n$ のユークリッドノルム.

$$\|\boldsymbol{x}\| \overset{\text{def}}{=} \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

($\overset{\text{def}}{=}$ は定義式を示す).

## 最適化問題の記法

次の 3 つの記法を扱う.

**(P1)**

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & (\text{または maximize} \quad f(\boldsymbol{x})) \\ \text{subject to} & g_i(\boldsymbol{x}) \le 0, \quad i = 1, \ldots, m & \\ & h_j(\boldsymbol{x}) = 0, \quad j = 1, \ldots, \ell. & \end{array}$$

制約条件 (constraints) $g_i(\boldsymbol{x}) \le 0,\ i = 1, \ldots, m,\ h_j(\boldsymbol{x}) = 0,\ j = 1, \ldots, \ell$ のもとで，目的関数 (objective function) $f(\boldsymbol{x})$ を最小化せよ（または最大化せよ）．

**(P2)** ：**(P1)** の変形その 1.

$$f(\boldsymbol{x}) \to \text{最小}$$

$$\text{条件：} \quad \begin{cases} g_i(\boldsymbol{x}) \le 0 & i = 1, \ldots, m \\ h_j(\boldsymbol{x}) = 0, & j = 1, \ldots, \ell \end{cases}$$

**(P3)** ：**(P1)** の変形その 2.

$$\min_{\boldsymbol{x} \in X} f(\boldsymbol{x})$$

$$X = \{\, \boldsymbol{x} \in \boldsymbol{R}^n \,:\, g_i(\boldsymbol{x}) \le 0,\, i = 1, \ldots, m;\, h_j(\boldsymbol{x}) = 0,\, j = 1, \ldots, \ell \,\}$$

$g_i(\boldsymbol{x}), h_j(\boldsymbol{x})$ を制約関数 (constraint function) という．目的関数は，評価関数と呼ばれることもある．**(P3)** の $X$ を実行可能領域 (feasible region) あるいは実行可能集合 (feasible set) と呼ぶ.

$X \neq \emptyset$ ($X$ は空集合でない) のとき，最適化問題は実行可能 (feasible) といい，$X = \emptyset$ ($X$ は空集合) のとき，最適化問題は実行不可能 (infeasible) という．$X$ の点を実行可能解 (feasible solution) という.

上の最適化問題の最適解 (optimal solution あるいは optimum solution) $\boldsymbol{x}^*$ は $\boldsymbol{x}^* \in X$ かつ

$$f(\boldsymbol{x}^*) \le f(\boldsymbol{x}), \quad \text{for all } \boldsymbol{x} \in X$$

をみたす点のことである．このような点が $X$ に存在するとき,「この最適化問題には最適解が存在する」という．また，最適解 $\boldsymbol{x}^*$ を

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \in X} f(\boldsymbol{x})$$

と書くことがある．最小化 (minimize) する要素 (argument) が $\boldsymbol{x}^*$ だからである (最大化の場合，$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x} \in X} f(\boldsymbol{x})$ ).

制約関数 $g_i, h_j$ などによる制約条件がない問題

$$\text{minimize} \quad f(\boldsymbol{x}), \qquad \boldsymbol{x} \in \boldsymbol{R}^n$$

あるいは

$$\min_{\boldsymbol{x} \in \boldsymbol{R}^n} f(\boldsymbol{x})$$

は制約のない最適化問題 (unconstrained optimization problem) と呼ばれる．制約がある場合は，制約付き最適化問題 (constrained optimization problem) と呼ぶ.

**問** (最適解の存在). 次の問題の最適解を求めよ．ない場合はなし，と記せ.

(i)

$$\begin{aligned} f(x) &= x^2 \to \min \\ g_1(x) &= x - 2 \leq 0 \\ g_2(x) &= -x + 1 \leq 0 \end{aligned}$$

(ii)

$$\begin{aligned} f(x) &= x^2 \to \max \\ g_1(x) &= x - 1 < 0 \\ g_2(x) &= -x \leq 0 \end{aligned}$$

(iii)

$$\begin{aligned} f(x) &= x^2 \to \min \\ g_1(x) &= x - 1 < 0 \\ g_2(x) &= -x \leq 0 \end{aligned}$$

(iv)

$$\begin{aligned} f(x) &= \frac{1}{x} \to \min \\ g_1(x) &= x - 1 \leq 0 \\ g_2(x) &= -x \leq 0 \end{aligned}$$

ただし，$x \neq 0$.

(iv)

$$\begin{aligned} f(x) &= \frac{1}{x} \to \min \\ g_1(x) &= x - 1 \leq 0 \\ g_2(x) &= -x - 1 \leq 0 \end{aligned}$$

ただし，$x \neq 0$.

## 大域的最適解と局所的最適解

先に述べた性質 ( $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}), \quad \text{for all } \boldsymbol{x} \in X)$ をもつ最適解 $\boldsymbol{x}^*$ は次に述べる局所的最適解と区別するために，大域的最適解 (global optimum solution) と呼ばれる．

ある解 $\tilde{\boldsymbol{x}}$ が局所最適解であるとは，$\tilde{\boldsymbol{x}}$ の充分近くに限れば，この解が最適であるということである．正確にいうと，次のようになる．

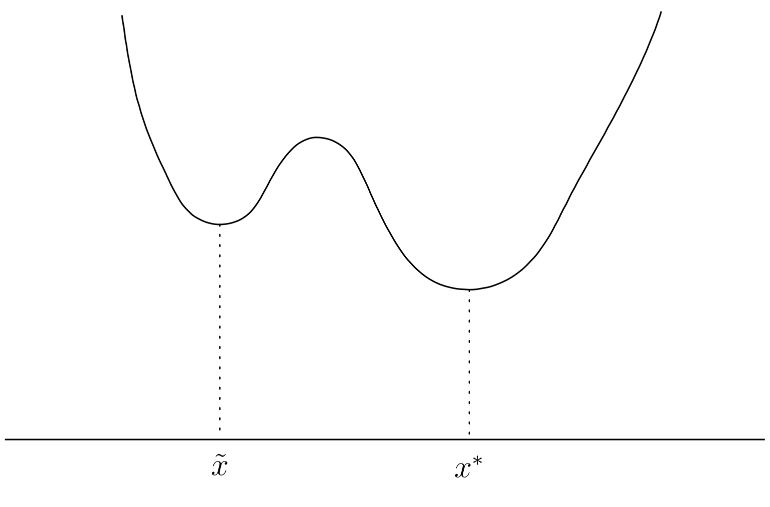
解 $\tilde{\boldsymbol{x}} \in X$ に対して，$\tilde{\boldsymbol{x}}$ の開近傍 $B(\tilde{\boldsymbol{x}}; r)$ ($\tilde{\boldsymbol{x}}$ を中心とする半径 $r > 0$ の開球 $B(\tilde{\boldsymbol{x}}; r) = \{\boldsymbol{x} \in \boldsymbol{R}^n : \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\| < r\}$ ) が存在して，

$$f(\tilde{\boldsymbol{x}}) \leq f(\boldsymbol{x}), \quad \text{for all } \boldsymbol{x} \in B(\tilde{\boldsymbol{x}}; r) \cap X$$

をみたすとき，$\tilde{\boldsymbol{x}}$ は局所最適解 (local optimum solution) であると呼ばれる．

下の図では，$\boldsymbol{x}^*$ は大域的最適解，$\boldsymbol{x}^*$ と $\tilde{\boldsymbol{x}}$ は局所最適解である．

**問** (局所最適解)．目的関数 $f(x) = \dfrac{1}{4}x^4 - \dfrac{5}{3}x^3 + 3x^2$ について $\displaystyle\min_{x \in \boldsymbol{R}} f(x)$ を考える．局所最適解をすべて求めよ．$r$ はどのようにとればよいか？



# 第2章　線形計画法

## 2.1　例題

まず，2つの例題を紹介しよう．

**例題P.**(生産計画問題 – Production planning problem)　ある工場では，2種類の原料 $R_1, R_2$ を使って2種類の製品 $P_1, P_2$ を生産している．次の条件がある．

(a) 全部で $R_1$ は20単位，$R_2$ は15単位使うことができる．

(b) $P_1$ を1単位生産するために，$R_1$ が2単位，$R_2$ が4単位必要である．また，$P_2$ を1単位生産するために，$R_1$ が5単位，$R_2$ が1単位必要である．

(c) $P_1$ について，1単位生産するごとに3単位の利益があり，$P_2$ について，1単位生産するごとに4単位の利益がある．

**総利益が最大となるようにするには，$P_1, P_2$ をそれぞれ何単位生産すればよいか？**

$P_1, P_2$ の生産量をそれぞれ $x_1, x_2$ 単位とする．

$$\begin{aligned} \text{総利益}: &\quad 3x_1 + 4x_2 \\ \text{資源の制約 1}: &\quad 2x_1 + 5x_2 \leq 20 \\ \text{資源の制約 2}: &\quad 4x_1 + x_2 \leq 15 \end{aligned}$$

よって，次の最適化問題を得る．

$$\begin{aligned} (\text{目的関数}) &\quad 3x_1 + 4x_2 \to \max && (2.1) \\ (\text{条件}) &\quad 2x_1 + 5x_2 \leq 20 && (2.2) \\ &\quad 4x_1 + x_2 \leq 15 && (2.3) \\ &\quad x_1 \geq 0, \quad x_2 \geq 0 && (2.4) \end{aligned}$$

**例題D.**(栄養問題 – Diet problem)　ある農場で飼っている家畜には，2種類の栄養素 $N_1, N_2$ が必要である．使用している飼料は $F_1, F_2$ の2種類である．次の条件がある．

(a') 全栄養素は $N_1$ について16単位，$N_2$ について24単位必要である．

(b') $F_1$ の 1 単位には，栄養素 $N_1$ が 4 単位，$N_2$ が 2 単位含まれている．また，$F_2$ の 1 単位には，栄養素 $N_1$ が 3 単位，$N_2$ が 7 単位含まれている．

(c') $F_1$ について，1 単位の価格は 4 単位であり，$F_2$ について，1 単位の価格は 5 単位である．

**栄養素 $1, 2$ の必要量を満たし，総コストが最小になる $F_1, F_2$ はそれぞれ何単位か？**

$F_1, F_2$ の量をそれぞれ $y_1, y_2$ 単位とする．

$$\begin{aligned} \text{総コスト}: &\quad 4y_1 + 5y_2 \\ \text{栄養素 1 の必要量}: &\quad 4y_1 + 3y_2 \geq 16 \\ \text{栄養素 2 の必要量}: &\quad 2y_1 + 7y_2 \geq 24 \end{aligned}$$

よって，次の最適化問題を得る．

$$\begin{aligned} (\text{目的関数}) &\quad 4y_1 + 5y_2 \to \min && (2.5) \\ (\text{条件}) &\quad 4y_1 + 3y_2 \geq 16 && (2.6) \\ &\quad 2y_1 + 7y_2 \geq 24 && (2.7) \\ &\quad y_1 \geq 0, \quad y_2 \geq 0 && (2.8) \end{aligned}$$

これらの問題は，第 1 章の問題 **(P1)**(**(P2)** でも同じ) において $f(\boldsymbol{x})$, $g_i(\boldsymbol{x})$ が線形の関数である場合に相当する．後で一般的に述べるが，このように，目的関数と制約条件が線形の関数のとき，最適化問題を線形計画問題とよぶ．

## 図的解法

例題 P をグラフを描くことによって解いてみよう．制約条件 (2.2), (2.3), (2.4) を $x_1, x_2$–軸上に描くと，図 2.1 の領域 $X$ が得られる．また，目的関数は同じ図上で動く直線 $\ell$ となり，目的関数の値は $x_2$ 切片に比例する．

直線 $\ell$ が $X$ の頂点 $O, A, B, C$ を通る場合を調べればよい．$O, A, B$ と動くに従って，$x_2$ 切片は増加する．また，$B$ から $C$ に移動すると目的関数は減少するので $B$ で最適となるのは明らかである．

図をみて解く方法は変数の数が多くなるとすぐに使えなくなる．従って，形式的な解法が必要である．

いま，式 (2.2),(2.3) に補助的な変数 (後で述べるスラック変数) $x_3$, $x_4$ ($x_3 \geq 0$, $x_4 \geq 0$) を追加し，(2.2),(2.3) を等式に置き換える．いいかえれば，(2.2),(2.3) の

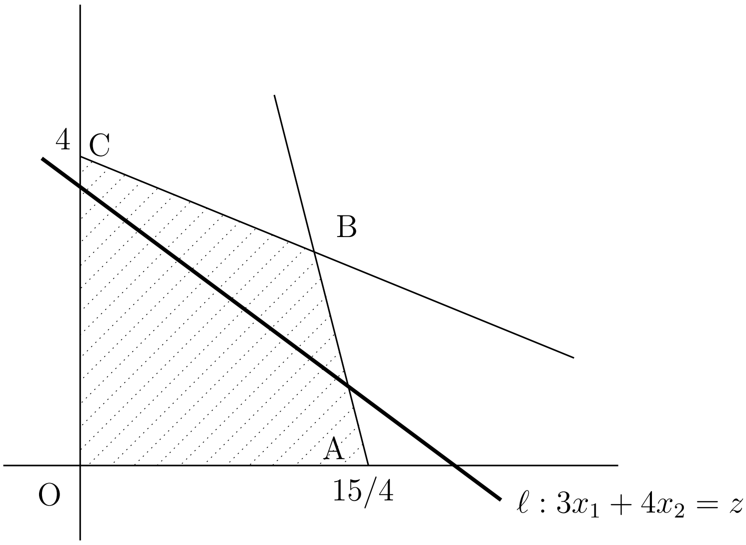



図 2.1: 例題 P の解を示す図

右辺と左辺との差を変数 $x_3$, $x_4$ で代表させる．したがって，

$$
\begin{aligned}
&(\text{目的関数}) && 3x_1 + 4x_2 + 0x_3 + 0x_4 \to \max && (2.9)\\
&(\text{条件}) && 2x_1 + 5x_2 + x_3 \qquad = 20 && (2.10)\\
& && 4x_1 + x_2 \qquad + x_4 = 15 && (2.11)\\
& && x_1, x_2, x_3, x_4 \geq 0 && (2.12)
\end{aligned}
$$

次のことに注意しよう．

1. 原点 $O$ では，$x_1 = x_2 = 0$ となり，$x_3 = 20$, $x_4 = 15$ と補助変数は右辺に等しい．
2. 頂点 $A$ では，$x_2 = x_4 = 0$ であり，$x_1$ と $x_3$ は正である．
3. 頂点 $B$ では，$x_3 = x_4 = 0$ であり，$x_1$ と $x_2$ は正である．

要するに，頂点が移動するごとに，ゼロの値をとる変数と正の値をとる変数が一つずつ入れ替わっている．このように，

- ゼロになる変数 (非基底変数) と正の変数 (基底変数) を一つずつ入れ替え，
- 入れ替えるごとに目的関数が減少（あるいは増加）するようにし，
- 目的関数がそれ以上減少（あるいは増加）させられるかどうか判定を行う

アルゴリズムが線形計画法では用いられる．このアルゴリズムによる解法をシンプレックス解法と呼ぶ（後述）．

**問.** 栄養問題を図的解法によって解け．

## 2.2 標準形

この節では，一般的な形で線形計画問題を記述し，標準形の問題を導入する．

問題 **(P1)**

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && g_i(\boldsymbol{x}) \le 0, \quad i = 1, \dots, m \\ & && h_k(\boldsymbol{x}) = 0, \quad k = 1, \dots, \ell. \end{aligned}$$

において，目的関数と制約条件が線形の場合，この問題を線形計画問題 (linear programming problem) という．この問題を一般形と呼んでおこう．

従って，線形計画問題は次の形をしている．

$$\sum_{j=1}^{n} c_j x_j \to \min \tag{2.13}$$

$$\sum_{j=1}^{n} g_{ij} x_j \le \alpha_i, \quad i = 1, \dots, m \tag{2.14}$$

$$\sum_{j=1}^{n} h_{kj} x_j = \beta_k, \quad k = 1, \dots, \ell. \tag{2.15}$$

### 標準形の問題

次の形の問題を標準形の線形計画問題という．

$$\sum_{j=1}^{n} c_j x_j \to \min \tag{2.16}$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i = 1, \dots, m \tag{2.17}$$

$$x_j \ge 0, \quad j = 1, \dots, n. \tag{2.18}$$

ベクトル $\boldsymbol{x} = (x_1, \cdots, x_n)^\top$, $\boldsymbol{b} = (b_1, \cdots, b_m)^\top$, $\boldsymbol{c} = (c_1, \cdots, c_n)^\top$, および $m \times n$ 行列

$$A = (a_{ij}) = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \cdot & \cdots & \cdot \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$$

を用いると，(2.16)–(2.18) は次の形に書かれる．

$$\boldsymbol{c}^\top \boldsymbol{x} \to \min \tag{2.19}$$

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.20}$$

$$\boldsymbol{x} \ge 0 \tag{2.21}$$

ここで，$\boldsymbol{x} \geq 0$ はベクトル $\boldsymbol{x}$ のすべての成分が非負という意味である．

明らかに，標準形は一般形 (2.13)–(2.15) よりも特殊な形をしている．しかしながら，適当な変換を行うことによって，一般形を標準形に帰着させることができる．

まず，一般形における変数 $x_k$ は必ずしも非負の制約条件 ($x_k \geq 0$) をもつとは限らない．$x_k$ に非負の条件がないときは，2 つの非負の変数 $x_k^+$, $x_k^-$ の差で $x_k$ を表す．すなわち，$x_k = x_k^+ - x_k^-$ ($x_k^+ \geq 0$, $x_k^- \geq 0$).

次に，不等式制約

$$\sum_{j=1}^{n} g_{ij} x_j \leq \alpha_i$$

については，補助変数 $z_i \geq 0$ を導入し，

$$\sum_{j=1}^{n} g_{ij} x_j + z_i = \alpha_i$$

とすれば，等式制約に変形できる．

このように，一般形は標準形に帰着されることがわかった．

特殊な場合ではあるが，一般的な生産計画問題と栄養問題をあらためて記しておこう．

**生産計画問題 P.** 原料 $R_1, \cdots, R_m$ を利用して製品 $P_1, \cdots, P_n$ を生産する．

(a) 原料 $R_i$ ($i = 1, \ldots, m$) は $b_i$ 単位まで利用できる．

(b) 製品 $P_j$ を 1 単位生産するためには，原料 $R_i$ は $a_{ij}$ 単位必要である．

(c) 製品 $P_j$ を 1 単位生産することによって，利益 $c_j$ 単位が得られる．

$P_j$ の生産量を $x_j$ 単位として，総利益を最大化する問題は，次の線形計画問題になる．

$$\sum_{j=1}^{n} c_j x_j \to \max \tag{2.22}$$

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, \ldots, m \tag{2.23}$$

$$x_j \geq 0, \quad j = 1, \ldots, n \tag{2.24}$$

行列形式では，

$$\boldsymbol{c}^\top \boldsymbol{x} \to \max \tag{2.25}$$

$$A\boldsymbol{x} \leq \boldsymbol{b} \tag{2.26}$$

$$\boldsymbol{x} \geq \boldsymbol{0} \tag{2.27}$$

この問題を標準形に変換するためには，上に述べたように，変数 $z_i \geq 0$ を導入し，

$$\begin{aligned}&\sum_{j=1}^{n} c_j x_j \to \max\\&\sum_{j=1}^{n} a_{ij} x_j + z_i = b_i, \quad i = 1, \ldots, m\\&x_j \geq 0, \quad j = 1, \ldots, n\end{aligned}$$

とする．行列形式では $\boldsymbol{z} = (z_1, \cdots, z_m)^\top$ として

$$\boldsymbol{c}^\top \boldsymbol{x} \to \max \quad (-\boldsymbol{c}^\top \boldsymbol{x} \to \min) \tag{2.28}$$

$$A\boldsymbol{x} + \boldsymbol{z} = \boldsymbol{b} \tag{2.29}$$

$$\boldsymbol{x} \geq \boldsymbol{0}, \quad \boldsymbol{z} \geq \boldsymbol{0} \tag{2.30}$$

このときの変数 $z_i$ をスラック変数と呼ぶ.

**栄養問題 D.** 栄養素 $N_1, \cdots, N_n$ を含む飼料 $F_1, \cdots, F_m$ を家畜に与える.

(a') 全栄養素は $N_j (j = 1, \ldots, n)$ について $c_j$ 単位必要である.

(b') $F_i$ の 1 単位には，栄養素 $N_j$ が $a_{ij}$ 単位含まれている.

(c') $F_i$ について，1 単位の価格は $b_i$ 単位である.

飼料 $F_i$ を $y_i$ 単位与えるものとし，このとき，栄養素の必要最低量を確保しつつ，コストを最小にする問題は，次のように書かれる.

$$\sum_{i=1}^{m} b_i y_i \to \min \tag{2.31}$$

$$\sum_{i=1}^{m} a_{ij} y_i \geq c_j, \quad j = 1, \ldots, n \tag{2.32}$$

$$y_i \geq 0, \quad i = 1, \ldots, m \tag{2.33}$$

行列形式では,

$$\boldsymbol{b}^\top \boldsymbol{y} \to \min \tag{2.34}$$

$$\boldsymbol{y}^\top A \geq \boldsymbol{c}^\top \quad (A^\top \boldsymbol{y} \geq \boldsymbol{c}) \tag{2.35}$$

$$\boldsymbol{y} \geq \boldsymbol{0} \tag{2.36}$$

となる.

標準形に変換するには，変数 $w_j \geq 0$ を導入し，

$$\begin{aligned}
&\sum_{i=1}^{m} b_i y_i \to \min \\
&\sum_{j=1}^{m} a_{ij} y_i - w_j = c_j, \quad j = 1, \ldots, n \\
&y_i \geq 0, \quad i = 1, \ldots, m
\end{aligned}$$

とする．行列形式では，$\boldsymbol{w} = (w_1, \cdots, w_n)^\top$ によって，

$$\begin{aligned}
&\boldsymbol{b}^\top \boldsymbol{y} \to \min \\
&\boldsymbol{y}^\top A - \boldsymbol{w}^\top = \boldsymbol{c}^\top \qquad (A^\top \boldsymbol{y} - \boldsymbol{w} = \boldsymbol{c}) \\
&\boldsymbol{y} \geq \boldsymbol{0}, \quad \boldsymbol{w} \geq \boldsymbol{0}
\end{aligned}$$

**問**．例題 2.1 を行列形式で書け．$A$, $\boldsymbol{b}$, $\boldsymbol{c}$ はどうなるか．

**注**．生産計画問題は最も解きやすいため，標準形とならんでよくとり上げられる．栄養問題は後でみるように，生産計画問題の双対問題である．上の記号の用法は双対性を示すためのものである．

## 2.3 線形計画法の基本定理

この節では，標準形について，シンプレックス法の基礎となる基本定理 [1] を述べる．主に，行列形式

$$\begin{aligned}
&\boldsymbol{c}^\top \boldsymbol{x} \to \min \\
&A\boldsymbol{x} = \boldsymbol{b} \\
&\boldsymbol{x} \geq \boldsymbol{0}
\end{aligned}$$

によって記述する ($\boldsymbol{c}^\top \boldsymbol{x}$:目的関数，$A\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$:制約条件，に注意)．

**仮定**． $A$ は横長 ($n \geq m$) の行列である．また，$A$ の階数 (ランク) は $m$ である ($\operatorname{rank} A = m$)．

階数の定義によって，この行列 $A$ には少なくとも 1 組の $m$ 個の列ベクトル $\boldsymbol{\alpha}_{j_1}, \cdots, \boldsymbol{\alpha}_{j_m}$ が存在してこの組は $m$ 次元の空間 $\boldsymbol{R}^m$ の基底となる．いいかえれば，すべての $m$-ベクトル $\boldsymbol{d} \in \boldsymbol{R}^m$ に対して適当にスカラー $z_1, \cdots, z_m \in \boldsymbol{R}$ をとり，

$$z_1 \boldsymbol{\alpha}_{j_1} + \cdots + z_m \boldsymbol{\alpha}_{j_m} = \boldsymbol{d}$$

とすることができる．同じことであるが，この列ベクトルをならべてできる行列 $\Gamma = (\boldsymbol{\alpha}_{j_1}, \cdots, \boldsymbol{\alpha}_{j_m})$ は正則である (別の言い方では，(1) 逆行列 $\Gamma^{-1}$ が存在する; (2) 行列式がゼロでない: $\det \Gamma \neq 0$)．

上に述べた基底により構成される行列 $\Gamma = (\boldsymbol{\alpha}_{j_1}, \cdots, \boldsymbol{\alpha}_{j_m})$ を基底行列 (basis matrix) という．一般に，基底の選び方には自由度があるので，基底行列も一意的ではない．また，便宜的に，$(1, \ldots, n)$ のうち，基底に対応する添字 $(j_1, \cdots, j_m)$ を除いた残りの添字を $(k_1, \cdots, k_{n-m})$ とする．

いま，$A\boldsymbol{x} = \boldsymbol{b}$ において，変数 $\boldsymbol{x} = (x_1, \cdots, x_n)$ において基底に対応しない部分をゼロとする $(x_{k_1} = \cdots = x_{k_{n-m}} = 0)$．また，$\boldsymbol{z} = (x_{j_1}, \cdots, x_{j_m})^\top$ とおく．このとき，$A\boldsymbol{x} = \boldsymbol{b}$ は $\Gamma \boldsymbol{z} = \boldsymbol{b}$ に他ならない．また，仮定より $\Gamma$ は正則だから

$$\boldsymbol{z} = \Gamma^{-1}\boldsymbol{b} \tag{2.37}$$

と解くことができる．これに残りの成分 $x_{k_1} = \cdots = x_{k_{n-m}} = 0$ を合わせると，$A\boldsymbol{x} = \boldsymbol{b}$ を満足する一つの解が得られた．このように，ある基底行列に対応しない変数 (非基底変数–nonbasic variable) をゼロとし，基底に対応する変数 (基底変数–basic variable) についてはこのように解を求めて得られた解を基底解 (basic solution) という．

ところが，一般に，(2.37) によって得られた解は非負とはいえないので，実行可能とは限らない．もし，(2.37) によって得られた解の各成分が非負ならば，これに対応する基底解も非負 ($\boldsymbol{x} \geq \boldsymbol{0}$) となるので，実行可能 (feasible) である．このとき，基底解を実行可能基底解 (basic feasible solution–bfs と略す) と呼ぶ．

したがって，実行可能基底解とは，$m$ 個の基底変数が非負であり，非基底変数はすべてゼロであるような $A\boldsymbol{x} = \boldsymbol{b}$ の解である．さらに，基底変数の正の成分の数がちょうど $m$ のとき，この解は非退化 (nondegenerate) と呼ばれる．

これで，線形計画法の基本定理を述べる準備ができた．

**基本定理．** 線形計画法の標準問題について，次の **(A)**, **(B)** が成立する．

**(A)** 実行可能解が存在すれば，実行可能基底解も存在する．

**(B)** 最適解が存在すれば，実行可能基底解のなかに最適解が存在する．(注：最適実行可能基底解—optimal basic feasible solution—optimal bfs)

以下，定理の証明を行う．

**(A):** 方針として，実行可能解から正の成分を 1 つずつ減らし，正の成分が $m$ になるまで減らすことができることを示す．

$\hat{\boldsymbol{x}} \geq \boldsymbol{0}$ を一つの実行可能解とする．簡単のため，変数の番号を適宜入れ替えたと考えて，$\hat{\boldsymbol{x}}$ の正の成分が $\hat{x}_1, \cdots, \hat{x}_q$ $(q \leq n)$ であると仮定する ($\hat{\boldsymbol{x}} = (\hat{x}_1, \cdots, \hat{x}_q, 0, \cdots, 0)$)．$q \leq m$ ならば，$\hat{\boldsymbol{x}}$ はすでに実行可能基底解であるから，$q > m$ と仮定する．$A$ の第 $j$ 列ベクトルを $\boldsymbol{\alpha}_j$ と書くと

$$\hat{x}_1\boldsymbol{\alpha}_1 + \cdots + \hat{x}_q\boldsymbol{\alpha}_q = \boldsymbol{b}.$$

$q > m$ であるから，$\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_q$ は一次従属であり，すべてはゼロではないスカラー $w_1, \cdots, w_q$ を適当にとると，

$$w_1\boldsymbol{\alpha}_1 + \cdots + w_q\boldsymbol{\alpha}_q = \mathbf{0}$$

この式に $-\varepsilon$ をかけて，その上の式に加えると，

$$(\hat{x}_1 - \varepsilon w_1)\boldsymbol{\alpha}_1 + \cdots + (\hat{x}_q - \varepsilon w_q)\boldsymbol{\alpha}_q = \boldsymbol{b}.$$

仮定より，充分小さな $\varepsilon$ に対して上の等式の左辺の係数 $\hat{x}_k - \varepsilon w_k$ $(k = 1, \ldots, q)$ はすべて正であり，従って実行可能である．そこで，徐々に $\varepsilon$ を増加させていくと，はじめに係数がゼロとなる係数がある．この係数が $r$ 番目であるとする．ゼロとなるときの $\varepsilon$ は $\varepsilon = \hat{x}_r/w_r$ であり，かつ，$\hat{x}_k - \varepsilon w_k \geq 0$ $(k = 1, \ldots, q)$ が成立している．よって，

$$\varepsilon = \min_{\substack{1 \leq k \leq q \\ w_k > 0}} \frac{\hat{x}_k}{w_k}$$

と選び，$\tilde{x}_k = \hat{x}_k - \varepsilon w_k$ とおけば，

$$\tilde{x}_1\boldsymbol{\alpha}_1 + \cdots + \tilde{x}_{r-1}\boldsymbol{\alpha}_{r-1} + \tilde{x}_{r+1}\boldsymbol{\alpha}_{r+1} + \cdots + \tilde{x}_q\boldsymbol{\alpha}_q = \boldsymbol{b}.$$

すなわち，正の係数が $q-1$ 個の解が得られた．$m = q-1$ ならば **(A)** が証明されたことになる．$m < q-1$ ならば，上の手続きを繰り返せばよい．

**(B):** **(A)** の証明で仮定した $\hat{x}$ が最適解であるとする．これが基底解でないとすると正の成分は $\hat{x}_1, \cdots, \hat{x}_q$ $(q > m)$ である．**(A)** の証明の手続きに従い，

$$(\hat{x}_1 - \varepsilon w_1)\boldsymbol{\alpha}_1 + \cdots + (\hat{x}_q - \varepsilon w_q)\boldsymbol{\alpha}_q = \boldsymbol{b}.$$

とし，$\boldsymbol{w} = (w_1, \cdots, w_q, 0, \cdots, 0)^\top$，

$$\hat{\boldsymbol{x}} - \varepsilon \boldsymbol{w} = (\hat{x}_1 - \varepsilon w_1, \cdots, \hat{x}_q - \varepsilon w_q, 0, \cdots, 0)^\top$$

とおく．$\varepsilon$ の正負にかかわらず，$|\varepsilon|$ が充分小さい場合，$\hat{\boldsymbol{x}} - \varepsilon \boldsymbol{w}$ は実行可能解である．しかも，$\hat{\boldsymbol{x}}$ は最適であるから，任意の $\varepsilon$ に対して

$$\boldsymbol{c}^\top \hat{\boldsymbol{x}} \leq \boldsymbol{c}^\top (\hat{\boldsymbol{x}} - \varepsilon \boldsymbol{w}).$$

これから，$\varepsilon \boldsymbol{c}^\top \boldsymbol{w} \leq 0$ であるが，$\varepsilon$ は正でも負でも良いから結局 $\boldsymbol{c}^\top \boldsymbol{w} = 0$ を得る．これに **(A)** の証明の手続きをあわせると

$$\boldsymbol{c}^\top \tilde{\boldsymbol{x}} = \boldsymbol{c}^\top \hat{\boldsymbol{x}}$$

となり，$\hat{\boldsymbol{x}}$ が最適なら $\tilde{\boldsymbol{x}}$ も最適であることがわかる．この手続きを $q = m$ となるまで繰り返すことによって，最適な基底解を見いだすことができる．**（証明終）**

### 2.3.1　基本定理と幾何学的考察

$A\boldsymbol{x} = \boldsymbol{b}$ の一つの行

$$a_{i1}x_1 + \cdots + a_{in}x_n = b_i$$

をみたす $x \in \boldsymbol{R}^n$ はこの空間のなかで一つのアファイン部分空間（一般に原点を通らない超平面）を構成する．したがって，$m$ 個の行をみたす $x$ の集合はこれらの部分空間の交わりであり，しかも第一象限 $(\boldsymbol{x} \geq \boldsymbol{0})$ に限定されている．この集合 $X$ が実行可能解の集合に他ならないが，$X$ は有界（充分大きな超立方体に含まれる）かも知れないし，あるいは非有界かも知れない．目的関数を表すアファイン部分空間 $\boldsymbol{c}^\top\boldsymbol{x} = L$ を $L$ を変えて移動させ，この部分空間と $X$ との共通部分が空でなく，かつ $L$ が最も小さくなる点が最適解となる．$X$ が非有界の場合，最適解が存在しないことも考えられる．

基底解はこれらの部分空間の交わりがなす多面体の頂点である．なぜなら，点 $\hat{\boldsymbol{x}}$ が頂点でない場合，$\hat{\boldsymbol{x}}$ を通る直線をひいて，$\hat{\boldsymbol{x}}$ をその直線にそって両方向に少し移動しても多面体の外部に出ることはない．これに対して点 $\hat{\boldsymbol{x}}$ が頂点ならば，どんな直線をひいたとしても，その直線にそって移動した点が多面体のなかにとどまるためには，直線にそって片方にしか動かすことができない．この操作は，先の証明における $\varepsilon$ の議論に対応している．

さて，基本定理が述べているのは，**(A)** は $X$ に頂点があるということであり，**(B)** は $X$ において超平面 $\boldsymbol{c}^\top\boldsymbol{x} = L$ を動かすことによって，頂点で最適とすることができるということである．

このように，基本定理によって述べられているのは，幾何学的には明らかなことである．

## 2.4　シンプレックス法

シンプレックス法は，基本定理によって保障された実行可能基底解（すなわち，$X$ の頂点）を辿ることによって，最適解に到達する方法で，次の特徴がある．

- ある実行可能基底解が最適か最適でないかを簡単に判定する方法がある．最適でないとき，別の実行可能基底解に移る．
- 別の実行可能基底解に移るとき，1 つの基底変数が非基底変数となり，他の 1 つの非基底変数が基底変数になる．
- このとき，目的関数の値は減少する．

実行可能基底解の組は有限個であるので，上の特徴から，実行可能基底解から別の実行可能基底解に移る手続きは有限回で終了する．

シンプレックス法は，シンプレックスタブローと呼ばれる表を用いて実行される．まず，例題 **P** を用いて説明しよう．

$$z = -3x_1 - 4x_2 + 0x_3 + 0x_4 \to \min \tag{2.38}$$
$$2x_1 + 5x_2 + x_3 \qquad = 20 \tag{2.39}$$
$$4x_1 + x_2 \qquad + x_4 = 15 \tag{2.40}$$
$$x_1, x_2, x_3, x_4 \geq 0 \tag{2.41}$$

初期解を $x_1 = x_2 = 0$ とすると，$x_3$, $x_4$ は右辺に等しく，$x_3 = 20$, $x_4 = 15$ となる．これは一つの bfs(実行可能基底解) であり，基底変数は $x_3$, $x_4$，非基底変数は $x_1$, $x_2$ である．

これを下の表の形に書く．この表をシンプレックスタブローという．

| 基底 | $x_1$ | $x_2$ | $x_3$ | $x_4$ | 定数 |
|---|---|---|---|---|---|
| $-z$ | $-3$ | $-4$ | 0 | 0 | 0 |
| $x_3$ | 2 | 5 | 1 | | 20 |
| $x_4$ | 4 | 1 | | 1 | 15 |

この解は最適ではない．なぜなら，目的関数において非基底変数に対応する係数 $(-3, -4)$ が負の数を含むので，非基底変数を $0 \to +$ と増加させることによって目的関数の値を減少させることができる．

非基底変数を増加させた分，基底変数のどれかを減少させなければならない．bfs から bfs に移るとは，ある非基底変数を + とすることで，別の基底変数を 0 にすることである．この例では $x_1$ を増加させ，$x_3$ あるいは $x_4$ を 0 にする．

$x_2$ は 0 のままにするので，(2.39) と (2.40) から

$$\begin{aligned} x_1 &= \frac{1}{2}(20 - x_3) \\ x_1 &= \frac{1}{4}(15 - x_4) \end{aligned}$$

はじめの式で $x_3 = 0$ とすると $x_1 = 10$, 後の式で $x_4 = 0$ とすると $x_1 = \frac{15}{4}$ だが，$x_1 = 10$ のときは $x_4 < 0$ となり実行可能でない．よって，$x_1 = 10$ と $x_1 = \frac{15}{4}$ の小さいほうを選ぶ．この操作によって，シンプレックスタブローは次のように計算される．

計算の方法としては，目的関数におけるマイナスの係数 (アンダーライン) を選び，その列の正の数でそれぞれの右辺をわった値を調べる．そのうちの最小値（ここでは $\min\{20/2, 15/4\} = 15/4$）を四角で囲んである．この値でこの行をわり，四

角で囲んだ数値を 1 にする．これに対応する変数を基底変数として計算を続行するにはこの変数についての他の行を消去すればよい．

そのため，下側の表の第 2 行に 2 をかけて上側の第 1 行から引き算を行い，結果を下側の第 1 行目とする．目的関数についても同様に下側の表の第 2 行に 3 をかけて上側の目的関数の行に加えることによって，下側の目的関数の行を得る．

| 基底 | $x_1$ | $x_2$ | $x_3$ | $x_4$ | 定数 |
|---|---|---|---|---|---|
| $-z$ | $\underline{-3}$ | $-4$ | 0 | 0 | 0 |
| $x_3$ | 2 | 5 | 1 | | 20 |
| $x_4$ | $\boxed{4}$ | 1 | | 1 | 15 |
| $-z$ | | $-\frac{13}{4}$ | | $\frac{3}{4}$ | $\frac{45}{4}$ |
| $x_3$ | | $\frac{9}{2}$ | 1 | $-\frac{1}{2}$ | $\frac{25}{2}$ |
| $x_1$ | 1 | $\frac{1}{4}$ | | $\frac{1}{4}$ | $\frac{15}{4}$ |

このとき，基底変数は $x_1$, $x_3$ であり，非基底変数は $x_2$, $x_4$ である．また，シンプレックスタブローは制約条件と目的関数が次のように変形されることを意味している．

$$z = \qquad -\frac{13}{4}x_2 + \qquad \frac{3}{4}x_4 - \frac{45}{4} \tag{2.42}$$

$$+\frac{9}{2}x_2 + x_3 - \frac{1}{2}x_4 = \frac{25}{2} \tag{2.43}$$

$$x_1 + \frac{1}{4}x_2 \qquad + \frac{1}{4}x_4 = \frac{15}{4} \tag{2.44}$$

$x_2$ の係数 $-\frac{13}{4}$ が負なので，$x_2 = 0 \to +$ として目的関数をさらに減少させることができる．(右辺)/（係数）の最小値は

$$\min\left\{\frac{25}{2}\Big/\frac{9}{2}, \frac{15}{4}\Big/\frac{1}{4}\right\} = \frac{25}{2}\Big/\frac{9}{2} = \frac{25}{9}$$

で，このとき，$x_2 \to$ 基底, $x_3 \to$ 非基底 となる．

シンプレックスタブローは次のように計算される．計算の方法は先に示したのと同じである．

| 基底 | $x_1$ | $x_2$ | $x_3$ | $x_4$ | 定数 |
|---|---|---|---|---|---|
| $-z$ | $\underline{-3}$ | $-4$ | $0$ | $0$ | $0$ |
| $x_3$ | $2$ | $5$ | $1$ | | $20$ |
| $x_4$ | $\boxed{4}$ | $1$ | | $1$ | $15$ |
| $-z$ | | $\underline{-\frac{13}{4}}$ | | $\frac{3}{4}$ | $\frac{45}{4}$ |
| $x_3$ | | $\boxed{\frac{9}{2}}$ | $1$ | $-\frac{1}{2}$ | $\frac{25}{2}$ |
| $x_1$ | $1$ | $\frac{1}{4}$ | | $\frac{1}{4}$ | $\frac{15}{4}$ |
| $-z$ | | | $\frac{13}{18}$ | $\frac{7}{18}$ | $\frac{365}{18}$ |
| $x_2$ | | $1$ | $\frac{2}{9}$ | $-\frac{1}{9}$ | $\frac{25}{9}$ |
| $x_1$ | $1$ | | $-\frac{1}{18}$ | $\frac{5}{18}$ | $\frac{55}{18}$ |

条件と目的関数は次のようになる．

$$z = \qquad \frac{13}{18}x_3 + \frac{7}{18}x_4 - \frac{365}{18} \tag{2.45}$$

$$+\, x_2 + \frac{2}{9}x_3 - \frac{1}{9}x_4 = \frac{25}{9} \tag{2.46}$$

$$x_1 \qquad - \frac{1}{18}x_3 + \frac{5}{18}x_4 = \frac{55}{18} \tag{2.47}$$

この解は最適である．なぜなら，非基底変数に対応する目的関数の係数がすべて正なので，非基底変数を増加させると目的関数は増加する．

最適解は，基底変数が右辺の値 $x_1 = \frac{55}{18}$, $x_2 = \frac{25}{9}$，非基底変数はゼロで，最適値 $z = -\frac{365}{18}$ はシンプレックスタブローの目的関数の行の右に現れる．

スラック変数を用いた場合のシンプレックスタブローの計算法をまとめると次のようになる．

**(0)** 与えられた問題をシンプレックスタブローとして表す．目的関数の右辺の項としてゼロを代入しておく．

**(i)** 非基底変数に対応する目的関数の係数に負のものがあるかどうか調べる．なければ現在の解が最適解．目的関数の右辺の項が最適値となる．あれば，負の係数に対応する非基底変数 $x_s$ を一つ選ぶ．

**(ii)** $x_s$ に対応する列の各要素（ただし正のもの）で，右辺の数値をわり，その結果のうち最小の値に対応する要素をピボット項（先のタブローで四角で囲んだ要素）とする（最小の値でないと実行可能にならない）．

**(iii)** ピボットに対応する行をピボット項の値でわる．

(iv) 他の行について，ピボットに対応する列の要素がゼロとなるようにピボット行を加えるか減じる．目的関数についても同様の操作を行う．(i) に戻る．

シンプレックスタブローにおいて，

- 基底変数については目的関数の係数はゼロ，変数の値は右辺に等しい．
- 非基底変数については変数の値はゼロ，目的関数の係数はゼロとは限らない．

なお，係数が負のものが複数あるとき，どれを選ぶのが良いかは一概にはいえないが，負の値が最も大きいものを選ぶ，変数の添字の値の小さいものから選ぶ，などの方法がある．

**問．** 次の線形計画問題にスラック変数を導入して，シンプレックス法を適用し，最適解を求めよ．

(1)

$$
\begin{aligned}
z &= -x_1 - x_2 \to \min \\
& 3x_1 + 2x_2 \leq 12 \\
& x_1 + 2x_2 \leq 8 \\
& x_j \geq 0, \quad j = 1, 2
\end{aligned}
$$

(2)

$$
\begin{aligned}
z &= -x_1 - 2x_2 \to \min \\
& 2x_1 + 6x_2 \leq 27 \\
& 8x_1 + 6x_2 \leq 45 \\
& 3x_1 + x_2 \leq 15 \\
& x_j \geq 0, \quad j = 1, 2
\end{aligned}
$$

**問．** 次の問題を線形計画問題として定式化し，解を求めよ (福島 [2] p.21)
農業を営む林さんは１０ヘクタールの芋畑を所有しているが，来年はその一部をかぼちゃとなすの畑に転用することを考えている．現在の芋畑をかぼちゃの畑にするには当座の資金として１ヘクタールあたり５万円が，なすの畑にするには当座の資金として１ヘクタールあたり８万円が必要であり，来年も芋を作るには１ヘクタールあたり２万円の当座の資金があれば十分である．現在，当座の資金として総額４０万円しかもっていないので，１０ヘクタールの芋畑を分割してこれらを栽培することにしたい．それぞれによる純利益を予測したところ,「芋」,「かぼちゃ」,「なす」について１ヘクタールあたりそれぞれ２０万円，３０万円，６０万円であることがわかった．林さんが最大の利益を得るにはどのように分割すればよいか．

シンプレックス法の操作を一般的に表すと次のようになる．

$$\begin{array}{rcl}
-z + c_1x_1 + \cdots + c_nx_n & = & b_0 \\
a_{11}x_1 + \cdots + a_{1n}x_n + x_{n+1} \qquad\qquad & = & b_1 \\
a_{21}x_1 + \cdots + a_{2n}x_n \qquad + x_{n+2} \qquad & = & b_2 \\
\cdots\cdots\cdots\cdots\cdots\cdots & & \cdot\cdot \\
a_{m1}x_1 + \cdots + a_{mn}x_n \qquad\qquad + x_{n+m} & = & b_m
\end{array}$$

において $c_s < 0$ とする．$x_s$ を非基底変数から基底変数に変えることにする．このとき，実行可能性を保ちながら $x_s$ の代わりに非基底となる変数をみいだすには，

$$\frac{b_r}{a_{rs}} = \min_{\substack{1 \le i \le m \\ a_{is} > 0}} \frac{b_i}{a_{is}} \tag{2.48}$$

を求め，$x_{n+r}$ を非基底とすればよい．このとき，新たな第 $r$ 行目は

$$\bar{a}_{rj} = \frac{a_{rj}}{a_{rs}}, \quad j = 1, \ldots, n+m \tag{2.49}$$

$$\bar{b}_r = \frac{b_r}{a_{rs}} \tag{2.50}$$

他の行は

$$\bar{a}_{ij} = a_{ij} - \bar{a}_{rj}a_{is}, \quad j = 1, \ldots, n+m \tag{2.51}$$

$$\bar{b}_i = b_i - \bar{b}_r a_{is} \tag{2.52}$$

$$i = 1, \ldots, m \ (i \neq r) \tag{2.53}$$

目的関数については

$$\bar{c}_j = c_j - \bar{a}_{rj}c_s, \quad j = 1, \ldots, n+m \tag{2.54}$$

$$\bar{b}_0 = b_0 - \bar{b}_r c_s \tag{2.55}$$

と計算する．

シンプレックスタブローでは次のようになる．この計算を繰り返すことで最適解が得られる．

| 基底 | $x_1$ | $\cdots$ | $x_s$ | $\cdots$ | $x_n$ | | $x_{n+r}$ | $\cdots$ | | 定数 |
|---|---|---|---|---|---|---|---|---|---|---|
| $-z$ | $c_1$ | $\cdots$ | $c_s$ | $\cdots$ | $c_n$ | | $\cdots$ | $\cdots$ | | $b_0$ |
| $x_{n+1}$ | $a_{11}$ | $\cdots$ | $a_{1s}$ | $\cdots$ | $a_{1n}$ | 1 | | | | $b_1$ |
| $\vdots$ | $a_{21}$ | $\cdots$ | $a_{2s}$ | $\cdots$ | $a_{2n}$ | | $\ddots$ | | | $b_2$ |
| $\vdots$ | $\cdots$ | $\cdots$ | $\cdots$ | $\cdots$ | $\cdots$ | | | $\ddots$ | | $\vdots$ |
| $x_{n+m}$ | $a_{m1}$ | $\cdots$ | $a_{ms}$ | $\cdots$ | $a_{mn}$ | | | | 1 | $b_m$ |
| $-z$ | $\bar{c}_1$ | $\cdots$ | 0 | $\cdots$ | $\bar{c}_n$ | | $\bar{c}_r$ | $\cdots$ | | $\bar{b}_0$ |
| $x_{n+1}$ | $\bar{a}_{11}$ | $\cdots$ | 0 | $\cdots$ | $\bar{a}_{1n}$ | 1 | $\bar{a}_{1,n+r}$ | | | $\bar{b}_1$ |
| $\vdots$ | $\cdots$ | $\cdots$ | | $\cdots$ | $\cdots$ | | $\cdots$ | | | $\vdots$ |
| $x_s$ | $a_{r1}$ | $\cdots$ | 1 | $\cdots$ | $\bar{a}_{rn}$ | | $\bar{a}_{r,n+r}$ | | | $\bar{b}_r$ |
| $\vdots$ | $\cdots$ | $\cdots$ | | $\cdots$ | $\cdots$ | | $\cdots$ | $\ddots$ | | $\vdots$ |
| $x_{n+m}$ | $\bar{a}_{m1}$ | $\cdots$ | 0 | $\cdots$ | $\bar{a}_{mn}$ | | $\bar{a}_{m,n+r}$ | | 1 | $\bar{b}_m$ |

## 2.5 二段階法

前節の問題がシンプレックス法で解けたのは，スラック変数が自明な bfs（$x_{n+i} = b_i, i = 1, \ldots, m$）を与えたからである．一度 bfs が見出されると，シンプレックスタブローの計算によって最適解が得られる．

ところが一般には，はじめの bfs を見出すのは自明ではない．二段階法 (two phase method) では与えられた目的関数とは別の目的関数を設定し，後の問題にシンプレックス法を適用して最適解を求める (phase one)．その最適解がはじめの問題の bfs となるので，この bfs からはじめて元の問題をシンプレックス法で解く (phase two).

いま，標準形の問題 (2.16–2.18)

$$
\begin{aligned}
&-z + \sum_{j=1}^{n} c_j x_j = 0 \\
&\sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i = 1, \ldots, m \\
&x_j \geq 0, \quad j = 1, \ldots, n.
\end{aligned}
$$

が与えられているとし，必要があれば $-1$ をかけて右辺が非負としておく．制約条件の各行について人工変数 (artificial variable) と呼ばれる変数 $x_{n+1}, \cdots, x_{n+m}$ を

加え，次の形に変形する．

$$-z + \sum_{j=1}^{n} c_j x_j = 0 \tag{2.56}$$

$$\sum_{j=1}^{n} a_{ij} x_j + x_{n+i} = b_i, \quad i = 1, \ldots, m \tag{2.57}$$

$$x_j \geq 0, \quad j = 1, \ldots, n+m. \tag{2.58}$$

人工変数を基底とすると初期の bfs として

$$x_1 = \cdots = x_n = 0, \qquad x_{n+i} = b_i, \quad i = 1, \ldots, m$$

が得られる．

ところで，もし bfs のなかで人工変数がすべてゼロになるものが求まれば，人工変数をすべて除去できるので，その bfs が元の問題の bfs になっていることがわかる．

人工変数がすべてゼロになるものを求めるために，別の目的関数として

$$w = x_{n+1} + \cdots + x_{n+m} \to \min$$

を考える．この最適解が $w = 0$ を満たせば $x_{n+1} = \cdots = x_{n+m} = 0$ であるから，最適解から人工変数をすべて除去でき，もとの解の bfs が得られる．また，逆にもとの解の bfs があれば，それは $x_{n+1} = \cdots = x_{n+m} = 0$ を意味するので，目的関数 $w$ の最適値は $w = 0$ を満たす．

制約条件から

$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{ij} x_j \quad i = 1, \ldots, m$$

である．

$$\begin{aligned} d_j &= -\sum_{i=1}^{m} a_{ij}, \quad j = 1, \ldots, n \\ w_0 &= \sum_{i=1}^{m} b_i \end{aligned}$$

とおけば,

$$w = \sum_{j=1}^{n} d_j x_j + w_0$$

と表される．

シンプレックスタブローを以下のように書く．この表では，人工変数に対応する $w, z$ の係数はゼロであり，人工変数はゼロになるか否かを判定するだけなので，単に基底の列に記述し，それ以外の列を設けないでよいことがわかる．

| 基底 | $x_1$ | $\cdots$ | $x_j$ | $\cdots$ | $x_n$ | 定数 |
|---|---|---|---|---|---|---|
| $-w$ | $d_1$ | $\cdots$ | $d_j$ | $\cdots$ | $d_n$ | $-w_0$ |
| $-z$ | $c_1$ | $\cdots$ | $c_j$ | $\cdots$ | $c_n$ | $0$ |
| $x_{n+1}$ | $a_{11}$ | $\cdots$ | $a_{1j}$ | $\cdots$ | $a_{1n}$ | $b_1$ |
| $x_{n+2}$ | $a_{21}$ | $\cdots$ | $a_{2j}$ | $\cdots$ | $a_{2n}$ | $b_2$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\cdots$ | $\cdots$ | $\vdots$ | $\vdots$ |
| $x_{n+m}$ | $a_{m1}$ | $\cdots$ | $a_{ms}$ | $\cdots$ | $a_{mn}$ | $b_m$ |

このシンプレックスタブローでは，$w$ を目的関数として計算を行うが，$z$ の行にも演算を施し，$w$ の最適値がゼロになった段階で $w$ の行を消去して通常のシンプレックスタブローの計算を続行すればよい（$w$ の最適値がゼロにならなければもとの問題は実行可能でない）.

例．栄養問題

$$
\begin{aligned}
&z = 2x_1 + 3x_2 \to \min \\
&x_1 + 2x_2 \geq 10 \\
&x_1 + 4x_2 \geq 12 \\
&3x_1 + 2x_2 \geq 15 \\
&x_j \geq 0, \quad j = 1, 2
\end{aligned}
$$

を標準形に直す.

$$
\begin{aligned}
&-z + 2x_1 + 3x_2 = 0 \\
&x_1 + 2x_2 - x_3 = 10 \\
&x_1 + 4x_2 - x_4 = 12 \\
&3x_1 + 2x_2 - x_5 = 15 \\
&x_j \geq 0, \quad j = 1, \ldots, 5
\end{aligned}
$$

これに人工変数 $x_7, x_8, x_9$ を導入して次のシンプレックスタブローを得る．このタブローはこれを $w$ について最適化する過程を示している.

| 基底 | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | 定数 |
|---|---|---|---|---|---|---|
| $-w$ | $-5$ | $\underline{-8}$ | $1$ | $1$ | $1$ | $-37$ |
| $-z$ | $2$ | $3$ | | | | $0$ |
| $x_6$ | $1$ | $2$ | $-1$ | | | $10$ |
| $x_7$ | $1$ | $\boxed{4}$ | | $-1$ | | $12$ |
| $x_8$ | $3$ | $2$ | | | $-1$ | $15$ |
| $-w$ | $\underline{-3}$ | | $1$ | $-1$ | $1$ | $-13$ |
| $-z$ | $1.25$ | | | $0.75$ | | $-9$ |
| $x_6$ | $0.5$ | | $-1$ | $0.5$ | | $4$ |
| $x_2$ | $0.25$ | $1$ | | $-0.25$ | | $3$ |
| $x_8$ | $\boxed{2.5}$ | | | $0.5$ | $-1$ | $9$ |
| $-w$ | | | $1$ | $\underline{-0.4}$ | $-0.2$ | $-2.2$ |
| $-z$ | | | | $0.5$ | $0.5$ | $-13.5$ |
| $x_6$ | | | $-1$ | $\boxed{0.4}$ | $0.2$ | $2.2$ |
| $x_2$ | | $1$ | | $-0.3$ | $0.1$ | $2.1$ |
| $x_1$ | $1$ | | | $0.2$ | $-0.4$ | $3.6$ |
| $-w$ | | | $0$ | | $0$ | $0$ |
| $-z$ | | | $1.25$ | | $0.25$ | $-16.25$ |
| $x_4$ | | | $-2.5$ | $1$ | $0.5$ | $5.5$ |
| $x_2$ | | $1$ | $-0.75$ | | $0.25$ | $3.75$ |
| $x_1$ | $1$ | | $0.5$ | | $-0.5$ | $2.5$ |

タブローの終りで，$w$ の値はゼロとなり，元の問題が実行可能であることが示されている．しかもこのとき，$z$ の係数がすべて正となるので，この実行可能解は元の問題の最適解である．すなわち，最適解 $x_1 = 2.5, x_2 = 3.75, x_4 = 5.5, z = 16.25$ が得られた．

**問．** 次の線形計画問題を 2 段階法によって解け．

$$\begin{aligned} z &= x_1 + x_2 \to \min \\ & 2x_1 + x_2 \geq 12 \\ & x_1 + 2x_2 \geq 15 \\ & x_1, x_2 \geq 0 \end{aligned}$$

## 2.6　双対性

双対性 (duality) は線形計画法の理論の中心である．アルゴリズムの面では内点法に関連し，具体的な意味付けも可能である．

標準形の問題を今，主問題 (primal problem) と呼んでおく．

**（主問題 $P$）**

$$z = \boldsymbol{c}^\top \boldsymbol{x} \to \min \tag{2.59}$$

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.60}$$

$$\boldsymbol{x} \geq \boldsymbol{0} \tag{2.61}$$

これに対する双対問題は次のように定義される．

**（双対問題 $D$）**

$$w = \boldsymbol{y}^\top \boldsymbol{b} \to \max \tag{2.62}$$

$$\boldsymbol{y}^\top A \leq \boldsymbol{c}^\top \tag{2.63}$$

このように，主問題である最小化問題が等式制約条件 (2.60) と変数についての非負条件 (2.61) をもつとき，双対問題は最大化問題であり，不等式制約 (2.62) をもち，変数に対する制約をもたない．また，$A, \boldsymbol{b}, \boldsymbol{c}$ について $A \Rightarrow A^\top$, $\boldsymbol{c} \Rightarrow \boldsymbol{b}$, $\boldsymbol{b} \Rightarrow \boldsymbol{c}$ の対応がある．

**双対問題 $D$ の双対は主問題 $P$ である．**

このことを示すために，$\boldsymbol{D}$ を主問題の形に書き直す．まず，非負の制約がない $\boldsymbol{y}$ を 2 つの非負ベクトルの差として表す．$\boldsymbol{y} = \boldsymbol{y}^+ - \boldsymbol{y}^-$, $\boldsymbol{y}^+ \geq \boldsymbol{0}$, $\boldsymbol{y}^- \geq \boldsymbol{0}$. さらにスラック変数ベクトル $\boldsymbol{z} \geq \boldsymbol{0}$ を導入し，(2.63) を $A^\top(\boldsymbol{y}^+ - \boldsymbol{y}^-) + \boldsymbol{z} = \boldsymbol{c}$ と書く．よって，双対問題は

$$\begin{aligned}
&-\boldsymbol{b}^\top \boldsymbol{y}^+ + \boldsymbol{b}^\top \boldsymbol{y}^- + \boldsymbol{0}^\top \boldsymbol{z} \to \min \\
&A^\top(\boldsymbol{y}^+ - \boldsymbol{y}^-) + \boldsymbol{z} = \boldsymbol{c} \\
&\boldsymbol{y}^+ \geq \boldsymbol{0},\ \boldsymbol{y}^- \geq \boldsymbol{0},\ \boldsymbol{z} \geq \boldsymbol{0}
\end{aligned}$$

と標準形に書ける．さらに書き直すと

$$
\begin{aligned}
&(-\boldsymbol{b}^\top, \boldsymbol{b}^\top, \boldsymbol{0}) \begin{pmatrix} \boldsymbol{y}^+ \\ \boldsymbol{y}^- \\ \boldsymbol{z} \end{pmatrix} \to \min \\
&(-A^\top, A^\top, -I) \begin{pmatrix} \boldsymbol{y}^+ \\ \boldsymbol{y}^- \\ \boldsymbol{z} \end{pmatrix} = -\boldsymbol{c} \\
&\begin{pmatrix} \boldsymbol{y}^+ \\ \boldsymbol{y}^- \\ \boldsymbol{z} \end{pmatrix} \geq \boldsymbol{0}
\end{aligned}
$$

となり，この双対問題は $\boldsymbol{\zeta}$ を変数として

$$
\begin{aligned}
&-\boldsymbol{c}^\top \boldsymbol{\zeta} \to \max \\
&\begin{pmatrix} -A \\ A \\ -I \end{pmatrix} \boldsymbol{\zeta} \leq \begin{pmatrix} -\boldsymbol{b} \\ \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix}
\end{aligned}
$$

となる．

制約条件は $-A\boldsymbol{\zeta} \leq -\boldsymbol{b}$, $A\boldsymbol{\zeta} \leq \boldsymbol{b}$, $-\boldsymbol{\zeta} \leq \boldsymbol{0}$ を意味しているので，これから，

$$
\begin{aligned}
&\boldsymbol{c}^\top \boldsymbol{\zeta} \to \min \\
&A\boldsymbol{\zeta} = \boldsymbol{b} \\
&\boldsymbol{\zeta} \geq 0
\end{aligned}
$$

を得る．すなわち主問題 $\boldsymbol{P}$ が得られた．//

生産計画問題 $\mathbf{P}$ の双対問題について考えよう．

**生産計画問題 P**

$$z = \boldsymbol{c}^\top \boldsymbol{x} \to \max \tag{2.64}$$

$$A\boldsymbol{x} \leq \boldsymbol{b} \tag{2.65}$$

$$\boldsymbol{x} \geq \boldsymbol{0} \tag{2.66}$$

**の双対問題は栄養問題 D**

$$w = \boldsymbol{y}^\top \boldsymbol{b} \to \min \tag{2.67}$$

$$\boldsymbol{y}^\top A \geq \boldsymbol{c}^\top \tag{2.68}$$

$$\boldsymbol{y} \geq \boldsymbol{0} \tag{2.69}$$

**である．**

このことを示すには，生産計画問題 $\mathrm{P}$ を標準形の主問題 $\boldsymbol{P}$ の形に書き，その双対 $\boldsymbol{D}$ が栄養問題 $\mathrm{D}$ に帰着することをいえばよい．

スラック変数 $\boldsymbol{\eta}$ を用いて問題 $\mathrm{P}$ を標準形に変えると

$$
\begin{aligned}
&(-\boldsymbol{c}^\top, 0)\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{\eta}\end{pmatrix} \to \min\\
&(-A, -I)\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{\eta}\end{pmatrix} = -\boldsymbol{b}\\
&\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{\eta}\end{pmatrix} \geq \boldsymbol{0}
\end{aligned}
$$

である．この双対 $\boldsymbol{D}$ は

$$
\begin{gathered}
\boldsymbol{y}^\top(-\boldsymbol{b}) \to \max\\
\begin{pmatrix}-A^\top\\ -I\end{pmatrix}\boldsymbol{y} \leq \begin{pmatrix}-\boldsymbol{c}\\ \boldsymbol{0}\end{pmatrix}
\end{gathered}
$$

となる．最後の条件は，$-A^\top\boldsymbol{y} \leq -\boldsymbol{c}$, $-\boldsymbol{y} \leq \boldsymbol{0}$ を意味するから

$$
\begin{aligned}
&\boldsymbol{y}^\top\boldsymbol{b} \to \min\\
&A^\top\boldsymbol{y} \geq \boldsymbol{c}\\
&\boldsymbol{y} \geq \boldsymbol{0}
\end{aligned}
$$

すなわち栄養問題 $\mathrm{D}$ が得られた．

**問．** 栄養問題 $\mathrm{D}$ を標準形に変換し，その双対をとることによって，栄養問題 $\mathrm{D}$ の双対が生産計画問題 $\mathrm{P}$ であることを証明せよ．

主問題と双対問題の間には次の定理によって述べられる関係がある．

**定理 (弱双対定理)．** 主問題 $\boldsymbol{P}$ の任意の実行可能解 (fs) $\tilde{\boldsymbol{x}}$ とそれに対応する双対問題 $\boldsymbol{D}$ の任意の実行可能解 (fs) $\tilde{\boldsymbol{y}}$ の間に

$$
\tilde{z} = \boldsymbol{c}^\top\tilde{\boldsymbol{x}} \geq \tilde{\boldsymbol{y}}^\top\boldsymbol{b} = \tilde{w} \tag{2.70}
$$

が成立する．

(証明) $\boldsymbol{P}$ における $A\tilde{\boldsymbol{x}} = \boldsymbol{b}$ の両辺に左から $\tilde{\boldsymbol{y}}^\top$ をかける（いいかえれば内積をとる）と

$$
\tilde{\boldsymbol{y}}^\top A\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{y}}^\top\tilde{\boldsymbol{b}}
$$

$\tilde{\boldsymbol{y}}^\top A \leq \boldsymbol{c}^\top$ かつ $\tilde{\boldsymbol{x}} \geq \boldsymbol{0}$ だから $\tilde{\boldsymbol{y}}^\top A\tilde{\boldsymbol{x}} \leq \boldsymbol{c}^\top\tilde{\boldsymbol{x}}$．よって，上の式とあわせると

$$\tilde{\boldsymbol{y}}^\top \boldsymbol{b} = \tilde{\boldsymbol{y}}^\top A\tilde{\boldsymbol{x}} \leq \boldsymbol{c}^\top\tilde{\boldsymbol{x}}$$

よって証明された．//

**問．** 生産計画問題 $\mathbf{P}$ の fs $\hat{\boldsymbol{x}}$ と栄養問題 $\mathbf{D}$ の fs $\hat{\boldsymbol{y}}$ の間に $\boldsymbol{c}^\top\hat{\boldsymbol{x}} \leq \hat{\boldsymbol{y}}^\top\boldsymbol{b}$ が成り立つことを上の議論に従って直接的に証明せよ．

**定理（双対定理）．** 主問題 $\boldsymbol{P}$ と双対問題 $\boldsymbol{D}$ のどちらかに最適解が存在するならば，他方にも最適解が存在し，それらの目的関数の値は等しい．すなわち，最適解をそれぞれ $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}$ とすると，

$$\bar{z} = \boldsymbol{c}^\top\bar{\boldsymbol{x}} = \bar{\boldsymbol{y}}^\top\boldsymbol{b} = \bar{w}$$

証明は省略する．

**定理（双対定理の系）．** 主問題 $\boldsymbol{P}$ と双対問題 $\boldsymbol{D}$ の双方が実行可能ならば，双方の問題に最適解が存在し，それらの目的関数の値は等しい．

### 2.6.1 双対性の解釈

**生産計画問題** $\mathbf{P}$ を思い出そう．ある企業 $P$ が 2 種類の原料 $R_1, R_2$ を使って 2 種類の製品 $P_1, P_2$ を生産する．(a) 全部で $R_1$ は 20 単位，$R_2$ は 15 単位使える．(b) $P_1$ を 1 単位生産するために，$R_1$ が 2 単位，$R_2$ が 4 単位必要である．また，$P_2$ を 1 単位生産するために，$R_1$ が 5 単位，$R_2$ が 1 単位必要である．(c) $P_1$ について，1 単位生産するごとに 3 単位の利益があり，$P_2$ について，1 単位生産するごとに 4 単位の利益がある．これらの条件のもとで総利益を最大とする $P_1, P_2$ の生産量を求める問題であった．

いま，別の企業 $D$ が $P$ から原料 $R_1, R_2$ をすべて買い取ろうとする．このとき，原料 $R_1, R_2$ にどのような価格付けを行えば $P$ は買取りに応じるだろうか．

$R_1, R_2$ に対する価格をそれぞれ $y_1, y_2$ とする．製品 $P_1$ を 1 単位生産した場合の利益 3 を上回り，かつ製品 $P_2$ を 1 単位生産した場合の利益 4 を上回る価格付けがされているとき，かつそのときに限って $P$ は買取りに応じるだろう．さきの条件は $2y_1 + 4y_2 \geq 3$ であり，あとの条件は $5y_1 + y_2 \geq 4$ と表される．すなわち，$P$ が買取りに応じるためには $y_1, y_2$ は

$$\begin{aligned} &2y_1 + 4y_2 \geq 3 \\ &5y_1 + y_2 \geq 4 \\ &y_1 \geq 0,\ y_2 \geq 0 \end{aligned}$$

の実行可能解であることが必要十分である．

$D$ は総買取りコストを最小にしたいから，目的関数

$$w = 20y_1 + 15y_2 \to \min$$

をもっている．

すなわち，主問題である

$$\begin{aligned}
&z = 3x_1 + 4x_2 \to \max \\
&2x_1 + 5x_2 \leq 20 \\
&4x_1 + x_2 \leq 15 \\
&x_1 \geq 0,\ x_2 \geq 0
\end{aligned}$$

の双対問題が得られた．

弱双対定理から，2 つの実行可能解 $\tilde{x}, \tilde{y}$ について

$$\tilde{z} = 3\tilde{x}_1 + 4\tilde{x}_2 \leq 20\tilde{y}_1 + 15\tilde{y}_2 = \tilde{w}$$

が成立している．ここで，等号が成立していなければ，$\tilde{x}, \tilde{y}$ のいずれか一方は最適ではない．つまり，$P$ は最大の利益を上げられないで損をしているか，あるいは $D$ の価格付けが高すぎることになる．両方の解が最適なとき，強双対定理から等号が成り立つことがわかる．このとき，$P$ があげうる最大利益と $D$ の適正な価格付けによるコストが一致する．

次に栄養問題 **D** について考える．ある農場 $\mathcal{D}$ の家畜には，2 種類の栄養素 $N_1, N_2$ が必要であり，飼料は $F_1, F_2$ の 2 種類である．(a') 全栄養素は $N_1$ について 16 単位，$N_2$ について 24 単位必要であり，(b') $F_1$ の 1 単位には，栄養素 $N_1$ が 4 単位，$N_2$ が 2 単位含まれている．また，$F_2$ の 1 単位には，栄養素 $N_1$ が 3 単位，$N_2$ が 7 単位含まれている．(c') $F_1$ について，1 単位の価格は 4 単位であり，$F_2$ について，1 単位の価格は 5 単位である．これらの制約条件のもとで総コストを最小化する問題が考察された．

いま，ある企業 $\mathcal{P}$ が栄養素 $N_1, N_2$ だけを含む 2 種類の栄養剤 $D_1, D_2$ を生産して，飼料に代えようとする（いささか非現実的な仮定ではあるが）．栄養剤の価格をどう設定すれば農場の要求を満足させられるだろうか．

$D_1, D_2$ の栄養素 1 単位あたりの価格をそれぞれ $x_1, x_2$ とする．$D_1, D_2$ を飼料 $F_1$ の代わりに使ったとして，コストが $F_1$ より安いという条件は

$$4x_1 + 2x_2 \leq 4$$

であり，$D_1, D_2$ を飼料 $F_2$ の代わりに使ったとして，コストが $F_2$ より安いという条件は

$$3x_1 + 7x_2 \leq 5$$

である．この条件が満たされれば $\mathcal{D}$ は栄養剤を飼料の代わりに使うだろう．

$\mathcal{P}$ は売り上げを最大にしたいので，$\mathcal{D}$ に売った場合の売り上げから得られる目的関数は

$$z = 16x_1 + 24x_2 \to \max$$

となる．すなわち，栄養問題 **D**

$$\begin{aligned}
&w = 4y_1 + 5y_2 \to \min \\
&4y_1 + 3y_2 \geq 16 \\
&2y_1 + 7y_2 \geq 24 \\
&y_1 \geq 0,\ y_2 \geq 0
\end{aligned}$$

の双対問題

$$\begin{aligned}
&z = 16x_1 + 24x_2 \to \max \\
&4x_1 + 2x_2 \leq 4 \\
&3x_1 + 7x_2 \leq 5 \\
&x_1 \geq 0,\ x_2 \geq 0
\end{aligned}$$

が得られた．さきに述べたように双対問題は生産計画型の問題である．

## 2.7 内点法

シンプレックス法は，広く使われている解法であるが，理論的には必ずしも計算効率がよい解法とはいえない．そこで，線形計画問題を解く効率のよい解法が多くの研究者によって考察された．その結果，生み出されたのが内点法である．

シンプレックス法では，実行可能領域 $X$ の頂点を辿るが，内点法では $X$ の内部を通って最適解に収束する解の列が生成される．理論的には，内点法のほうがシンプレックス法よりも効率がよい（計算複雑さが小さい）ことが証明されている．

内点法には様々なアルゴリズムがあるが，主問題と双対問題を同時に解きながら解を見いだす主双対法について簡単に述べておこう [2]．まず，標準形の主問題

$$\begin{aligned}
&\boldsymbol{c}^\top \boldsymbol{x} \to \min \\
&A\boldsymbol{x} = \boldsymbol{b} \\
&\boldsymbol{x} \geq \boldsymbol{0}
\end{aligned}$$

および双対問題

$$\begin{aligned}
&\boldsymbol{b}^\top \boldsymbol{y} \to \max \\
&A^\top \boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c} \\
&\boldsymbol{s} \geq \boldsymbol{0}
\end{aligned}$$

を思い出そう．双対問題には補助変数 $\boldsymbol{s}$ が使われている．

ここで，双対定理から，$\boldsymbol{x}$ と $\boldsymbol{y}, \boldsymbol{s}$ が最適解であることと 2 つの目的関数の値は一致すること，すなわち，

$$\boldsymbol{c}^\top \boldsymbol{x} = \boldsymbol{b}^\top \boldsymbol{y} \tag{2.71}$$

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.72}$$

$$A^\top \boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c} \tag{2.73}$$

$$\boldsymbol{x} \geq \boldsymbol{0},\ \boldsymbol{s} \geq \boldsymbol{0} \tag{2.74}$$

は等価である．

$$\boldsymbol{c}^\top \boldsymbol{x} - \boldsymbol{b}^\top \boldsymbol{y} = (A^\top \boldsymbol{y} + \boldsymbol{s})^\top \boldsymbol{x} - \boldsymbol{b}^\top \boldsymbol{y} = \boldsymbol{s}^\top \boldsymbol{x} - (A\boldsymbol{x} - \boldsymbol{b})^\top \boldsymbol{y} = \boldsymbol{s}^\top \boldsymbol{x} = 0$$

と，$\boldsymbol{s} \geq \boldsymbol{0}, \boldsymbol{x} \geq \boldsymbol{0}$ より

$$\boldsymbol{s}^\top \boldsymbol{x} = 0 \iff s_j x_j = 0, \quad j = 1, \ldots, n$$

に注意すれば，(2.71) は

$$XS\boldsymbol{e} = \boldsymbol{0} \tag{2.75}$$

と書きかえることができる．ここで，$X = diag(x_1, \cdots, x_n)$ ($(x_1, \cdots, x_n)$ を対角要素とする対角行列), $S = diag(s_1, \cdots, s_n)$, $\boldsymbol{e} = (1, \ldots, 1)^\top$ である．

主双対法では，非線形方程式の組 (2.72–2.75) を解くことを目的とする．このために，パラメータ $\mu > 0$ を導入し，$\boldsymbol{x} \geq \boldsymbol{0}$, $\boldsymbol{s} \geq \boldsymbol{0}$ を $\boldsymbol{x} > \boldsymbol{0}$, $\boldsymbol{s} > \boldsymbol{0}$ で置き換えて $x_j s_j = 0$ を $x_j s_j = \mu$ で置き換える．よって，(2.72–2.75) の代わりに，次の式を考察する．

$$XS\boldsymbol{e} = \mu \boldsymbol{e} \tag{2.76}$$

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.77}$$

$$A^\top \boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c} \tag{2.78}$$

$$\boldsymbol{x} > \boldsymbol{0},\ \boldsymbol{s} > \boldsymbol{0} \tag{2.79}$$

この式の解は，(2.72–2.75) の解と一致しないが，パラメータ $\mu$ をゼロに近づけるともとの式を表すので，解ももとの式の解に収束することが期待される．実際，(2.76–2.79) の解を $(\boldsymbol{x}(\mu), \boldsymbol{y}(\mu), \boldsymbol{s}(\mu))$ とし，もとの方程式の解を $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{\boldsymbol{s}})$ とするとき，

$$\lim_{\mu \to 0} (\boldsymbol{x}(\mu), \boldsymbol{y}(\mu), \boldsymbol{s}(\mu)) = (\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{\boldsymbol{s}})$$

であることが証明されている（ここでは省略する）．

内点法の計算では，$\mu_k$，$(\boldsymbol{x}(k), \boldsymbol{y}(k), \boldsymbol{s}(k))$ （簡単のため，$(\boldsymbol{x}(\mu_k), \boldsymbol{y}(\mu_k), \boldsymbol{s}(\mu_k))$ の代わりに $(\boldsymbol{x}(k), \boldsymbol{y}(k), \boldsymbol{s}(k))$ と書く）を与えたときの (2.76–2.79) の近似解を求めて $(\boldsymbol{x}(k+1), \boldsymbol{y}(k+1), \boldsymbol{s}(k+1))$ とし，さらに次の $\mu_{k+1}$ を作る．このとき，

$$\mu_{k+1} < \mu_k, \quad \mu_k \to 0\ (k \to \infty)$$

を満たし，

$$\boldsymbol{x}(k+1) > 0, \quad \boldsymbol{s}(k+1) > 0, \quad \lim_{\mu_k \to 0}(\boldsymbol{x}(k), \boldsymbol{y}(k), \boldsymbol{s}(k)) = (\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \bar{\boldsymbol{s}})$$

であるようにする．$\boldsymbol{x}(k) > 0, \boldsymbol{s}(k) > 0$ すなわち実行可能領域の内部に点があるため，内点法という．

$\mu_k$ を与えたときの近似解を求めるために，(2.76–2.79) を線形近似する．まず，次の式から $\Delta\boldsymbol{x}, \Delta\boldsymbol{y}, \Delta\boldsymbol{s}$ を求める．$(\boldsymbol{x}(k), \boldsymbol{y}(k), \boldsymbol{s}(k))$ とそれから定まる $\mu_k$（後の (2.84), (2.85) 参照）が近似解として既に得られているとし，$X(k) = diag(x_1(k), \cdots, x_n(k))$, $S(k) = diag(s_1(k), \cdots, s_n(k))$ とおいている．

$$\begin{aligned} S(k)\Delta\boldsymbol{x} + X(k)\Delta\boldsymbol{s} &= \mu_k\boldsymbol{e} - X(k)S(k)\boldsymbol{e} && (2.80) \\ A\Delta\boldsymbol{x} &= b - A\boldsymbol{x}(k) && (2.81) \\ A^\top\Delta\boldsymbol{y} + \Delta\boldsymbol{s} &= c - A^\top\boldsymbol{y}(k) - \boldsymbol{s}(k) && (2.82) \end{aligned}$$

この方程式は $(\Delta\boldsymbol{x}, \Delta\boldsymbol{y}, \Delta\boldsymbol{s})$ について線形であり，係数行列は正則であることが示されるので，一意な解をもつ．

次に適当なステップ幅 $\lambda > 0$ を決めて

$$\begin{pmatrix} \boldsymbol{x}(k+1) \\ \boldsymbol{y}(k+1) \\ \boldsymbol{s}(k+1) \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}(k) + \lambda\Delta\boldsymbol{x} \\ \boldsymbol{y}(k) + \lambda\Delta\boldsymbol{y} \\ \boldsymbol{s}(k) + \lambda\Delta\boldsymbol{s} \end{pmatrix} \qquad (2.83)$$

とする．ステップ幅 $\lambda$ は $\boldsymbol{x}(k+1) > 0$, $\boldsymbol{s}(k+1) > 0$ が満たされるように決める必要がある．たとえば，可能な最大ステップ幅を

$$\lambda_{\max} = \max\{\lambda : \boldsymbol{x}(k) + \lambda\Delta\boldsymbol{x} \geq 0,\ \boldsymbol{s}(k) + \lambda\Delta\boldsymbol{s} \geq 0\}$$

とした上で，ある定数 $0 < \beta < 1$ について

$$\lambda = \beta\lambda_{\max}$$

とするなどの方法が用いられている．

さらに，$\mu_k$ とそれから計算された $(\boldsymbol{x}(k+1), \boldsymbol{y}(k+1), \boldsymbol{s}(k+1))$ から $\mu_{k+1}$ を定める方法として

$$\mu_{k+1} = \frac{\boldsymbol{x}(k+1)^\top\boldsymbol{s}(k+1)}{\mathrm{C}n} \qquad (\mathrm{C}\text{ は }\mathrm{C} > 1\text{ をみたす定数}) \qquad (2.84)$$

あるいは

$$\mu_{k+1} = \frac{\boldsymbol{x}(k+1)^\top\boldsymbol{s}(k+1)}{n^2} \qquad (2.85)$$

などが用いられている．

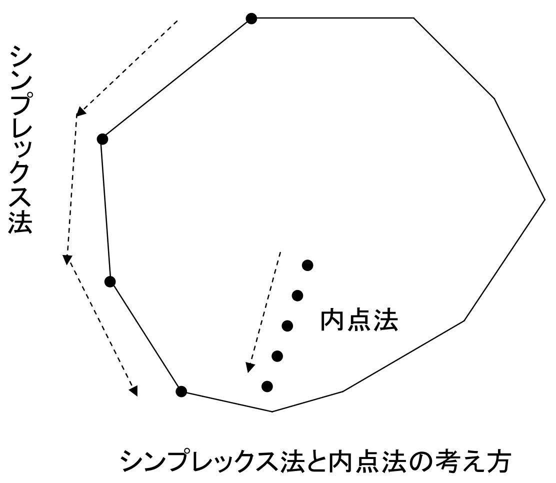



図 2.2: 内点法とシンプレックス法の考え方の比較

# 第3章　非線形最適化問題

一般には最適化問題

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & (\text{または maximize} \quad f(\boldsymbol{x})) \\ \text{subject to} & g_i(\boldsymbol{x}) \le 0, \quad i = 1, \ldots, m & \\ & h_j(\boldsymbol{x}) = 0, \quad j = 1, \ldots, \ell. & \end{array}$$

において目的関数や制約条件は線形ではない．非線形問題を扱う手法はかなり限られたものとなるが，それらのなかで深く考察されているのは，凸関数である．ここでは，まず，凸集合と凸関数について述べよう．

## 3.1　凸集合と凸関数

**定義**（凸集合; convex set）．
$\boldsymbol{R}^n$ の部分集合 $S$ ($S \subseteq \boldsymbol{R}^n$) は，任意の $\boldsymbol{x}, \boldsymbol{y} \in S$ および任意の $0 \le \lambda \le 1$ に対して

$$\lambda \boldsymbol{x} + (1 - \lambda) \boldsymbol{y} \in S \tag{3.1}$$

が成り立つとき，凸集合であると呼ばれる．

記号で書くと，

$$S : \text{凸集合} \overset{def}{\Longleftrightarrow} \forall \boldsymbol{x}, \boldsymbol{y} \in S, \quad \forall \lambda \in [0, 1], \quad \lambda \boldsymbol{x} + (1 - \lambda) \boldsymbol{y} \in S$$

幾何学的には，$S$ に含まれる任意の 2 点に対して，それらをむすぶ線分が $S$ に含まれることをいう．

**例 3.1．**　任意の $\boldsymbol{a} \in \boldsymbol{R}^n$ について $\{\boldsymbol{a}\}$ は凸集合.

**例 3.2．**　$\boldsymbol{R}^n$ の超平面

$$H(\boldsymbol{a}, \alpha) = \{\boldsymbol{x} \in \boldsymbol{R}^n : \boldsymbol{a}^\top \boldsymbol{x} = \alpha\}$$

は凸集合（ここで，$\boldsymbol{a} \in \boldsymbol{R}^n, \alpha \in \boldsymbol{R}$ は与えられたベクトルとスカラー）．

**例 3.3．**　開球 $B(\boldsymbol{z}; r) = \{\boldsymbol{x} \in \boldsymbol{R}^n : \|\boldsymbol{x} - \boldsymbol{z}\| < r\}$ および閉球 $\bar{B}(\boldsymbol{z}; r) = \{\boldsymbol{x} \in \boldsymbol{R}^n : \|\boldsymbol{x} - \boldsymbol{z}\| \le r\}$ は凸集合.

**例 3.4．** 開区間と閉区間は凸集合.

**例 3.5．** $S_1$, $S_2$ が凸集合ならば，$S_1 \cap S_2$ も凸集合．これに対して，$S_1 \cup S_2$ は凸集合とは限らない.

**例 3.6．** $S_1$, $S_2$ が凸集合ならば,

$$S_1 + S_2 = \{\, \boldsymbol{x} + \boldsymbol{y} \,:\, \boldsymbol{x} \in S_1,\, \boldsymbol{y} \in S_2 \,\}$$

も凸集合.

**問．**

(i) 例 1〜4 が成り立つことを証明せよ.

(ii) $\boldsymbol{a}_1, \boldsymbol{a}_2 \in \boldsymbol{R}^n$, $\boldsymbol{a}_1 \neq \boldsymbol{a}_2$ とする．$\{\boldsymbol{a}_1, \boldsymbol{a}_2\}$ は凸集合でないことを示せ.

(iii) $S$ を凸集合，$\alpha$ を実数とする．このとき,

$$\alpha S = \{\, \alpha \boldsymbol{x} \,:\, \boldsymbol{x} \in S\}$$

は凸集合であることを示せ.

**定義**（凸包; convex hull）.
$\boldsymbol{R}^n$ の部分集合 $S$ $(S \subseteq \boldsymbol{R}^n)$ に対して $S$ を含むすべての凸集合の共通部分を $Co(S)$ と書き，$S$ の凸包と呼ぶ.

よって，$S$ が凸集合 $\iff$ $Co(S) = S$.

**定義**（凸関数; convex function）.
$S \subseteq \boldsymbol{R}^n$ を凸集合とする．$S$ で定義された実数値関数 $f(\boldsymbol{x})$ は任意の $\boldsymbol{x}, \boldsymbol{y} \in S$ および任意の $0 \leq \lambda \leq 1$ に対して

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$$

が成り立つならば，凸関数（下に凸）であると呼ばれる.
また，任意の $\boldsymbol{x}, \boldsymbol{y} \in S$, $\boldsymbol{x} \neq \boldsymbol{y}$, および任意の $0 < \lambda < 1$ に対して

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) < \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$$

が成り立つならば，狭義凸関数 (strictly convex function) であると呼ばれる.

**問．** $f(x) = x^2$, $x \in \boldsymbol{R}$ は凸関数であることを示せ.

**問．** $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^2$, $\boldsymbol{x} \in \boldsymbol{R}^n$ は凸関数であることを示せ.

**定義**（凹関数; concave function）.
$S \subseteq \boldsymbol{R}^n$ を凸集合とする．$S$ で定義された実数値関数 $f(\boldsymbol{x})$ は任意の $\boldsymbol{x}, \boldsymbol{y} \in S$ およ

び任意の $0 \leq \lambda \leq 1$ に対して

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \geq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$$

が成り立つならば，凹関数（上に凸）であると呼ばれる．
　また，任意の $\boldsymbol{x}, \boldsymbol{y} \in S$, $\boldsymbol{x} \neq \boldsymbol{y}$, および任意の $0 < \lambda < 1$ に対して

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) > \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$$

が成り立つならば，狭義凹関数であると呼ばれる．

　明らかに $f(\boldsymbol{x})$ : 凸関数 $\iff$ $-f(\boldsymbol{x})$ : 凹関数．よって，凸関数だけを論じることにしてよい．

**問．**

(1) $f(x, y) = x^2 + y^2$ は凸関数であることを証明せよ．

(2) $f(x, y) = x^2 - y^2$ は凸関数ではないことを証明せよ．

**問．** $f(\boldsymbol{x}) = \displaystyle\sum_{i=1}^{n} a_i x_i + b = a_1 x_1 + \cdots + a_n x_n + b$ は凸関数かつ凹関数であることを示せ．

**定義**（エピグラフ; epigraph）．
$f\colon S \to \boldsymbol{R}$ に対して，$f$ のエピグラフ $epi(f)$ は次式で表される $\boldsymbol{R}^{n+1}$ の部分集合である．

$$epi(f) = \{\, (\boldsymbol{x}, \alpha) \,:\, \alpha \geq f(\boldsymbol{x}),\, \alpha \in \boldsymbol{R},\, \boldsymbol{x} \in S \,\}$$

**定理．**
$f(\boldsymbol{x})$ が凸関数であることの必要十分条件は $epi(f)$ が凸集合であることである．

　(証明)
(必要性：) $(\boldsymbol{x}, \alpha), (\boldsymbol{y}, \beta) \in epi(f)$ と仮定すると，$\alpha \geq f(\boldsymbol{x})$, $\beta \geq f(\boldsymbol{y})$. $f$: 凸の仮定から

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y}) \leq \lambda\alpha + (1-\lambda)\beta \tag{3.2}$$

よって，

$$\lambda(\boldsymbol{x}, \alpha) + (1-\lambda)(\boldsymbol{y}, \beta) \in epi(f) \tag{3.3}$$

だから，$epi(f)$ は凸である．
　(十分性：) (3.3) を仮定してよい．(3.3) は $f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda\alpha + (1-\lambda)\beta$ と等価であるから，(3.2) において $\alpha = f(\boldsymbol{x})$, $\beta = f(\boldsymbol{y})$ と選べば，$f(\boldsymbol{x})$ が凸関数であることがわかる．

**問．** 凸集合 $S$ で定義された関数 $f_1(\boldsymbol{x}), f_2(\boldsymbol{x})$ がともに凸のとき，(i) $g(\boldsymbol{x}) = f_1(\boldsymbol{x}) + f_2(\boldsymbol{x})$ は凸関数，(ii) $\alpha$ を $\alpha > 0$ である実数とするとき $h(\boldsymbol{x}) = \alpha f_i(\boldsymbol{x})$ $(i = 1, 2)$ は凸関数, を示せ.

## 3.2 最適化における凸関数の意義

制約のない問題

$$\min_{\boldsymbol{x} \in S} f(\boldsymbol{x})$$

を考える ($S$ は $\boldsymbol{R}^n$ の凸集合).

**定理．** $f(\boldsymbol{x})$ は凸関数であると仮定する．このとき，

**(i)** $f(\boldsymbol{x})$ の局所最適解はすべて大域的最適解となる．

**(ii)** 大域的最適解を与える点の集合

$$M = \{\, \bar{\boldsymbol{x}} \in \boldsymbol{R}^n \,:\, \bar{\boldsymbol{x}} = \arg\min_{\boldsymbol{x} \in S} f(\boldsymbol{x}) \,\}$$

は凸集合.

**(iii)** $f(\boldsymbol{x})$ が狭義凸ならば $M$ は一点からなる.

**(iv)** 実数 $\alpha$ を任意にとるとき，$L_\alpha = \{\, \boldsymbol{x} \in \boldsymbol{R}^n \,:\, f(\boldsymbol{x}) \le \alpha \,\}$ は凸集合.

(証明) $\bar{\boldsymbol{x}}$：大域的最適解，$\hat{\boldsymbol{x}}$：局所的最適解とし，$f(\bar{\boldsymbol{x}}) < f(\hat{\boldsymbol{x}})$ と仮定して矛盾を導く.

局所的最適解の仮定より，

$$\exists \delta > 0, \quad \forall \boldsymbol{x} \in B(\hat{\boldsymbol{x}}; \delta) \cap S, \quad f(\boldsymbol{x}) \ge f(\hat{\boldsymbol{x}}) \tag{3.4}$$

ある $\lambda \in (0, 1)$ について

$$\boldsymbol{x}_\lambda = \lambda \hat{\boldsymbol{x}} + (1 - \lambda)\bar{\boldsymbol{x}} \in B(\hat{\boldsymbol{x}}; \delta/2) \cap S$$

とすることができる．このとき，

$$f(\boldsymbol{x}_\lambda) = f(\lambda \hat{\boldsymbol{x}} + (1 - \lambda)\bar{\boldsymbol{x}}) \le \lambda f(\hat{\boldsymbol{x}}) + (1 - \lambda) f(\bar{\boldsymbol{x}}) < f(\hat{\boldsymbol{x}})$$

この式は (3.4) に矛盾する．よって，$f(\hat{\boldsymbol{x}}) = f(\bar{\boldsymbol{x}})$.

また，$\hat{\boldsymbol{x}} \ne \bar{\boldsymbol{x}}$ ならば, $\hat{\boldsymbol{x}}$ と $\bar{\boldsymbol{x}}$ を結ぶ線分上の点はすべて $f(\lambda\hat{\boldsymbol{x}} + (1-\lambda)\bar{\boldsymbol{x}}) = f(\bar{\boldsymbol{x}})$ をみたすので，$M$ は凸である.

次に，$f(\boldsymbol{x})$ が狭義凸であると仮定する．$\bar{\boldsymbol{x}} \neq \hat{\boldsymbol{x}}$ なら，$\boldsymbol{x}' = \frac{\bar{\boldsymbol{x}}+\hat{\boldsymbol{x}}}{2}$ とおくと $f(\boldsymbol{x}') < f(\bar{\boldsymbol{x}})$ となり，$\bar{\boldsymbol{x}}$ が大域的最適解であるという仮定に矛盾する．よって，$M$ は一点からなる．

**(iv)** は,

$$f(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda)f(\boldsymbol{y}) \leq \lambda\alpha + (1-\lambda)\alpha = \alpha$$

より明らか.

**定理．** 最適化問題

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \qquad (\text{または maximize} \quad f(\boldsymbol{x})) \\ \text{subject to} & g_i(\boldsymbol{x}) \leq 0, \quad i = 1, \ldots, m \end{array}$$

において $f(\boldsymbol{x})$ および $g_i(\boldsymbol{x}), i = 1, \ldots, m$ が $\boldsymbol{R}$ で定義された凸関数であると仮定する．このとき，すぐ前の定理と同じ結論 **(i)**, **(ii)**, **(iii)**, **(iv)** が成立する．

証明は省略する．簡単であるので，読者自ら試みられたい．

## 3.3 2次形式

$Q = (q_{ij})$ を対称な $n$ 次正方行列とする．すなわち，$q_{ij} = q_{ji}$ $(1 \leq i, j \leq n)$ である．

この節では

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^\top Q\boldsymbol{x} + \boldsymbol{r}^\top\boldsymbol{x} + d = \frac{1}{2}\sum_{i,j=1}^{n} q_{ij}x_ix_j + \sum_{i=1}^{n} r_ix_i + d \tag{3.5}$$

と仮定する．すなわち，$f(\boldsymbol{x})$ は 2 次関数である．これを 2 次形式ともいう．

**定理．** 関数 $f(\boldsymbol{x})$ は (3.5) で与えられるものとする．このとき，

**(a)** $f(\boldsymbol{x})$ が凸関数であるための必要十分条件は $Q$ が非負定値であることであり，

**(b)** $f(\boldsymbol{x})$ が狭義凸関数であるための必要十分条件は $Q$ が正定値であることである．

ここで，$Q$ が非負定値であるとは，$Q$ の固有値がすべて非負であることであり，$Q$ が正定値であるとは，$Q$ の固有値がすべて正であることである．

(証明) 線形代数で学んだ次の事実を思い出す.「対称行列 $Q$ は直交行列 $T$ $(T^\top = T^{-1})$ によって対角化可能である.」式で書くと,

$$TQT^\top = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$$

ここで，$\lambda_1 \geq 0 \cdots \lambda_n \geq 0$ である．

簡単のため $f(\boldsymbol{x}) = \boldsymbol{x}^\top Q\boldsymbol{x}$ と仮定する．($f(\boldsymbol{x})$ が (3.5) で与えられる場合も全く同様に示すことができる．）任意の $\boldsymbol{x}$ に対して $\boldsymbol{y} = T\boldsymbol{x}$ とおくと

$$\boldsymbol{x}^\top Q\boldsymbol{x} = \boldsymbol{x}^\top T^\top TQT^\top T\boldsymbol{x} = \boldsymbol{y}^\top TQT^\top \boldsymbol{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \cdots + \lambda_n y_n^2$$

よって，$0 \leq \mu \leq 1$ について

$$\begin{aligned} f(\mu\boldsymbol{x} + (1-\mu)\tilde{\boldsymbol{x}}) &= \lambda_1(\mu y_1 + (1-\mu)\tilde{y}_1)^2 + \cdots + \lambda_n(\mu y_n + (1-\mu)\tilde{y}_n)^2 \\ &\leq \lambda_1(\mu y_1^2 + (1-\mu)\tilde{y}_1^2) + \cdots + \lambda_n(\mu y_n^2 + (1-\mu)\tilde{y}_n^2) \\ &= \mu(\lambda_1 y_1^2 + \cdots + \lambda_n y_n^2) + (1-\mu)(\lambda_1\tilde{y}_1^2 + \cdots + \lambda_n\tilde{y}_n^2) \\ &= \mu f(\boldsymbol{x}) + (1-\mu)f(\tilde{\boldsymbol{x}}) \end{aligned}$$

すなわち $f(\boldsymbol{x})$ は凸である．狭義凸の場合も同様に示される．

**問．** 次の式を行列形式で表せ．**(i)** は非負定値か？また **(ii)** が正定値であるための条件を求めよ．

**(i)** $f(\boldsymbol{x}) = x_1^2 + 3x_1x_2 + 4x_2^2 - 5x_1 + x_2 + 10$

**(ii)** $f(\boldsymbol{x}) = ax_1^2 + 2bx_1x_2 + cx_2^2 + dx_1 + ex_2 + f$

## 3.4 微分可能な関数の最適性と凸性の判定条件

**定義．** $f(\boldsymbol{x})\colon \boldsymbol{R}^n \to \boldsymbol{R}$ が $p$ 回偏微分可能で偏導関数がすべて連続であるとき，$f(\boldsymbol{x})$ は $C^p$ クラスであるという．以下，最適性や凸性に関するいくつかの結果を述べる．

**定理．** $f(\boldsymbol{x})$ は $C^1$ クラスと仮定する．このとき，$\hat{\boldsymbol{x}}$ が $f(\boldsymbol{x})$ の局所最適解ならば $\nabla f(\hat{\boldsymbol{x}}) = \boldsymbol{0}$ である．

(証明) $\nabla f(\hat{\boldsymbol{x}}) \neq \boldsymbol{0}$ と仮定し，$\boldsymbol{a}^\top = -\nabla f(\hat{\boldsymbol{x}})/\|\nabla f(\hat{\boldsymbol{x}})\|$ とおく．$\boldsymbol{x} = \hat{\boldsymbol{x}} + \varepsilon\boldsymbol{a}$（$\varepsilon$：十分小）とすると，

$$f(\boldsymbol{x}) - f(\hat{\boldsymbol{x}}) = \nabla f(\hat{\boldsymbol{x}} + \theta\varepsilon\boldsymbol{a})\varepsilon\boldsymbol{a} < 0$$

$(0 < \theta < 1)$ ゆえ，局所最適性に反する．//

(注意：記法を簡単化するため，$\nabla f(\boldsymbol{x}) = (\partial f/\partial x_1, \cdots, \partial f/\partial x_n)$ は行ベクトルと仮定する．)

**定理．** $f(\boldsymbol{x})$ は $C^2$ クラスで，$\nabla f(\hat{\boldsymbol{x}}) = \boldsymbol{0}$ かつ $\nabla^2 f(\hat{\boldsymbol{x}})$ は正定値行列であると仮定する．このとき，$\hat{\boldsymbol{x}}$ は孤立局所最適解である．すなわち，ある $\delta > 0$ があって，任意の $\boldsymbol{x} \in B(\hat{\boldsymbol{x}};\delta)$, $\boldsymbol{x} \neq \hat{\boldsymbol{x}}$ に対して $f(\boldsymbol{x}) > f(\hat{\boldsymbol{x}})$ が成り立つ．

(証明)　$\boldsymbol{x} = \hat{\boldsymbol{x}} + \varepsilon \boldsymbol{h}$ とおくと

$$f(\boldsymbol{x}) = f(\hat{\boldsymbol{x}}) + \frac{1}{2}\varepsilon^2 \boldsymbol{h}^\top \nabla^2 f(\hat{\boldsymbol{x}} + \theta\varepsilon\boldsymbol{h})\boldsymbol{h} > f(\hat{\boldsymbol{x}}) \qquad (0 < \theta < 1)$$

**定理．**　$f(\boldsymbol{x})$ は $C^1$ クラスと仮定する．このとき，$f(\boldsymbol{x})$ が凸であることの必要十分条件は任意の $\boldsymbol{x}, \boldsymbol{y} \in \boldsymbol{R}^n$ について

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x}) \tag{3.6}$$

なることである．

(証明)　[必要性：]　$0 < \lambda < 1$ について

$$(1-\lambda)f(\boldsymbol{x}) + \lambda f(\boldsymbol{y}) \geq f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x})) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x} + \theta\lambda(\boldsymbol{y} - \boldsymbol{x}))\lambda(\boldsymbol{y} - \boldsymbol{x})$$

これから

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x} + \theta\lambda(\boldsymbol{y} - \boldsymbol{x}))(\boldsymbol{y} - \boldsymbol{x})$$

この式で $\lambda \to 0$ とすると (3.6) を得る．

[十分性：]　$\boldsymbol{z} = \lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}$ とすると $\boldsymbol{z} - \boldsymbol{x} = (1-\lambda)(\boldsymbol{y} - \boldsymbol{x})$, $\boldsymbol{y} - \boldsymbol{z} = \lambda(\boldsymbol{y} - \boldsymbol{x})$.

$$\begin{aligned} f(\boldsymbol{x}) &\geq f(\boldsymbol{z}) + \nabla f(\boldsymbol{z})(\boldsymbol{x} - \boldsymbol{z}) = f(\boldsymbol{z}) + \nabla f(\boldsymbol{z})(1-\lambda)(\boldsymbol{x} - \boldsymbol{y}) && (3.7) \\ f(\boldsymbol{y}) &\geq f(\boldsymbol{z}) + \nabla f(\boldsymbol{z})(\boldsymbol{y} - \boldsymbol{z}) = f(\boldsymbol{z}) + \nabla f(\boldsymbol{z})\lambda(\boldsymbol{y} - \boldsymbol{x}) && (3.8) \end{aligned}$$

(3.7)$\times\lambda +$ (3.8)$\times(1-\lambda)$ より

$$\lambda f(\boldsymbol{x}) + (1-\lambda)f(\boldsymbol{y}) \geq f(\boldsymbol{z}) = f(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y})$$

を得る．//

**定理．**　$f(\boldsymbol{x})$ は $C^2$ クラスと仮定する．このとき，$f(\boldsymbol{x})$ が凸であることの必要十分条件は任意の $\boldsymbol{x} \in \boldsymbol{R}^n$ について $\nabla^2 f(\boldsymbol{x})$ が非負定値であることである．

(証明)

$$f(\boldsymbol{y}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x}) + \frac{1}{2}(\boldsymbol{y} - \boldsymbol{x})^\top \nabla^2 f(\boldsymbol{x} + \theta(\boldsymbol{y} - \boldsymbol{x}))(\boldsymbol{y} - \boldsymbol{x})$$

に注意すれば十分性は一つ前の定理より明らかである．

必要性を証明するため，$\nabla^2 f(\boldsymbol{x})$ が非負定値でないと仮定する．このとき，ある $\boldsymbol{x}, \boldsymbol{z} \in \boldsymbol{R}^n$, $\|\boldsymbol{z}\| = 1$ について $\boldsymbol{z}^\top \nabla^2 f(\boldsymbol{x})\boldsymbol{z} < 0$ がみたされる．$\boldsymbol{y} = \boldsymbol{x} + \mu\boldsymbol{z}$ とおくと

$$f(\boldsymbol{y}) - \{f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})\} = \frac{1}{2}\mu^2 \boldsymbol{z}^\top \nabla^2 f(\boldsymbol{x} + \theta(\boldsymbol{y} - \boldsymbol{x}))\boldsymbol{z} < 0$$

となるので，前の定理より $f(\boldsymbol{x})$ は凸でない．

## 制約のある問題における最適性の必要条件

この節では，$f\colon \boldsymbol{R}^n \to \boldsymbol{R}$ は $C^1$ クラスと仮定する．制約のない問題の場合，$\nabla f(\hat{\boldsymbol{x}}) = 0$ が $\hat{\boldsymbol{x}}$ が $f(\boldsymbol{x})$ の局所最適解であるための必要条件であった（前節の定理参照）．

制約がある場合は，ラグランジュの必要条件とクーン・タッカーの条件が最適性の必要条件としてよく知られている．これらについて簡単に述べよう．

まず，等式制約のみをもつ最適化問題

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & h_j(\boldsymbol{x}) = 0, \quad j = 1, \ldots, \ell. \end{array}$$

を考察する．これに対してラグランジュ関数 $L(\boldsymbol{x}, \boldsymbol{\lambda})$ を導入する．$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_\ell)$ はラグランジュ乗数と呼ばれる．

$$L(x, \lambda) = f(\boldsymbol{x}) + \sum_{j=1}^{\ell} \lambda_j h_j(\boldsymbol{x}) \tag{3.9}$$

**定理．**（ラグランジュの必要条件） $\bar{\boldsymbol{x}}$ が上の最適化問題の最適解ならば，ある $\boldsymbol{\lambda} = \bar{\boldsymbol{\lambda}}$ が存在して，

$$\left.\frac{\partial L}{\partial x_k}\right|_{\substack{\boldsymbol{x}=\bar{\boldsymbol{x}} \\ \boldsymbol{\lambda}=\bar{\boldsymbol{\lambda}}}} = 0, \quad k = 1, \ldots, n \tag{3.10}$$

$$h_j(\bar{\boldsymbol{x}}) = 0, \quad j = 1, \ldots, \ell \tag{3.11}$$

が成立する．

次に，不等式制約をもつ最適化問題

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & g_i(\boldsymbol{x}) \le 0, \quad i = 1, \ldots, m. \end{array}$$

を考える．これに対して先に定義した (3.9) と同じ形の関数を用いる．

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i g_i(\boldsymbol{x}) \tag{3.12}$$

**定理．**（クーン・タッカーの必要条件） $\bar{\boldsymbol{x}}$ が上の最適化問題の最適解ならば，ある $\boldsymbol{\lambda} = \bar{\boldsymbol{\lambda}}$ が存在して，

$$\left.\frac{\partial \mathcal{L}}{\partial x_k}\right|_{\substack{\boldsymbol{x}=\bar{\boldsymbol{x}} \\ \boldsymbol{\lambda}=\bar{\boldsymbol{\lambda}}}} = 0, \quad k = 1, \ldots, n \tag{3.13}$$

$$g_i(\bar{\boldsymbol{x}}) \le 0, \qquad \bar{\lambda}_i \ge 0, \qquad \bar{\lambda}_i g_i(\bar{\boldsymbol{x}}) = 0, \qquad i = 1, \ldots, m \tag{3.14}$$

が成立する．

ラグランジュ乗数定理とクーン・タッカーの定理の証明は省略する．講義時に幾何学的解釈を述べる予定である．

**例．** 次の最適化問題を考える．

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) = \dfrac{x_1^2}{4} + x_2^2 - \dfrac{x_1}{2} - 2x_2 \\ \text{subject to} & h(\boldsymbol{x}) = x_1 + x_2 = 0 \end{array}$$

ラグランジュ関数を

$$L(\boldsymbol{x}, \lambda) = f(\boldsymbol{x}) + \lambda h(\boldsymbol{x}) = \frac{x_1^2}{4} + x_2^2 - \frac{x_1}{2} - 2x_2 + \lambda(x_1 + x_2)$$

とする．

$$\begin{aligned} \frac{\partial L}{\partial x_1} = 0 \quad &\Rightarrow \quad x_1 = 1 - 2\lambda \\ \frac{\partial L}{\partial x_2} = 0 \quad &\Rightarrow \quad x_2 = 1 - \lambda/2 \end{aligned}$$

$x_1 + x_2 = 2 - 5/2\lambda = 0$ より $\lambda = 4/5$．よって，$x_1 = -3/5, x_2 = 3/5$．もとの目的関数は凸であり，この解は実際に最適解になっている．

## 3.5 非線形最適化のための計算法

この節では，非線形問題に対するいくつかの計算法を紹介する．

### 3.5.1 非線形方程式の近似解法

最適化問題そのものではないが，非線形方程式の近似解を求める方法は様々な面で非線形最適化に関連している．ここではニュートン法を紹介する．

#### ニュートン法

1 変数 $x \in \boldsymbol{R}$ の関数 $f(x)$ が連続微分可能であると仮定する．

$$f(x) = 0 \tag{3.15}$$

の解を求める数値計算法として最も良く知られているのがニュートン法である．

ニュートン法の原理は線形近似にもとづいている．すなわち，ある近似解 $x_0$ が与えられたとき，$x_0$ において (3.15) を線形近似し，近似による線形方程式の解を次の解 $x_1$ とする．これを繰り返すことによって，近似解を真の解に近づけようとする．

$y = f(x)$ の $x = x_0$ における線形近似は，

$$y = f'(x_0)(x - x_0) + f(x_0)$$

であるから，この式において $y = 0$ としたときの $x$ を $x_1$ とすると，

$$x_1 = x_0 - f(x_0)/f'(x_0)$$

ニュートン法のアルゴリズムを次にあげる．

**(N0)** 解の収束判定のためのパラメータ $\epsilon$ を設定する．初期解 $x = x_0$ を与え，$k = 0$ とする．

**(N1)**
$$x_{k+1} = x_k - f(x_k) \Big/ f'(x_k) \tag{3.16}$$

**(N2)** $\|x_{k+1} - x_k\| \Big/ \|x_k\| < \epsilon$ ならば終了．そうでなければ **(N1)** にもどる．

多変数関数の場合のニュートン法は次のようになる．$\boldsymbol{x} = (x_1, \cdots, x_n)^\top$ に関する $n$ 個の非線形連立方程式

$$
\begin{gathered}
f_1(x_1, \cdots, x_n) = 0 \\
f_2(x_1, \cdots, x_n) = 0 \\
\cdots \\
f_n(x_1, \cdots, x_n) = 0
\end{gathered}
$$

を

$$\boldsymbol{f(x) = 0}$$

と書く．また，

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \cdot & \cdots & \cdot \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

と定義する．このとき，ニュートン法のアルゴリズムとして (3.16) の代わりに次の式を用いればよい．

$$\boldsymbol{x_{k+1}} = \boldsymbol{x_k} - \left( \left. \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x_k}} \right)^{-1} \boldsymbol{f(x_k)} \tag{3.17}$$

**最適化への応用**

制約のない最適化問題

$$\min_{\boldsymbol{x} \in \boldsymbol{R}^n} f(\boldsymbol{x})$$

を考察する．このとき，必要条件 $\nabla f(\boldsymbol{x}) = \boldsymbol{0}$ にニュートン法を適用すれば，

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \left(\nabla^2 f(\boldsymbol{x}_k)\right)^{-1} \left(\nabla f(\boldsymbol{x}_k)\right)^\top \tag{3.18}$$

を繰り返し計算することになる．

**注意．** ニュートン法は初期値のとり方によっては収束しないこともあるが，収束ははやい．また，上の行列は正則であるものと仮定している．

### 3.5.2 区間縮小法による 1 次元探索

1 変数関数の最小値を求めるために，最適値が含まれている区間を逐次縮小していく区間縮小法がよく用いられる．ここでは，黄金分割によるアルゴリズムを紹介する．

いま，$f(x)$ が単峰性 (unimodal) 関数，すなわち，

(i) $\bar{x} = \arg\min_{x \in \boldsymbol{R}} f(x)$ が定まり，

(ii) $x < \bar{x}$ では $f(x)$ は単調減少

(iii) $x > \bar{x}$ では $f(x)$ は単調増加

であるとする．

そこで，初期値 $a, b$ が $\bar{x} \in [a, b]$ を含むようにとられていると仮定する．そこで，$x_1, x_2 \in [a, b]$, $x_1 < x_2$ ととる．

**区間縮小法**

(A) $f(x_1) \leq f(x_2)$ ならば，$\bar{x} \in [a, x_2]$ なので，$b \leftarrow x_2, x_2 \leftarrow x_1$ とし，$\underline{x_1 \text{ を新たに決める}}$

(B) $f(x_1) > f(x_2)$ ならば，$\bar{x} \in [x_1, b]$ なので，$a \leftarrow x_1, x_1 \leftarrow x_2$ とし，$\underline{x_2 \text{ を新たに決める}}$

そこで，下線部分：$x_1$ あるいは $x_2$ をどのように決めるかを問題とする．

有力な考え方として，区間が一定の比率で減少するようにすれば全体として区間縮小の効率がよいことに注意しよう．

$$\frac{x_2 - a}{b - a} = \frac{b - x_1}{b - a} = \frac{x_1 - a}{x_2 - a} = \frac{b - x_2}{b - x_1} = \tau$$

とおき，一般性を失うことなく，$[a, b] = [0, 1]$ とする．これから，$\tau^2 + \tau = 1$ を得るから，これを解いて

$$\tau = \frac{\sqrt{5} - 1}{2} = 0.618$$

この値は黄金分割比として知られている．

アルゴリズムとしては，(A) が成り立つ場合，新たな $x_1$ を $\hat{x}_1$ とすれば，

$$\hat{x}_1 = a + (b - a)(\tau - \tau^2) = a + (b - a)\tau^3$$

となる．(B) の場合も同様に新たな $x_2$ を $\hat{x}_2$ として

$$\hat{x}_2 = b - (b - a)\tau^3$$

と決めることができる．これを区間 $[a, b]$ が十分小さくなるまで繰り返し，

$$\bar{x} = \frac{a + b}{2}$$

とすればよい．

### 3.5.3 最急降下法

多変数の制約のない最小化問題について考察する．この節で述べる最急降下法は，効率はあまり良くないとされているが，単純なアイデアによっているため，応用範囲は広い．

$f(\boldsymbol{x})$ は $\boldsymbol{x} = (x_1, \cdots, x_n)^\top$ の関数 ($f\colon \boldsymbol{R}^n \to \boldsymbol{R}$) と仮定し，

$$\min_{\boldsymbol{x} \in \boldsymbol{R}^n} f(\boldsymbol{x})$$

を考える．

いま，次のような一般的な繰り返し計算法を考える．

**(D0)** 初期解 $\boldsymbol{x} = \boldsymbol{x}_0$ を与える．$k = 0$ とおく．次の **(D1), (D2)** を収束条件が満たされるまで繰り返す．

**(D1)** <u>探索方向 $\boldsymbol{d}_k$ を定める</u>.

**(D2)**

$$t_k = \arg\min_{t \in \boldsymbol{R}} f(\boldsymbol{x}_k + t\boldsymbol{d}_k)$$

を計算し，$\boldsymbol{x}_{k+1} \leftarrow \boldsymbol{x}_k + t_k \boldsymbol{d}_k$ とする．

**(D3)** $k \leftarrow k + 1$ とする．

なお，収束条件は先に挙げたように，ある $\epsilon > 0$ に対して $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\| / \|\boldsymbol{x}_k\| < \epsilon$ が満たされることとする（他の判定条件もあるが，省略する）．

ここで，**(D2)** では先ほどの 1 次元探索技法が使える．従って，**(D1)** における探索方向をどのように決めるかが問題となる．

最急降下法 (steepest descent method) においては探索方向は

$$\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)^\top \tag{3.19}$$

と決められる．すなわち，$f(\boldsymbol{x})$ が形成する等高線と垂直方向に探索を行う．

なお，非線形計画法全般に関する詳しい参考書として，今野，山下 [3] があげられる．

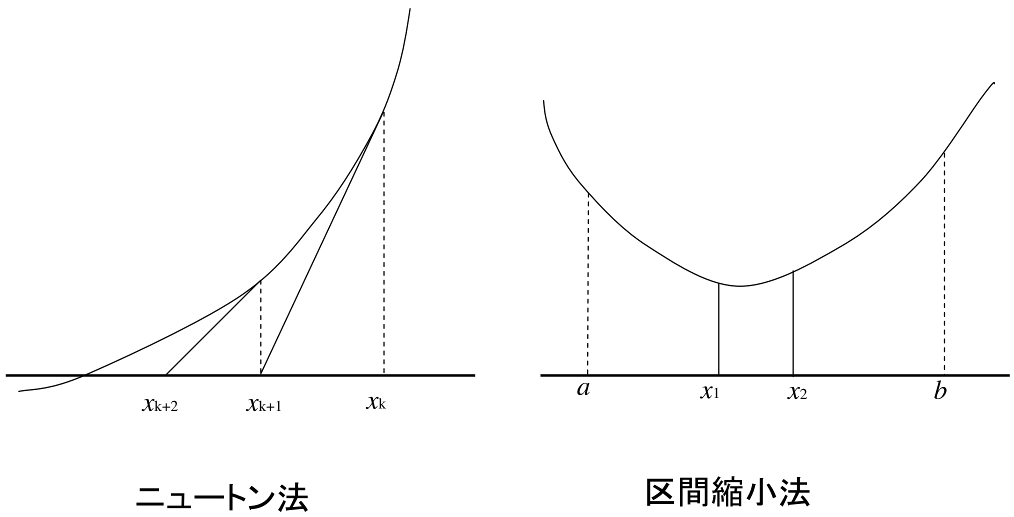



図 3.1: ニュートン法と区間縮小法

# 第4章　組み合わせ最適化問題

これまで扱ってきた最適化問題

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & (\text{または maximize} \quad f(\boldsymbol{x})) \\ \text{subject to} & g_i(\boldsymbol{x}) \le 0, \quad i=1,\dots,m & \\ & h_j(\boldsymbol{x}) = 0, \quad j=1,\dots,\ell. & \end{array}$$

では，変数 $x_i$ $(i=1,\dots,n)$ は実数値をとると仮定されていた．ところが，組み合わせ最適化問題では,「変数は整数値をとる」というように制約を受け，連続的に変化することはできない．このとき，最適化問題の解法には様々な困難さが生じる．

まず，2つの代表的な組み合わせ最適化問題を挙げよう．

**ナップザック問題 (N)．**　シンドバッドは宝物殿にはいり，なんでも好きなものをもっていってよいということになった．宝物にはすべて値札と重さが示されている．シンドバッドはナップザックに宝物を詰めていこうと思うが，総重量の制限があって，制限以上には持ち出せない．

宝物を $1,\dots,n$ とし，それぞれの価値を $c_1,\cdots,c_n$, 重さを $d_1,\cdots,d_n$(キロ) とする．重量制限は $b$(キロ) 以下とする．$x_i=1$ を宝物 $i$ を詰めて持ち出すこと，$x_i=0$ は宝物 $i$ を持ち出さないことに相当するとする．このとき，問題は明らかに次の形に定式化される．

$$\sum_{i=1}^{n} c_i x_i \to \max \tag{4.1}$$

$$\sum_{i=1}^{n} d_i x_i \le b \tag{4.2}$$

$$x_i = 1 \quad \text{OR} \quad x_i = 0 \quad i=1,\dots,n \tag{4.3}$$

この問題は一見簡単そうであるが，実は効率の良いアルゴリズムがありそうにない厄介な問題 (NP-完全問題) として知られている．なお，上の問題を許容集合

$$M_N = \{\, \boldsymbol{x} = (x_1,\cdots,x_n) \,:\, x_i \in \{0,1\}, i=1,\dots,n, \ \ \sum_{i=1}^{n} d_i x_i \le b \,\}$$

を用いて書くと次のように書ける．

$$\max_{\boldsymbol{x}\in M_N} \sum_{i=1}^{n} c_i x_i$$

**巡回セールスマン問題 (T)．** あるセールスマンが都市 $1, 2, \ldots, n$ をセールスのためにすべて巡回して元の都市に戻ることを命ぜられた．なるべく効率良く巡回するため，コストを最小にしたい（あるいは時間最小，総旅行距離最小でもよい）．都市 $i$ から都市 $j$ に行く際のコストは $d(i, j)$ で与えられるものとする $(1 \leq i, j \leq n)$．いま，

$$M_T = \{\, (i_1, \cdots, i_n) : (i_1, \cdots, i_n) \text{ は } (1, 2, \ldots, n) \text{ の任意の順列} \,\}$$

とおくと，巡回の際のコスト最小化問題は

$$\min_{(i_1, \cdots, i_n) \in M_T} d(i_1, i_2) + \cdots + d(i_{n-1}, i_n) + d(i_n, i_1) \tag{4.4}$$

と書くことができる．この問題も NP-完全問題として知られている．

NP-完全問題は，総当たり探索よりも効率がよいと保証される解法が存在しないと考えられている問題のクラスである．NP-完全問題については，4.4 節を参照されたい．

このような NP-完全問題については，厳密な最適解を求める手法としては複雑な手続き（分枝限定法 [4] など）のみが知られており，しかも効果的でないことも多い．そこで，ここでは厳密な解法は省略し，ヒューリスティック（発見的）解法，すなわち厳密な意味で必ず最適解を与える保証はないが，多くの場合良好な解を見出す方法を述べる．ヒューリスティック解法の代表として，グリーディアルゴリズム（貪欲算法）およびメタ戦略について述べよう．

## 4.1 グリーディアルゴリズム

グリーディアルゴリズム（greedy algorithm – 貪欲算法と訳される）は，多くの問題に適用できる単純な解法として最もわかりやすいものの一つである．ただし，グリーディアルゴリズムによる解は，他のよりこみいった解法に比べて，優ることは少ないと考えられる．しかしながら，複雑な問題については，まずためしてみることのできる解法の一つがグリーディアルゴリズムであるので，一般的な有用性は大きい．

### 一般的なグリーディアルゴリズム

**(I)** 選ばれた解のリスト $L$ を空にする（$L := \emptyset$）．

**(II)** 解の候補 $s$ を選ぶ．

**(III)** $L$ に $s$ を追加したとき，制約条件が満たされなければ終了．制約条件が満たされるならば，$L$ に $s$ を追加して **(II)** に戻る．

このアルゴリズムは，完全ではない．**(II)** でどのように解の候補 $s$ を選ぶかについては具体的な問題が与えられたとき，それに応じて考える必要がある．また，解の候補 $s$ を選ぶ方法についても，自由度がある．

**ナップザック問題 (N)．** 解の候補 $x_i$ を選ぶ方法として，それまで選んでいない宝物のなかで価値 $c_i$ が最大のものを選ぶのが自然である．この場合，グリーディアルゴリズムとは，高価なものから順に袋に詰めていって，重量をこえる寸前でやめるというだけのことである．

**最小木問題．** グラフ $G = (V(G), E(G))$ において辺 $e \in E(G)$ に実数値 $m(e)$ が付いているとする．$G$ のスパニングトリー $T$ で，$m(e)$ の値が最小になるものを求めるのが最小木問題である．クラスカルのアルゴリズムは，上記のグリーディアルゴリズムの一種である（なお，クラスカルのアルゴリズムについては，グラフ理論で講述した）．

**クラスカルの最小木アルゴリズム**

**(I)** サブグラフ $T$ を空にする（$T := \emptyset$）．

**(II)** 解の候補 $e$ として，これまでにとられていない辺で $m(e)$ が最大のものを選ぶ．$E(T) := E(T) \cup \{e\}$ と追加したとき，$T$ がサイクルをもつならば，解の候補を取り直す．$T$ がサイクルをもたないならば，辺を追加してステップ **(III)** へ．

**(III)** $T$ がスパニングトリーになれば終了．そうでなければ **(II)** に戻る．

後の例にみられるように，グリーディアルゴリズムは，一つの定まったアルゴリズムというよりは，どんどん解を取り込んでいくという考え方そのものである．

## 4.2 メタ戦略

一般に，発見的解法は個々の問題によって異なるべきであるが，多くの発見的手法に用いられる枠組みには共通なものがある．このような共通な枠組みとしての方法論をメタヒューリスティック (metaheuristic) あるいはメタ戦略 (metastrategy) という．メタ戦略については最近多くの研究がなされている．

メタ戦略の代表的な方法として局所探索，模擬焼きなまし法，タブー探索，遺伝アルゴリズムが知られている．また，最近ではこれらの方法を混合して用いる試みも行われている．以下では，これらの方法について簡単に紹介しよう．
このためにまず，近傍探索の概念について考察する必要がある．実ベクトル空間 $\boldsymbol{R}^n$ の場合，$\boldsymbol{x} \in \boldsymbol{R}^n$ の近傍とは，$\boldsymbol{x}$ を中心とする開球あるいは閉球で定義され，

$\boldsymbol{x}$ に近接した要素の全体という意味である．これに対して，組み合わせ最適化における $\boldsymbol{x} \in M$ の近傍とは，ある種の近傍内操作によって $\boldsymbol{x}$ から移ることのできる要素 $\boldsymbol{y}$ の全体である．$\boldsymbol{x}$ の近傍を $N(\boldsymbol{x})$ と表すことにする．近傍内操作は問題によって異なる．

**例 (N)** ナップザック問題では，$\boldsymbol{x} = (x_1, \cdots, x_n)$ の近傍の要素 $\boldsymbol{x}' = (x_1', \cdots, x_n')$ はある $i \in \{1, \ldots, n\}$ について $x_i = 1$ のとき $x_i' = 0$，$x_i = 1$ のとき $x_i' = 0$ と反転させ，$j \neq i$ なるすべての $j$ について $x_j' = x_j$ としたものとする．このような要素全体の集合を $N(\boldsymbol{x})$ とする．

**例 (T)** 巡回セールスマン問題におけるある解を $\boldsymbol{x} = (i_1, \cdots, i_n)$ とする．この順列の 2 つの要素 $i_k$, $i_\ell$ $(k < \ell)$ をとり，それらを交換したものを $x'$ とする．すなわち

$$\begin{aligned} \boldsymbol{x} &= (i_1, \cdots, i_k, \cdots, i_\ell \cdots, i_n) \\ &\qquad\qquad \downarrow \\ \boldsymbol{x}' &= (i_1, \cdots, i_\ell, \cdots, i_k \cdots, i_n) \end{aligned}$$

この操作により $\boldsymbol{x}$ から移ることのできる $\boldsymbol{x}'$ の全体を近傍 $N(\boldsymbol{x})$ とする．(注：この操作は数学の用語では互換–transposition–と呼ばれる．)

第 1 章で述べた局所的最適解の概念は $N(\boldsymbol{x})$ を用いることによって組み合わせ最適化に適用することができる．すなわち，$\hat{\boldsymbol{x}} \in M$ が目的関数 $f$ の局所的最適解であるとは，任意の $\boldsymbol{x} \in N(\hat{\boldsymbol{x}}) \cap M$ に対して

$$f(\hat{\boldsymbol{x}}) \leq f(\boldsymbol{x})$$

を満たすことをいう．局所的最適解に対して，大域的最適解 $\bar{\boldsymbol{x}}$ が

$$f(\bar{\boldsymbol{x}}) \leq f(\boldsymbol{x}), \quad \text{for all} \ \ \boldsymbol{x} \in M$$

を意味することは明らかであろう．

## 4.3 メタ戦略の諸手法

### 局所探索

まず, 局所探索 (local search) のアルゴリズムを掲げよう.

**L0.** 初期解 $x$ を選ぶ.

**L1.** 終了の条件が満たされるまで, **L2** を繰り返す.

**L2.** $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_r \in N(\boldsymbol{x})$ を選び, そのなかで最も良い解を

$$\boldsymbol{z}_q = \arg \min_{\ell=1,\ldots,r} f(\boldsymbol{z}_\ell)$$

とする. $f(\boldsymbol{z}_q) < f(\boldsymbol{x})$ ならば, $\boldsymbol{x} := \boldsymbol{z}_q$ とする. そうでなければ, 近傍内に現在の解より良い解がないので終了する.

**End of L.**

**注意.** **L2** における $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_r$ は一般に $N(\boldsymbol{x})$ の要素をすべて取り尽くしているとは限らない. $N(\boldsymbol{x})$ から $r$ 個の要素をランダムに選ぶ場合も多い.

局所探索では, ある解 $\boldsymbol{x}$ の近傍にその解よりも優る解がないとき, 探索はそこで停止する. このような場合, $\boldsymbol{x}$ は局所最適解である. このように局所探索は局所最適解から脱出できないという欠点がある. この欠点を緩和するために, いくつかの異なる初期解から出発して複数の局所最適解を求め, その中で最も良い解をとる方法がある. この方法はマルチスタート局所探索 (multi-start local search) と呼ばれる.

### 模擬焼きなまし法

模擬焼きなまし法 (simulated annealing) は局所最適解からの脱出を確率のメカニズムを利用することによって行う. 従って, アルゴリズムにおいて確率を利用するときは, 乱数を用いる. 脱出の確率は目的関数の値と '温度' に対応するパラメータの指数関数で決定される. 時間が経過するにつれて温度は低くなり, それに従って脱出の確率も低くなる. 次のアルゴリズムでは, アルゴリズムの任意の時点においてそれまでに発見された最良の解を $\tilde{x}$ で表す. また, $\gamma$ $(0 < \gamma < 1)$ は温度の低下を制御するパラメータである. アルゴリズムは終端温度 $t_T$ に達するまで繰り返される.

**S0.** 初期解 $x$ と初期温度 $t$ を設定し, $\tilde{\boldsymbol{x}} := \boldsymbol{x}$ とする.

**S1.** **S2**, **S3** を $t \leq t_T$ となるまで繰り返す.

**S2.** ステップ **S2.1**, **S2.2** を $L$ 回繰り返す.

**S2.1.** $\boldsymbol{z} \in N(\boldsymbol{x})$ を選ぶ

**S2.2.** $f(\boldsymbol{z}) \leq f(\boldsymbol{x})$ ならば $\boldsymbol{x} := \boldsymbol{z}$ とし, さらに $f(\boldsymbol{x}) < f(\tilde{\boldsymbol{x}})$ ならば $\tilde{\boldsymbol{x}} := \boldsymbol{x}$ とおく. $f(\boldsymbol{z}) > f(\boldsymbol{x})$ ならば 確率 $\exp\left(\frac{f(\boldsymbol{x})-f(\boldsymbol{z})}{t}\right)$ で $\boldsymbol{x} := \boldsymbol{z}$ とする.

**S3.** $t := \gamma t$ とする.

**End of S.**

## タブー探索

局所探索での問題点である局所最適性に対して, タブー探索 (tabu search) では決定論的な解決を試みる．すなわちある解が局所最適解であったとしても, 常にそこから脱出する．脱出した後, もとの局所最適解に戻らないメカニズムが必要になる．タブー探索では, タブーリストと呼ばれる集合を設け, その中の解は禁止することによって同じ解を繰り返し探索することがないようにする．

次に述べる基本的なタブー探索のアルゴリズムではタブーリスト (タブー集合) $T$ を用いている．

**T0.** 初期解 $\boldsymbol{x}$ と初期タブーリスト $T$ を設定する．($T = \emptyset$ としてよい．) $\tilde{\boldsymbol{x}} := \boldsymbol{x}$ とおく．

**T1.** **T2** を終了条件が満たされるまで繰り返す．

**T2.** $N(\boldsymbol{x}) - T = \emptyset$ のとき, 終了条件が満たされているか調べる．満たされていなければ, $T$ を更新する．

$\boldsymbol{z}_1, \ldots, \boldsymbol{z}_r \in N(\boldsymbol{x}) - T$ を選ぶ．

$$\boldsymbol{z}_q = \arg\min_{\ell=1,\ldots,r} f(\boldsymbol{z}_\ell)$$

に対して $\boldsymbol{x} := \boldsymbol{z}_q$ とする．$f(\tilde{\boldsymbol{x}}) \geq f(\boldsymbol{x})$ ならば $\tilde{\boldsymbol{x}} := \boldsymbol{x}$ と最良解を更新する．$T$ を更新する．($T$ には現在の解より前の解がいくつか含まれるようにする．)

**End of T.**

終了の条件としては, 規定された回数より繰り返しが多いこと, 最良解 $\tilde{\boldsymbol{x}}$ が更新されなくなってからの繰り返しが一定回数より多いことなどがある．

## 遺伝アルゴリズム

遺伝アルゴリズム (genetic algorithm–GA と略すことがある) は, 複数の解によって構成される母集団を保持し, それらを突然変異および交叉させることによって集団の多様性を保ちながら, 最適解を選択していく手法である．この方法は, 生物の進化論の類推にもとづくパラダイムによっている点, 他の手法と異なる特徴がある．

遺伝アルゴリズムでは，染色体とも呼ばれる要素 $x$ の集合を母集団 $P$ と呼び，母集団を形成する要素に対して突然変異および交叉と呼ばれる操作を行う．これらの操作および要素の選択によって，母集団を構成する要素は変化し，次の世代の母集団を形成する．従って，母集団は世代 $k = 0, 1, 2, \ldots$ の関数 $P(k)$ である．従って，遺伝アルゴリズムを次のように一般的に表すことができる．

**(GA0)** 初期母集団 $P(0)$ を与える．$k = 0$ とおく．

**(GA1)** $P(k)$ の要素に交叉および突然変異を行い，得られた要素の全体を $C$ とする（交叉および突然変異については後で説明する）.

**(GA2)** $P(k) \cup C$ から $P(k)$ 個の要素を選択し，$P(k+1)$ とする（選択の方法については後で説明する）.

**(GA3)** アルゴリズムの終了条件が満たされれば終了する．そうでなければ $k \leftarrow k+1$ とし，**(GA1)** に戻る.（終了条件としては，予め決められた繰り返し回数を超えたとき，過去何世代かにわたって母集団における最良解の改善がみられないときなどがある.）

**突然変異**

遺伝アルゴリズムにおける突然変異とは，ある要素 $\boldsymbol{x} \in P$ から別の要素を生成する操作である．典型的な操作は $\boldsymbol{x}$ の近傍の要素を一つ選ぶことである．たとえば，ナップザック問題では $\boldsymbol{x} = (x_1, \cdots, x_n)$ のある成分を $0 \leftrightarrow 1$ と反転させる操作，巡回セールスマン問題では $\boldsymbol{x} = (i_1, \cdots, i_n)$ の 2 つの成分 $i_k$, $i_\ell$ を交換する操作は突然変異とみなすことができる.

**交叉**

突然変異は, 近傍の解を選ぶ操作と考えられ，他のメタ戦略と共通のアイデアによっているが, 交叉は 2 つの解 $\boldsymbol{x}, \boldsymbol{x}'$ から新しい解（$\boldsymbol{y}, \boldsymbol{y}'$ と書く）を作り出す作用であり遺伝アルゴリズムに独特の操作である. 交叉には，様々な方法がある．ここでは，ナップザック問題と先の巡回セールスマン問題を用いて例示しよう.

**例** (N)　ナップザック問題では $0, 1$ の列からなるベクトル $\boldsymbol{x} = (x_1, \cdots, x_n)$, $\boldsymbol{x}' = (x'_1, \cdots, x'_n)$ から新たな解を作るとき，まず，ある成分 $x_i$ を決める．この成分の前後を入れ替えて $\boldsymbol{y}, \boldsymbol{y}'$ を作る．すなわち

$$\begin{array}{cc}
\boldsymbol{x} = (x_1, \cdots, x_i, x_{i+1}, \cdots, x_n), & \boldsymbol{x}' = (x'_1, \cdots, x'_i, x'_{i+1}, \cdots, x'_n) \\
\Downarrow & \\
\boldsymbol{y} = (x_1, \cdots, x_i, x'_{i+1}, \cdots, x'_n), & \boldsymbol{y}' = (x'_1, \cdots, x'_i, x_{i+1}, \cdots, x_n)
\end{array}$$

たとえば，$\boldsymbol{x} = (1, 0, 0, 1, 1, 1, 0)$, $\boldsymbol{x}' = (0, 1, 1, 0, 1, 0, 1)$ とし，第 3 成分の後で交叉を行うとすると

$$\begin{array}{cc}
\boldsymbol{x} = (1, 0, 0 \,|\, 1, 1, 1, 0), & \boldsymbol{x}' = (0, 1, 1 \,|\, 0, 1, 0, 1) \\
\Downarrow & \\
\boldsymbol{y} = (1, 0, 0 \,|\, 0, 1, 0, 1), & \boldsymbol{y}' = (0, 1, 1 \,|\, 1, 1, 1, 0)
\end{array}$$

が得られる．この方法は 1 点を定めてその前後で交叉させるため，1 点交叉と呼ばれる.

**例** (T) 巡回セールスマン問題では，解が順列で表現されるため，交叉は前の例よりも複雑になる．すなわち，交叉を行った結果も順列である必要がある．ここでは，例を用いて説明しよう．2 つの順列

$$\boldsymbol{x} = (2, 3, 9, 6, 1, 7, 8, 5, 4), \quad \boldsymbol{x}' = (7, 4, 1, 2, 5, 6, 3, 9, 8)$$

から $\boldsymbol{y}, \boldsymbol{y}'$ を作るものとする．まず，交叉させる 1 点を決める必要があるが，これを第 5 成分の後とする．$\boldsymbol{y}, \boldsymbol{y}'$ の第 1〜5 成分はそれぞれ $\boldsymbol{x}, \boldsymbol{x}'$ の第 1〜5 成分を継承するものとする．すなわち，$\boldsymbol{y} = (2, 3, 9, 6, 1, *, *, *, *)$，$\boldsymbol{y}' = (7, 4, 1, 2, 5, *, *, *, *)$ 次に $*$ で表された部分を決めるが，$\boldsymbol{y}$ ではこの部分は数字 $7, 8, 5, 4$ でなければならない．そこで，これらの数字を $\boldsymbol{x}'$ におけるそれらの順序に従ってならべると $\boldsymbol{x}' = (\underline{7}, \underline{4}, 1, 2, \underline{5}, 6, 3, 9, \underline{8})$ の下線部の順序によって $7, 4, 5, 8$ と並べ替えることができる．よって，

$$\boldsymbol{y} = (2, 3, 9, 6, 1 \,|\, 7, 4, 5, 8)$$

とする．同様に，$\boldsymbol{y}'$ における $6, 3, 9, 8$ の順序は $\boldsymbol{x} = (2, \underline{3}, \underline{9}, \underline{6}, 1, 7, \underline{8}, 5, 4)$ を用いて

$$\boldsymbol{y}' = (7, 4, 1, 2, 5 \,|\, 3, 9, 6, 8)$$

とする．

**選択**

母集団の各個体には適合度と呼ばれる評価値が与えられる．最適化問題の場合，適合度は目的関数の値をそのまま用いるのが最も単純であるが，選択の方法を考慮すると，目的関数の値を変換するほうがより一般的である．当然ではあるが，目的関数の値がより良いほど適合度がより高いように変換される．

遺伝アルゴリズムでは，母集団から突然変異と交叉によって生成された要素の集合ともとの母集団とを合わせると要素数は増加しているので，もとの母集団の大きさ（$N$ と書く）を維持するために選択を行う必要がある．一般的な方法は，適合度の高いものから順に $N$ 個とる方法，適合度が高いものほど選択される確率が高くなるように乱数を発生させて個々の要素を選択する方法，あるいはそれらの組み合わせなどがある．

遺伝アルゴリズムは，他のメタ戦略にもまして研究が盛んであり，多くの変形や拡張，さらには他のメタ戦略との混合なども考察されている．その理由の 1 つとして，遺伝アルゴリズムが複雑な構成をもっているため，工夫をこらすほど性能が向上することが期待できることがあげられる．しかしながら，遺伝アルゴリズムが注目されている主な理由は他にある．それは，このアルゴリズムが生物学の遺伝法則の類推に基づいているため，生物における学習や自然淘汰の概念が遺伝アルゴリズムにおいて利用でき，その一方で，生物学への貢献も期待できるのではないか，などの問題提起がなされているからである．このように，遺伝アル

ゴリズムは科学・工学における大きなアイデアの変化（パラダイム）にかかわっており，それゆえ多くの研究者に注目されている．情報科学では，遺伝アルゴリズムのアイデアから生じた計算機構が進化計算 (evolutionary computing) として目下研究されている．

遺伝アルゴリズムについては，表面的な紹介にとどまった．興味のある読者は遺伝アルゴリズムや進化計算の入門書，専門書 [4] を参照されたい．

## 4.4 補足–NP 完全性

ソーティングなどで計算時間のオーダーに言及したが、このような形式による計算効率の評価を計算複雑さ (computational complexity) の考察という．

計算複雑さの議論の中心となるのが NP 完全性の理論である．以下にこの理論のあらましを述べよう．なお，この節は，萩原, 西原 [5],pp. 15–20 を参考にした．

**計算機科学最大の問題: P=NP ?**
ソーティングや最短路などの問題は要素の数に比べて比較的短い時間で解けるが、それに対して次に述べるナップザック決定問題や充足可能性問題は最悪の場合、計算複雑さが変数の指数オーダーになりそうである．

実は前の問題は多項式オーダーで解けるという意味でクラス P に属するといい、後の問題は後述の意味で非決定的計算モデルで多項式オーダーであるという意味で、クラス NP に属するという．また、クラス NP の中で、以下に述べる意味で最も難しいクラスの問題を NP 完全という．

クラス NP は P を含むことは以下にみるように容易にわかる．その逆、NP に属する問題が P にも属するかどうかが大問題である．一見したところクラス NP は P よりもはるかに広いようだが、実は問題「P=NP ?」は肯定的にも否定的にも解決されていない．

**充足可能性問題:** $x_1, x_2, \cdots, x_n$ がブール値変数であるとする．これらの論理関数

$$f(x_1, x_2, \cdots, x_n) = (\ell_i \wedge \cdots \wedge \ell_j) \vee (\ell_p \wedge \cdots \wedge \ell_q) \vee \cdots \vee (\ell_s \wedge \cdots \wedge \ell_t)$$

（ただし、$\ell = x$ あるいは $\ell = \neg x$）が $x_1, x_2, \cdots, x_n$ の値 (0/1) を適当に選ぶことによって $f(x_1, x_2, \cdots, x_n) = 1$ となり得る（充足可能）かどうか判定せよ．この問題は NP 完全性の理論においてキーの役割を果たす．

0/1 **ナップザック決定問題:** 与えられた実数 $Y, Z$ について、次の式を満たす変数の

組 $x_1, \cdots, x_n$ があるかどうかを決定する問題を 0/1 ナップザック決定問題という．

$$\sum_{\ell \le i \le j} p_i x_i \le Y$$
$$\sum_{\ell \le i \le j} w_i x_i \le Z$$
$$x_i = 0 \text{ or } x_i = 1, \quad \ell \le i \le j$$

この問題も NP 完全であることが知られている．

**計算のモデル:** 計算複雑さに関して厳密な議論を行うには、どのような計算のモデルを用いるのかを規定しなければならない．我々が通常使用する計算機にはメモリーの制限がある、アクセスに時間がかかる場合とそうでない場合がある、など理論的考察には障害となる様々な制約がある．そこで、計算のモデルとして理想的な計算機が考えられている．通常考察されるのは、Turing 機械である．Turing 機械に関する記述は省略するが、我々が通常使用するコンピュータにメモリーの制約をなくするなどの理想化を行ったモデルと同等であることがわかっている (エイホ他, アルゴリズムの設計と解析 I, pp. 4–30)．ここで Turing 機械というときは、上記のように理想化された通常のコンピュータのモデルであると考えることにしよう．

**クラス P:** ある問題を記述するとき、問題の規模、たとえば変数やデータの数に対して計算量を評価する．問題の規模を $n$ と書く．ある問題を解くアルゴリズムが問題の規模の多項式オーダーの時間であるとき、その問題はクラス P に属するという．より正確にいえば、「あるプログラムと正数 $p$ があって, その問題の任意の例 (個別問題) について計算量が $O(n^p)$ であるとき、その問題はクラス P に属するという.」

ソーティングの問題には最悪の場合でも $O(n \log n)$ のアルゴリズムがあるから、クラス P にはいる．クラス P という分け方は大ざっぱで、$O(n^2)$ でも $O(n \log n)$ でもクラス P であることに変わりはない．

**非決定性機械とクラス NP:** 通常のプログラミング言語に次の命令を追加する．

$$\text{choice}(L_1, L_2, \cdots, L_n)$$

これは、ラベル $L_1, L_2, \cdots, L_n$ のどれかを選び、そのラベルがついた文に制御を移す命令である．どのラベルを選ぶかの基準は与えられないので、この命令を用いるアルゴリズムは非決定的である．

いま、アルゴリズムの終了が yes か no かどちらかの判定であると仮定する．(答が yes か no かの判定である問題を決定問題あるいは判定問題という．)．プログラムでこの命令に出会うと、すべての選択を並列に実行すると考え、すべての選択肢のどれかでアルゴリズムが yes の答が出たとき、この問題の答が yes であるとし、選択肢のすべてにおいて no のとき、問題の答は no であるとする．

ある問題（の任意の例）が、上記の非決定的アルゴリズムによって多項式オーダーの時間で解けるとき、その問題はクラス NP に属するという．

非決定的アルゴリズムとは、無限個数の並列機械があって、解の候補をすべて並列に同じアルゴリズムで処理することができる、と考えてもよい．

別の言い方をすると、クラス NP の問題とは、「ある解の候補が任意に与えられたとき、通常の決定的機械によって、それが実際に解であるかどうかを多項式オーダーの時間で確かめることができる」と説明できる．

先の 2 つの問題がクラス NP に属することは容易にわかる．充足可能性問題では、$x_1, \cdots, x_n$ にすべての 0/1 の組み合わせを与え、その中に問題の条件 $f(x_1, \cdots, x_n) = 1$ を満たすものがあるかどうかチェックすればよい．ナップザック決定問題では、やはり、変数 $x_1, \cdots, x_n$ にすべての 0/1 の組み合わせを与え、その中に、問題の制約条件を満たすものがあるかどうか調べればよい．これらのチェックが多項式時間でできることは明らかである．

**多項式還元可能性:** この定義は NP 完全性を論じる前提として必要である．決定問題 $Q_1$ と $Q_2$ に対して、$Q_1$ から $Q_2$ へ多項式還元可能とは次の 2 つの条件が満たされる場合をいう．

**i)** $Q_1$ の任意の個別問題 $q$ を $Q_2$ の問題 $f(q)$ に変換する多項式時間（決定的）アルゴリズムが存在する．

**ii)** $q$ の答が yes のときかつそのときに限り $f(q)$ の答が yes である．

$Q_1$ から $Q_2$ へ多項式還元可能であるとき、$Q_1 \propto Q_2$ と書く．

このことは、

$$(q\text{ を解く手続き}) = (q \to f(q)\text{ の変換手続き}) + (f(q)\text{ を解く手続き})$$

従って、$Q_1$ を解くのは ($Q_1$ の問題規模の多項式時間) + ($Q_2$ を解く時間) でできることを意味する．(決定的多項式時間を無視すれば、$Q_1$ は高々$Q_2$ と同等の時間で解ける．従って、$Q_2$ は $Q_1$ と同等あるいは $Q_1$ よりも難しい．)

次の性質がある．

a. $Q_1 \propto Q_2$ のとき、$Q_2 \in$P ならば $Q_1 \in$P.

b. $Q_1 \propto Q_2$ かつ $Q_2 \propto Q_3$ ならば $Q_1 \propto Q_3$.

問題 $Q$ があって、NP に属する任意の問題が $Q$ に還元可能であるとき、$Q$ は NP 困難 (NP hard) であるという．($Q$ は NP のどの問題よりも難しい．) NP 困難でかつ NP に属する問題を NP 完全（NP complete）という．(そんな問題がそもそも存在するのかと思うだろうが、実は、応用上重要な多くの問題が NP 困難あるいは NP 完全である．)

c. $Q$ が NP 完全かつ $Q \propto S$ のとき、$S$ は NP 困難.

d. $Q$ が NP 困難あるいは NP 完全で $Q \in$P ならば P=NP.

**Cook の定理:** 充足可能性問題は NP 完全である．

(証明は難しい．省略するが、証明の方針は、任意の非決定的プログラムを充足可能性問題に還元できることを示すことによる．そのために、非決定的プログラムの各命令と各ステップを充足可能性問題でシミュレートする．)

次に、ある問題 $Q'$ が NP 完全であることを示すには、$Q'$ が NP に属し、かつ、それまでに NP 完全であることが示された問題 $Q$ が $Q'$ に還元可能であることを示せばよい．($Q'$ を解くアルゴリズムをサブルーチンに使って $Q$ を解く．)

この方法で、多くの決定問題が NP 困難あるいは NP 完全であることが知られている．また、上の定義からすべての NP 完全問題は同等の難しさをもつ．興味深いのは、効率の良いアルゴリズムが見つかっていない問題で応用上重要な問題の多くが NP 困難あるいは NP 完全であることである．(なお、実際的な注意として、ある問題が NP 完全か NP 困難かは微妙なことがあるので、NP 困難といっておくほうが無難かも知れない．)

**決定問題と最適化問題:** 例として 0/1 ナップザック問題には最適化問題 (例 5.6) と 0/1 ナップザック決定問題があったことを思い出そう．前者は最適解を見いだす問題で、後者は答が yes か no かである．NP に関する議論では、決定問題を考察するが、理論的には、「最適解の充分な近似」を認めることにすれば、計算量の問題からは、決定問題と最適化問題の間には余り差はない．たとえば、次の最適化問題を考える．

$$\begin{aligned}&\text{minimize } f(x_1,\cdots,x_n)\\&\text{subject to } g(x_1,\cdots,x_n)\le Z\\&x_i=0 \text{ or } 1,\quad i=1,\cdots,n\end{aligned}$$

これに対する決定問題は次の式を満たす $x_1,\cdots,x_n$ が存在するかどうかを判定する問題となる．

$$\begin{aligned}&f(x_1,\cdots,x_n)\le Y\\&g(x_1,\cdots,x_n)\le Z\\&x_i=0 \text{ or } 1,\quad i=1,\cdots,n\end{aligned}$$

決定問題を多項式回数繰り返せば、最適化問題の解を充分な精度で近似することができる．そのための一方法としては、まず、no の答が出るような充分小さな $Y_1$ と yes の答が出るような大きな $Y_2$ を見いだし、$Y_3=(Y_1+Y_2)/2$ として、決定問題を解く．答が no なら、$Y_4=(Y_2+Y_3)/2$ として $[Y_3,Y_2]$ を調べる．答が yes なら $Y_4=(Y_1+Y_3)/2$ として $[Y_1,Y_3]$ を調べる (はさみうち法)．これを繰り返すと区間

が前の 1/2 に逐次狭まり、最適解はこの区間に含まれているので、最適解を近似することができる．$M$ 回繰り返すと初めの区間の $2^{-M}$ に縮小されることに注意しよう．

# 関連図書


[1] 坂和正敏，経営数理システムの基礎，森北出版，1991.

[2] 福島雅夫，数理計画入門，朝倉書店，1996.

[3] 今野浩，山下浩，非線形計画法，日科技連，1978.

[4] 茨木俊秀，離散最適化法とアルゴリズム，岩波書店，1993.

[5] 萩原宏, 西原清一, 現代データ構造とプログラム技法, オーム社, 1987.