

## 12 群 (電子情報通信基礎) - 2 編 (離散数学)

## 6 章 数理計画法

(執筆: 田村明久) [2009 年 3 月受領]

## 概要

最適化問題 (optimization problem) は形式的には以下のように記述される。与えられた空間  $X$ , その部分集合  $S$  及び  $X$  上で定義された実数値関数  $f$  に対して,

$$\begin{aligned} \text{目的: } & f(x) \rightarrow \text{最大化 (あるいは最小化)} \\ \text{制約: } & x \in S \end{aligned} \quad (6.1)$$

ここで  $S$  は制約を満たす  $X$  の要素の集合を表し, 許容領域 (feasible region) あるいは実行可能領域と呼ばれる。また  $S$  の要素は許容解 (feasible solution) あるいは実行可能解と呼ばれる。  $S$  が非空であるとき最適化問題は実行可能であるといい, そうでないとき実行不可能であるという。関数  $f$  は目的関数 (objective function) と呼ばれる。  $S$  上で  $f$  をいくらでも大きく (最小化の場合は小さく) できるとき, 最適化問題は非有界であるという。許容解  $x^*$  が条件

$$f(x^*) \geq f(x) \quad (\forall x \in S) \quad (\text{最小化の場合は } f(x^*) \leq f(x) \quad (\forall x \in S)) \quad (6.2)$$

を満たすとき,  $x^*$  を最適解 (optimal solution),  $f(x^*)$  を最適値 (optimal value) という。また最適解が存在しない場合でも, 最適化問題が実行可能で有界であるならば,  $\sup\{f(x) \mid x \in S\}$  (最小化の場合は  $\inf\{f(x) \mid x \in S\}$ ) を最適値という。最適化問題 (6.1) は, その実行可能性や有界性を判定し, 最適解 (あるいは最適値) が存在する場合はこれを計算することを求めている。数理計画 (mathematical programming) とは, 数学的に記述された最適化問題の構造を解析し, 最適解やその近似解を計算するアルゴリズムの開発やそのソフトウェア化を目指す研究分野である。

数理計画で扱われる最適化問題は,  $X, S, f$  の形態により分類される。  $X, S, f$  がグラフ, マトロイドや劣モジユラ関数などの離散構造を用いて記述されるとき, 組合せ最適化問題 (combinatorial optimization problem) や離散最適化問題 (discrete optimization problem) と呼ばれる [本編 4 章, 5 章参照]。本章では, 全空間  $X$  を  $n$  次元実線形空間  $R^n$  とし, 制約が有限個の関数  $g_i: R^n \rightarrow R$  ( $i = 1, 2, \dots, m$ ) と  $R^n$  の部分集合  $U$  を用いて表現される最適化問題

$$\begin{aligned} \text{目的: } & f(x) \rightarrow \text{最大化 (あるいは最小化)} \\ \text{制約: } & g_i(x) \leq 0 \quad (i = 1, 2, \dots, m) \\ & x \in U \end{aligned} \quad (6.3)$$

を扱う。ここで  $U$  は整数性などの不等式を用いて表現しにくい制約を表す。例えば,  $Z^n$  をすべての成分が整数である  $n$  次元ベクトル全体からなる集合としたとき, 整数性は  $U = Z^n$  と記述できる。問題 (6.3) において,  $f$  が線形関数, 各  $g_i$  がアフィン関数 ( $g_i(x) = \sum_{j=1}^n a_{ij}x_j - b_i$ ) で  $U = R^n$  であるとき, それを線形計画問題 (linear programming problem) という。ただし,  $x_j$  は  $x$  の第  $j$  成分である。幾何的には, 線形計画問題は凸多面体上で線形関数を最大化

(あるいは最小化)する問題である。線形計画問題に対しては、多項式時間解法 (polynomial time algorithm)\*が提案されている。線形計画問題に、全部あるいは一部の 변수が整数という制約を追加したものを整数計画問題 (integer programming problem) という。特に、変数を 0 か 1 に制限したものを 0-1 整数計画問題 (0-1 integer programming problem) という。組合せ最適化問題の多くは、0-1 整数計画問題としても記述できる。整数計画問題は難しい問題の代表例である。問題 (6・3) において、 $f$  と各  $g_i$  が凸関数、 $U$  が凸集合であり、最小化問題であるものを凸計画問題 (convex programming problem) という。凸計画問題は、凸集合上で凸関数を最小化する問題で、線形計画問題を包含する。凸計画問題では「極小解 = 最小解」という性質があるため、一般的に解きやすい。ここで、極小解とは適当な近傍内に制限したとき最小となる許容解のことで、大域的な最小性を要求している最小解とは異なる概念である。問題 (6・3) において、線形性を仮定しないものを非線形計画問題 (nonlinear programming problem) といひ、凸計画問題や 0-1 整数計画問題を含む。非線形計画問題の最適解を求めることは一般に難しく、多くの場合は極大解や極小解あるいは Karush-Kuhn-Tucker 条件と呼ばれる局所的条件を満たす許容解の求解が目標となる。非線形計画問題の最適解の求解法を研究する分野を特に大域的最適化という。そのほかにも数理計画には、最適性の原理に基づいた再帰方程式を構成することで最適化問題の求解を目指す動的計画 (dynamic programming)、目的関数や制約に不確実な要素を含む最適化問題を扱う確率計画、目的関数が複数存在する問題を扱う多目的計画や制約式が無限に存在する問題を扱う半無限計画などもある。数理計画全般については文献 1, 2) を参照されたい。

最適化問題に対するアプローチあるいは計算手法の研究は、以下のように分類できる。(a) 厳密解法の開発: 与えられた最適化問題に対して、最適解 (通常は一つ) を求めるアプローチで、計算量の小さいアルゴリズムの開発や実装時の高速化がテーマである。(b) 近似解法の開発: 計算量が小さくかつ最適解に近い許容解を求めるアルゴリズムの開発がテーマである<sup>3)</sup>。近似解法と呼ぶときは、最適解に対する近さの指標である近似比の解析がなされていることが一般的である。(c) 発見的解法の開発: 短時間でできる限り最適解に近い許容解を求めるアルゴリズムの開発がテーマであり、実用面で重要である。問題ごとの構造を生かし、近傍探索などの工夫をこらしたメタ解法やメタ戦略と総称される解法が多数開発されている<sup>4)</sup>。(d) 列挙解法の開発: 最適性などのある種の条件を満たす許容解をすべて求めるアルゴリズムの開発がテーマであり、出力する解一つ当たりの計算量の解析が重要な課題となる。実用面では、数理的に記述しにくい制約を含む問題に利用される。(e) 確率的解法の開発: 解法に確率的要素を加え、最適解あるいは近似解の計算の効率化を図ることがテーマである<sup>5)</sup>。(f) ロバスト解法の開発: 問題データの変化に対して、許容性や目的関数値の良さを保持する解を求める解法の開発がテーマである。需要予測などの不確実な要素を含む問題に対して有効となる。

### 【本章の構成】

組合せ最適化に関連する問題として、線形計画問題、整数計画問題、動的計画問題、特殊な凸計画問題である錐線形計画問題とその特殊ケースである半正定値計画問題を紹介する。

\*  $f$  や  $g_i$  中の係数がすべて整数であるとき、問題のデータを 2 進数表記した際のビット長  $L$  と  $n, m$  に関する多項式回の四則演算、代入演算及び比較演算で問題を解くアルゴリズムのこと。

## 12 群 - 2 編 - 6 章

## 6-1 線形計画

(執筆者: 田村明久)[2009 年 3 月 受領]

## 6-1-1 線形計画の概要

線形計画問題 (linear programming problem) は、有限個の 1 次等式、等号つき 1 次不等式を満たすベクトルのなかで線形関数を最大化あるいは最小化する最適化問題である。線形計画問題自体が実務問題を記述する場合や整数計画問題などの難しい問題を解く際の緩和問題や近似問題として利用されることが多く、数理計画において最も基本的な問題である。

線形計画問題に対する最初の実用的解法は、1947 年にダンツイク (Dantzig) により提案された単体法である (文献 6 参照) 改訂単体法や積形式などの計算技法の提案により、大規模な問題にも対応した実用的に強力な解法で、商用ソフトウェアでも実装されている。しかし計算量に関する理論的保証はない。ピボット選択規則によるバリエーションはあるが、多くのバリエーションに対して最悪時には変数の個数  $n$  と制約式の個数  $m$  に関して指数時間を要することが示されている。単体法が多項式時間解法となり得るかは未解決問題である。線形計画問題に対する最初の多項式時間解法は、1979 年にカチアン (Khachian<sup>7)</sup>) により提案された楕円体法である。楕円体法は実用には不向きであるというのが一般的認識である。しかし、いくつかの組合せ最適化に対しては既知の多項式解法が楕円体法によるものしかなく理論的な価値は薄れていない<sup>8)</sup>。また、1984 年のカーマーカー (Karmarkar<sup>9)</sup>) による多項式時間解法の提案以降、内点法と総称される解法の研究が爆発的に行われ、多くの多項式時間解法が提案された<sup>10)</sup>。内点法は、対数バリア関数やニュートン法など従来の非線形最適化で用いられている手法を利用したもので、商用ソフトウェアにも実装されている。上記以外にも計算幾何学の技法を利用した解法も提案されており、その計算時間は  $n$  に関しては高次の指数関数となるが  $m$  に関しては線形であるため、 $n$  が小さい場合には高速となる。線形計画問題に対しては、問題の入力ビット数には依存せず、 $m$  と  $n$  に関する多項式時間で線形計画問題を解く解法 (すなわち強多項式時間解法) が存在するかは、大きな未解決問題である。

本節では、双対定理、相補性定理などの諸定理を紹介し、単体法を解説する。

## 6-1-2 線形計画問題

最適化問題 (6.1) のうちで、 $m \times n$  実行列  $A$  及び  $b \in R^m$  と  $c \in R^n$  が与えられたもとの、

$$\begin{aligned} \text{目的: } & c^T x \rightarrow \text{最大化} \\ \text{制約: } & Ax = b, x \geq 0 \end{aligned} \quad (6.4)$$

と記述できる問題を線形計画問題、特に標準形線形計画問題という。ここで  $c^T$  は列ベクトル  $c$  の転置を表し、 $c, x$  の第  $j$  成分をそれぞれ  $c_j, x_j$  と表すと、 $c^T x = \sum_{j=1}^n c_j x_j$  となる。一般の線形計画問題は、次の操作を用いて標準形に変換できる。(a) 最小化問題であるならば、目的関数を  $-1$  倍することで最大化問題に変換し;(b)  $a^T x \leq \beta$  という 1 次不等式制約を、非負の変数  $z$  を左辺に加えることで、 $a^T x + z = \beta$  のように 1 次等式制約に書き換え (このような変数  $z$  をスラック変数 (slack variable) という); (c) 非負制約のない変数  $x_i$  を、 $x_i = x'_i - x''_i$ 、 $x'_i, x''_i \geq 0$  のように二つの非負変数  $x'_i, x''_i$  の差として置き換える。すなわち、数学的には標

準形のみを扱えばよい。

標準形 (6.4) に対して、一般性を失うことなく以下を仮定する。

$$\text{行列 } A \text{ の階数 } \text{rank } A \text{ が行数 } m \text{ に等しい。} \quad (6.5)$$

この仮定は 1 次従属な行を削除するなどの操作で達成できる。  $A$  の列の添字集合を  $V$  としたとき、(6.5) より  $m \times m$  小行列  $A_B$  が正則となるように  $V$  の部分集合  $B$  を取ることができる。ここで  $A_B$  は添字集合  $B$  に対応する  $A$  の列を集めてできた行列とし、 $c_B$  などと同様に定義する。このような  $B$  を基底 (basis) といい、 $N = V \setminus B$  を非基底 (nonbasis) という。  $(B, N)$  という列の分割を用いると標準形 (6.4) を

$$\begin{aligned} \text{目的: } z &\rightarrow \text{最大化} \\ \text{制約: } z &= c_B^T A_B^{-1} b + (c_N^T - c_B^T A_B^{-1} A_N) x_N \\ x_B &= A_B^{-1} b - A_B^{-1} A_N x_N \\ &x_B, x_N \geq 0 \end{aligned} \quad (6.6)$$

と書き換えられる。制約に現れる等式系は基底  $B$  に対する辞書 (dictionary) と呼ばれ、 $x$  の非負性と  $z$  を最大化することを前提とすれば、辞書の情報のみで問題 (6.6) を記述できる。  $i \in B$  に対応する変数  $x_i$  を基底変数 (basic variable)、  $j \in N$  に対応する変数  $x_j$  を非基底変数 (nonbasic variable) という。辞書の形式から、どのような非基底変数の値に対しても、等式制約を満たす基底変数の値が存在することが分かる。特に、 $x_N = 0$ 、 $x_B = A_B^{-1} b$  と定めた解を基底  $B$  に対する基底解 (basic solution) と呼び、 $A_B^{-1} b \geq 0$  が満たされるときは許容基底解 (basic feasible solution) と呼ぶ。線形計画問題の基本的な性質として以下が成り立つ。

定理： 実行可能で有界な線形計画問題は、最適基底解をもつ。

線形計画問題は連続変数をもつ問題であるが、上記の定理より基底という離散的な構造を有する最適化問題ともみなせる。

### 6-1-3 双対理論

標準形 (6.4) に対して、次のように定義する  $R^m$  上の線形計画問題を

$$\begin{aligned} \text{目的: } b^T y &\rightarrow \text{最小化} \\ \text{制約: } A^T y &\leq c \end{aligned} \quad (6.7)$$

を (6.4) の双対問題 (dual problem) といい、これと対比するために (6.4) を主問題 (primal problem) という。双対問題を標準形に書き換え、その双対問題を構成すると主問題と等価になるため、双対といわれる。更に、主問題と双対問題の間には以下の関係が成り立つ。

弱双対定理 (weak duality theorem): (6.4) と (6.7) の許容解  $x, y$  に対して、 $c^T x \leq b^T y$  が常に成り立つ。

強双対定理 (strong duality theorem): (6.4) または (6.7) の一方が最適解をもてば、他方

も最適解をもち、最適値も一致する。また (6.4) と (6.7) がともに実行可能ならば両者は最適解をもつ。

弱双対定理は簡単な計算で示すことができ、これより以下の性質が簡単に導ける。

系： 主問題と双対問題の許容解  $x, y$  が同じ目的関数値をもつならばそれぞれ最適解となる。

系： 主問題と双対問題の一方が非有界ならば他方は実行不可能である。

以下の定理は強双対定理より導出されるが、許容解の最適性を確認するために有益である。

相補性定理： (6.4) と (6.7) の許容解  $x, y$  が最適解であるための必要十分条件は相補性条件  $x^T(A^T y - c) = 0$  を満たすことである。

$x \geq 0$  と  $A^T y \geq c$  を考慮すると、相補性条件 (complementary slackness) は「すべての  $j \in V$  に対して  $x_j = 0$  または  $A_j^T y = c_j$ 」と書き換えられる。ただし  $A_j$  は  $j$  に対応する  $A$  の列である。最適解の対  $(x, y)$  は相補性条件を満たすが、更にそれぞれの  $j$  に対して  $x_j > 0$  または  $A_j^T y > c_j$  を満たすとき強相補性を満たすという。このような対は常に存在する。

定理： (6.4) と (6.7) が最適解をもつとき、強相補性を満たす最適解の対が存在する。

#### 6-1-4 単体法

単体法 (simplex method) の動きを説明するためにまずは以下の仮定をおく (仮定を外す方法は後述する)。

$$A_B^{-1}b \geq 0 \text{ である基底 } B \text{ が与えられている。} \quad (6.8)$$

$B$  に対する辞書 (6.6) より、基底解  $(x_B, x_N) = (A_B^{-1}b, 0)$  の目的関数値は  $c_B^T A_B^{-1}b$  となる。辞書 (6.6) において、目的関数の非基底変数  $x_N$  の係数ベクトルが非正である場合、すなわち、

$$c_N^T - c_B^T A_B^{-1}A_N \leq 0 \quad (\text{最適性規準}) \quad (6.9)$$

の場合は、任意の  $j \in N$  に対して  $x_j$  を単独で増加させても目的関数値は増加せず、 $(x_B, x_N) = (A_B^{-1}b, 0)$  が最適解であることが保証される。

次に最適性規準 (6.9) が満たされない場合を考える。このとき、 $c_s - c_B^T A_B^{-1}A_s > 0$  となる  $s \in N$  が存在するので、このような  $s$  を一つ選ぶ。ここでほかの非基底変数は 0 に固定したままで、 $x_s$  を単独で増やすことを考える。実行可能性を保存したままで  $x_s > 0$  とできるならば、目的関数値が厳密に増加する。 $B$  に対する辞書より、 $x_s$  が増加することで値が減少する基底変数  $x_i$  は、 $(A_B^{-1}A_s)_i > 0$  という不等式で特徴づけられる。仮に

$$A_B^{-1}A_s \leq 0 \quad (\text{非有界性}) \quad (6.10)$$

とすると、 $x_s$  をいくら増やしても実行可能性は崩れず、目的関数もいくらでも大きくできる。この場合、問題は非有界となる。一方、 $A_B^{-1}A_s \leq 0$  でない場合には、 $x_s$  を

$$\min\{(A_B^{-1}b)_i / (A_B^{-1}A_s)_i \mid i \in B, (A_B^{-1}A_s)_i > 0\} \quad (6\cdot11)$$

まで増加させたときに、(6・11)の最小値を与える  $r \in B$  において  $x_r = 0$  となり、実行可能性を保存するという意味でこれ以上は  $x_s$  を増やせない。 $r$  の候補が複数ある場合はその一つを選び、 $B = B \cup \{s\} \setminus \{r\}$  と基底を更新する。このとき、 $A_B$  が正則であること、新しい許容基底解が  $x_s$  を最大限まで増加させた許容解と一致すること、目的関数値は非減少であることが保証される。基底の更新及びそれに伴う辞書の更新をピボット演算という。最適解が得られるか非有界が判明するまで、ピボット演算を繰り返すのが単体法である。

単体法が終了したならば最適解が得られるか非有界であることが判明するが、このままでは終了する保証はなく、実際に巡回と呼ばれる無限ループに陥る可能性がある。巡回を回避するにはいくつかの方法があるが、最もエレガントな方法は最小添字規則 (smallest-subscript rule) である。最小添字規則では、 $V = \{1, 2, \dots, n\}$  と列添字を順番付け、非基底変数  $s$  の候補が複数ある場合は添字最小のものをを選び、基底変数  $r$  の候補が複数ある場合も添字が最小のものを選ぶというものである。この規則に従い基底を更新すると有限回の繰返して単体法が終了することが保証されている。

最後に仮定 (6・8) を外す方法を紹介する。一般性を失うことなく、仮定 (6・5) より基底  $B$  が与えられているとし、対応する基底解は許容解ではないとする ( $A_B^{-1}b \geq 0$  ではない)。

$$\begin{aligned} \text{目的: } & -x_a \rightarrow \text{最大化} \\ \text{制約: } & x_B = A_B^{-1}b - A_B^{-1}A_N x_N + 1x_a \\ & x_B, x_N, x_a \geq 0 \end{aligned} \quad (6\cdot12)$$

という人工変数  $x_a$  を追加した人工問題を考える。(6・12)の制約の最後の 1 は、すべての成分が 1 であるベクトルとする。(6・12)の最適値が 0 であることと、元問題 (6・4) が実行可能であることが同値である。 $x_a$  を増やすことで (6・12) の許容解が得られる。また、(6・12) で  $(A_B^{-1}b)_i$  が最小となる添字  $r \in B$  を一つ選び、 $B = B \cup \{a\} \setminus \{r\}$  とすることで (6・12) の許容基底解が得られ、単体法が (6・12) に適用できる。このように、人工問題を解くことで元問題の実行可能性を判定し、実行可能ならば元問題を解くという単体法を 2 段階単体法という。

単体法を動かすには、辞書の必要な部分のみを  $A_B^{-1}$  と元データの  $A, b, c$  から計算するだけでよい。このような考え方を導入したものを改訂単体法という。辞書は双対問題の完全な情報もち、主問題と双対問題の一方を解けば両方を解いたことになる。どちらの問題を解けば有利かについては、基底のサイズが小さい方がよい。また、初期辞書が双対実行可能な場合には、双対実行可能性を保存したまま最適解を求めるといふ単体法も考えられるが、このような方法は双対単体法と呼ばれている。

### 6-1-5 線形計画問題の解の整数性

線形計画問題 (6・4) が最適解をもつとき、整数最適解 (ベクトルの全成分が整数) をもつための十分条件がいくつか知られている。例えば、単模性「 $A$  が整数行列ですべての基底  $B$  に対して  $\det A_B = \pm 1$ 」の下では、 $b$  が整数ベクトルならば基底解はすべて整数ベクトルとなるため、最適解が存在すれば整数最適解が存在する。このような場合、整数計画問題において整数条件を外した線形緩和問題が整数最適解をもつため、元の整数計画問題が容易に解ける。

## 12 群 - 2 編 - 6 章

## 6-2 整数計画

(執筆者: 田村明久)[2009 年 3 月 受領]

## 6-2-1 整数計画問題とその例

線形計画問題に全部あるいは一部の 변수が整数という制約を追加した問題を整数計画問題 (integer programming problem) という. 特に, 全部の 변수が整数という制約を付加したものを全整数計画問題 (all integer programming problem) と呼び, 一部の 변수に整数制約を付加したものを混合整数計画問題 (mixed integer programming problem) と呼ぶ. 同様に, 全部の 변수が 0 か 1 という制約を課したものを 0-1 全整数計画問題と呼び, 一部の 변수が 0 か 1 という制約を課したものを 0-1 混合整数計画問題と呼ぶ. 整数計画問題は, NP 困難と呼ばれるクラスに含まれ, 難しい問題の代表例である. 商用ソフトウェアの高速化に伴い, 実務での利用も増加している. 以下に組合せ最適化問題を 0-1 整数計画問題として定式化する例を示す.

## (1) 最大重み安定集合問題

グラフ  $G = (V, E)$  に対して,  $S \subseteq V$  のどの 2 頂点間にも辺が存在しないとき,  $S$  を安定集合あるいは独立集合という. 各頂点  $v \in V$  に対して重み  $w_v$  を導入し, 重みの総和が最大となる安定集合を求める問題を最大重み安定集合問題といい, 以下のように定式化できる.

$$\begin{aligned} \text{目的: } & \sum_{v \in V} w_v x_v \rightarrow \text{最大化} \\ \text{制約: } & x_u + x_v \leq 1 \quad \forall (u, v) \in E \\ & x_v \in \{0, 1\} \quad \forall v \in V \end{aligned} \quad (6 \cdot 13)$$

問題 (6・13) の許容解  $x_v$  ( $v \in V$ ) に対して, 各辺  $(u, v)$  で  $x_u = 0$  か  $x_v = 0$  となるので,  $x_v = 1$  となる頂点  $v$  全体は安定集合である. 逆に安定集合について, その要素  $v$  に対してのみ  $x_v = 1$  と定めれば, (6・13) の許容解を得る. これより, (6・13) は最大重み安定集合問題と一致する.

## (2) 最大重みマッチング問題

グラフ  $G = (V, E)$  に対して,  $M \subseteq E$  のどの 2 辺も端点を共有しないとき,  $M$  をマッチングという. 各辺  $e \in E$  に対して重み  $w_e$  を導入し, 重みの総和が最大となるマッチングを求める問題を最大重みマッチング問題という. 各頂点  $v$  に対して,  $v$  に接続する辺全体からなる集合を  $\delta(v)$  と記述すると, 最大重みマッチング問題は以下のように定式化できる.

$$\begin{aligned} \text{目的: } & \sum_{e \in E} w_e x_e \rightarrow \text{最大化} \\ \text{制約: } & \sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V \\ & x_e \in \{0, 1\} \quad \forall e \in E \end{aligned} \quad (6 \cdot 14)$$

問題 (6・14) の許容解  $x_e$  ( $e \in E$ ) に対して,  $x_e = 1$  となる辺全体はマッチングとなり, 逆も成り立つため, この問題は最大重みマッチング問題と等価である. 最大重み安定集合問題と最大重みマッチング問題は似ているが, その難しさにおいて大きく異なる. 前者は NP 困難となるが, 後者に対してはエドモンズ (Edmonds) が多項式時間解法を提案している (4 章 4-6-3 項を参照).

## 6-2-2 整数計画問題に対する解法

整数計画問題は難しい問題であるため、本章の概要で示したように厳密解法や発見的解法の構築といういくつかのアプローチがある。以下では、厳密解法について説明する。

### (1) 分枝限定法 (branch and bound method)

整数計画問題に限らず分枝限定法は厳密解を求める基本的な方法で、分枝操作と限定操作よりなる。分枝操作は与えられた問題を場合分けすることでいくつかの子問題に分割する操作である。分枝操作を子問題にも適用し、小問題を頂点とする系統木を用いて分枝の様子を表現する。例えば、0-1 整数計画問題ではある変数の値を 0 と 1 に固定した二つの子問題を生成することで分枝操作を実現できる。限定操作は解く必要のない小問題を省く操作で、(a) 小問題の緩和問題が実行不可能；(b) 小問題の緩和問題の最適値が実行途中で見つけた最良の許容解（暫定解という）の目的関数値よりも悪い；(c) 小問題の緩和問題の最適解が小問題の許容解であるのいずれかが成立すれば、小問題自身を解く必要がなくなる。分枝限定法においては、「前処理を用いていかに問題を小さくするか」、「いかによい暫定解を得るか」、「いかによい緩和問題を構成するか」、「分枝操作をどのように行うか」、「どの小問題を優先的に扱うか」などが重要であり、多くの工夫がなされている。特に緩和問題の選定においては、高速に解けることと元問題の最適値に近い最適値をもつという相反する要件を満たす必要がある。整数計画問題に対しては、整数制約を外すことで得られる線形緩和 (linear relaxation) と次で述べる切除平面法が有効である。一般的な緩和法としては、(a) ラグランジュ緩和 (Lagrangian relaxation)：一部の制約式を目的関数に繰り込み、制約を破ることをペナルティ化し、制約を除くことで問題の簡易化を図る方法；(b) 半正定値緩和：半正定値計画問題を用いた緩和法（本章 4 節参照）などがある。

### (2) 切除平面法 (cutting plane method)

元の整数計画問題  $[P]$  とその線形緩和問題  $[P_{lin}]$  を考える。 $[P]$  のどの許容解も満たす不等式  $\sum a_i x_i \leq b$  を妥当不等式 (valid inequality) という。 $[P_{lin}]$  の最適解が満たさないような妥当不等式を  $[P_{lin}]$  に制約として加えることで最適値の改善が望まれる。このように線形緩和問題の最適解を切除するように妥当不等式を繰り返し加えていく方法を切除平面法という。

$[P]$  のすべての許容解が  $\sum a_i x_i = b$  という 1 次等式を満たす場合には、 $\sum (a_i - \lfloor a_i \rfloor) x_i \geq b - \lfloor b \rfloor$  が  $[P]$  の妥当不等式となる。ただし  $\lfloor b \rfloor$  は実数  $b$  を超えない最大の整数とする。この妥当不等式をゴモリーカット (Gomory cut) という。ゴモリー (Gomory) は、線形計画問題に対する単体法にこのゴモリーカットの構成法を組み込んだ切除平面法 (ゴモリーの切除平面法という) を提案し、追加するカットの選択を工夫することで有限個のカットの追加で整数計画問題を解くことができることを示した (文献 11) 参照)。

### (3) 分枝切除法 (branch and cut method)

分枝限定法と切除平面法を組み合わせた解法である。分枝限定法の各子問題に対して、分枝する操作と切除平面を追加する操作を併用する。線形計画問題に対するソフトウェアの高速化に伴い、有力となった解法である。

## 12 群 - 2 編 - 6 章

## 6-3 動的計画

(執筆: 田村明久) [2009 年 3 月 受領]

## 6-3-1 逐次決定過程と動的計画問題

各離散時刻  $t$  ( $t = 0, 1, \dots, T$ ) における状態の集合  $S_t$ ,  $0 \leq t \leq T-1$  について状態  $u_t \in S_t$  から状態  $u_{t+1} \in S_{t+1}$  へ遷移したときの費用  $c_t(u_t, u_{t+1})$  と最終時刻  $t = T$  での状態  $u_T \in S_T$  に対する費用  $c_T(u_T)$  が与えられたもとで、各時刻での状態遷移をどのようにするかを決定する過程を逐次決定過程 (sequential decision process) という。動的計画問題 (dynamic programming problem) は、逐次決定過程において過程の総費用を最小化する問題である。ここで、過程を状態遷移の列  $(u'_0, u'_1, \dots, u'_T)$  とみなし、その総費用を  $\sum_{t=0}^{T-1} c_t(u'_t, u'_{t+1}) + c_T(u'_T)$  と定める。

## 6-3-2 最適性の原理

動的計画問題に対しては、最適性の原理「動的計画問題の最適解については、時刻  $t$  までの決定過程で遷移した状態に関して、以後の決定過程は最適でなければならない」を利用して再帰方程式を構築し、これを解くことにより最適解を求める。上記の場合では、時刻  $t$  での状態が  $u \in S_t$  の場合の  $t$  以降の過程の最小費用を  $f_t(u)$  と表すと、最適性の原理より

$$\begin{aligned} f_T(u) &= c_T(u) & (\forall u \in S_T) \\ f_t(u) &= \min_{v \in S_{t+1}} \{c_t(u, v) + f_{t+1}(v)\} & (\forall u \in S_t; t = 0, 1, \dots, T-1) \end{aligned} \quad (6.15)$$

という再帰方程式が導け、最適値は  $\min_{u \in S_0} f_0(u)$  となる。動的計画問題の利用においては、どのように逐次決定過程を構築するかが肝要となる。

## (1) 例: 0-1 ナップザック問題

与えられた正整数  $b, c_j, a_j$  ( $j = 1, 2, \dots, n$ ) に対して

$$\begin{aligned} \text{目的: } & \sum_{j=1}^n c_j x_j \rightarrow \text{最大化} \\ \text{制約: } & \sum_{j=1}^n a_j x_j \leq b \\ & x_j \in \{0, 1\} \quad (j = 1, 2, \dots, n) \end{aligned} \quad (6.16)$$

と記述できる問題を 0-1 ナップザック問題という。これを動的計画問題として定式化してみる。 $t = 1, 2, \dots, n$  と  $u = 1, 2, \dots, b$  に対して、

$$f_t(u) = \max \left\{ \sum_{j=1}^t c_j x_j \mid \sum_{j=1}^t a_j x_j \leq u, x_j \in \{0, 1\} \right\} \quad (6.17)$$

と定義すると (6.16) の最適値は  $f_n(b)$  である。最適性の原理より以下の再帰方程式を得る。

$$\begin{aligned} f_1(u) &= \begin{cases} 0 & (u < a_1) \\ c_1 & (a_1 \leq u) \end{cases} \\ f_t(u) &= \max\{f_{t-1}(u), f_{t-1}(u - a_t) + c_t\} \quad (t = 2, 3, \dots, n) \end{aligned} \quad (6.18)$$

再帰方程式 (6.18) を  $t, u$  が小さい値から順次計算していくことで最適値  $f_n(b)$  が求まる。

## 12 群 - 2 編 - 6 章

## 6-4 錐線形計画

(執筆: 田村明久)[2009 年 3 月 受領]

## 6-4-1 錐線形計画問題

錐線形計画問題 (cone linear programming problem) は、凸計画問題の一種で線形計画問題の拡張に当たり、閉凸錐 $\mathcal{K} \subset R^n$ ,  $c \in R^n$ ,  $a_i \in R^n$  ( $i = 1, 2, \dots, m$ ) 及び  $b \in R^m$  に対して、

$$\begin{aligned} \text{目的: } & c^\top x \rightarrow \text{最小化} \\ \text{制約: } & a_i^\top x = b_i \quad (i = 1, 2, \dots, m) \\ & x \in \mathcal{K} \end{aligned} \quad (6.19)$$

と記述できる問題である。  $c^\top x = \sum_{j=1}^n c_j x_j$  は内積の一例であるが、より一般的な内積を用いた記述法もある。

閉凸錐  $\mathcal{K} \subset R^n$  に対して、集合  $\mathcal{K}^* = \{v \in R^n \mid \forall u \in \mathcal{K}, u^\top v \geq 0\}$  は閉集合かつ凸錐となり、 $\mathcal{K}$  の双対錐と呼ばれる。一般に  $(\mathcal{K}^*)^* = \mathcal{K}$  が成立し、 $\mathcal{K}^* = \mathcal{K}$  を満たす閉凸錐を自己双対錐という。錐線形計画問題 (6.19) に対して、 $R^m$  上の最適化問題

$$\begin{aligned} \text{目的: } & b^\top y \rightarrow \text{最大化} \\ \text{制約: } & s + \sum_{i=1}^m y_i a_i = c \\ & s \in \mathcal{K}^* \end{aligned} \quad (6.20)$$

を (6.19) の双対問題と定義する。双対問題 (6.20) も (6.19) の形式に変換でき、双対問題の双対問題は主問題と一致し、以下の弱双対定理が成立する。

弱双対定理: (6.19) と (6.20) の許容解  $x$  と  $(y, s)$  に対し、 $c^\top x \geq b^\top y$  が成り立つ。

強双対定理に関しては、線形計画問題のときに比べてより強い条件を必要とする。

定理: 主問題 (6.19) に内点許容解<sup>†</sup>が存在し、双対問題 (6.20) に許容解が存在するならば、最適値は一致し、双対問題には最適解が存在する。

定理: 主問題 (6.19) に許容解が存在し、双対問題 (6.20) に内点許容解<sup>‡</sup>が存在するならば、最適値は一致し、主問題には最適解が存在する。

定理: 主問題 (6.19) と双対問題 (6.20) に内点許容解が存在するならば、最適値は一致し、主問題と双対問題に最適解が存在する。

## (1) 例その 1: 線形計画問題

$\mathcal{K} = \{x \in R^n \mid x \geq 0\}$  とすると (6.19) は線形計画問題となり、更に  $\mathcal{K}$  は自己双対錐となる

\*  $\mathcal{K}$  が凸錐であるとは、任意の  $x, y \in \mathcal{K}$  と任意の実数  $\lambda, \mu \geq 0$  に対して  $\lambda x + \mu y \in \mathcal{K}$  を満たすことと定義し、特に閉集合である凸錐を閉凸錐という。

†  $\mathcal{K}$  の内点 (その点を中心とする十分小さな開球が  $\mathcal{K}$  に含まれる) である許容解のこと。

‡  $s$  が  $\mathcal{K}^*$  の内点である許容解  $(y, s)$  のこと。

ため (6.20) は線形計画問題における双対問題と一致する。

## (2) 例その 2 : 半正定値計画問題 (semidefinite programming problem)

$n \times n$  実対称行列全体の集合を  $S_n$  とし,  $n \times n$  半正定値行列<sup>§</sup>全体を  $\mathcal{P}_n$  と表記する.  $S_n$  の任意の要素  $X, Y$  に対し, これらの内積を  $X \bullet Y = \sum_{i=1}^n \sum_{j=1}^n X_{ij}Y_{ij}$  と定義すると,  $\mathcal{P}_n$  は  $n \times n$  実行列全体  $R^{n \times n}$  の自己双対錐となる. ただし  $X_{ij}$  と  $Y_{ij}$  はそれぞれ行列  $X$  と  $Y$  の  $(i, j)$  成分である. 入力  $A_i \in S_n$  ( $i = 1, 2, \dots, m$ ),  $b \in R^m$ ,  $C \in S_n$  に対して

$$\begin{aligned} \text{目的: } & C \bullet X \rightarrow \text{最小化} \\ \text{制約: } & A_i \bullet X = b_i \quad (i = 1, 2, \dots, m) \\ & X \in \mathcal{P}_n \end{aligned} \quad (6.21)$$

と記述できる特殊な錐線形計画問題を半正定値計画問題という. 主問題 (6.21) の双対問題は

$$\begin{aligned} \text{目的: } & b^\top y \rightarrow \text{最大化} \\ \text{制約: } & S + \sum_{i=1}^m y_i A_i = C \\ & S \in \mathcal{P}_n \end{aligned} \quad (6.22)$$

と定義され, 6-4-1 項の弱双対定理や強双対定理が成立する. 半正定値計画問題に対しては, 主双対内点法など効率的な解法が提案されている<sup>10, 12)</sup>.

### 6-4-2 半正定値緩和の応用例

#### (1) 最大重み安定集合問題に対する半正定値緩和

最大重み安定集合問題 (6.13) は,

$$\begin{aligned} \text{目的: } & \sum_{v \in V} w_v x_v \rightarrow \text{最大化} \\ \text{制約: } & x_u x_v = 0 \quad \forall (u, v) \in E \\ & x_v(1 - x_v) = 0 \quad \forall v \in V \end{aligned} \quad (6.23)$$

とも定式化できる. ここで  $X_{uv} = x_u x_v$  という新たな変数と  $X_{uv}$  を成分とする行列  $X$  を導入し,

$$\begin{aligned} \text{目的: } & \sum_{v \in V} w_v x_v \rightarrow \text{最大化} \\ \text{制約: } & X_{uv} = 0 \quad \forall (u, v) \in E \\ & X_{vv} - x_v = 0 \quad \forall v \in V \\ & X - xx^\top = O \end{aligned} \quad (6.24)$$

と書き直すことができる. ここでグラフの頂点数を  $n$  とし,  $O$  は  $n \times n$  ゼロ行列とする. (6.24)

の最後の制約を  $X - xx^\top \in \mathcal{P}_n$  と緩和すると,  $X - xx^\top \in \mathcal{P}_n \Leftrightarrow Y = \begin{pmatrix} 1 & x^\top \\ x & X \end{pmatrix} \in \mathcal{P}_{1+n}$  より, 半正定値緩和 ( $Y$  の第 1 行と第 1 列の添字は 0 とする)

$$\begin{aligned} \text{目的: } & \sum_{v \in V} w_v Y_{v0} \rightarrow \text{最大化} \\ \text{制約: } & Y_{uv} = 0 \quad \forall (u, v) \in E \\ & Y_{vv} - Y_{v0} = 0 \quad \forall v \in V \\ & Y_{00} = 1, Y \in \mathcal{P}_{1+n} \end{aligned} \quad (6.25)$$

<sup>§</sup> 任意の  $x \in R^n$  に対して  $x^\top A x \geq 0$  を満たす実対称行列  $A$  のこと.

を得る．任意の重み  $w$  に対して元問題 (6.13) と半正定値緩和 (6.25) の最適値が一致するためのグラフ的な必要十分条件も知られている<sup>8)</sup>．

## (2) センサ位置同定問題に対する半正定値緩和

平面  $R^2$  において,  $n$  個の位置不明センサの位置  $x_1, x_2, \dots, x_n \in R^2$  を同定する問題を考える．ただし, アンカーセンサと呼ばれる位置  $a_1, a_2, \dots, a_m \in R^2$  が既知のセンサに対して, 位置不明センサどうして距離が既知であるものの対の集合  $P_1$  と, 位置不明センサとアンカーセンサで距離が既知であるものの対の集合  $P_2$  がデータとして与えられているとする．すなわち, 制約条件は

$$\begin{aligned} (x_i - x_j)^\top (x_i - x_j) &= d_{ij}^2 \quad \forall (i, j) \in P_1 \\ (x_i - a_k)^\top (x_i - a_k) &= d_{ik}^2 \quad \forall (i, k) \in P_2 \end{aligned} \quad (6.26)$$

と書ける．ただし  $d_{ij}, d_{ik}$  は既知の距離を意味する． $2 \times n$  の変数行列  $X = (x_1, x_2, \dots, x_n)$  を導入することで, このセンサ位置同定問題は (ただし  $I_2$  は  $2 \times 2$  単位行列)

$$\begin{aligned} \text{目的: } Z &= \begin{pmatrix} I_2 & X \\ X^\top & Y \end{pmatrix} \text{ を求める} \\ \text{制約: } & (I_2, X)^\top (I_2, X) = Z \quad (6.27) \\ & Y_{ii} - Y_{ij} - Y_{ji} + Y_{jj} = d_{ij}^2 \quad \forall (i, j) \in P_1 \\ & Y_{ii} - x_i^\top a_k - a_k^\top x_i + a_k^\top a_k = d_{ik}^2 \quad \forall (i, k) \in P_2 \end{aligned}$$

と定式化できる．このセンサ位置同定問題は難しく, 厳密解の計算にはよい緩和問題が必要となる． $Z$  は階数 2 の半正定値行列になるが, 階数が 2 であることを緩め, すなわち, 制約の第 1 式を  $Z = \begin{pmatrix} I_2 & X \\ X^\top & Y \end{pmatrix} \in \mathcal{P}_{2+n}$  と置き換え, 目的を  $O \bullet Z$  の最小化に置き換えることで半正定値緩和を得ることができる．

### 参考文献

- 1) G.L. Nemhauser, A.H.G. Rinnooy Kan and M.J. Todd (Ed.), "Optimization," North-Holland, 1989; 伊理正夫, 今野 浩, 刀根 薫 (監訳), "最適化ハンドブック," 朝倉書店, 1995.
- 2) 久保幹雄, 田村明久, 松井知己 (編), "応用数理計画ハンドブック," 朝倉書店, 2002.
- 3) V.V. Vazirani, "Approximation Algorithms," Springer-Verlag, 2001; 浅野孝夫 (訳), "近似アルゴリズム," シュプリンガー・フェアラーク東京, 2002.
- 4) 柳浦睦憲, 茨木俊秀, "組合せ最適化-メタ戦略を中心として-, "朝倉書店, 2001.
- 5) R. Motwani and P. Raghavan, "Randomized Algorithms," Cambridge University Press, 1995.
- 6) G.B. Dantzig, "Linear Programming and Extensions," Princeton University Press, 1963.
- 7) L.G. Khachiyan, "A polynomial algorithm in linear programming," (in Russian), Dokl.Akad.Nauk SSSR, vol.244, pp.1093-1096, 1979.
- 8) M. Grötschel, L. Lovász and A. Schrijver, "Geometric Algorithms and Combinatorial Optimization, (2nd Ed.)," Springer-Verlag, 1993.
- 9) N. Karmarkar, "A new polynomial-time algorithm for linear programming," Combinatorica, vol.4, pp. 373-395, 1984.
- 10) 小島政和, 土谷 隆, 水野眞治, 矢部 博, "内点法," 朝倉書店, 2001.
- 11) 今野 浩, 鈴木久敏, "整数計画法と組合せ最適化," 日科技連出版社, 1982.
- 12) 田村明久, 村松正和, "最適化法," 朝倉書店, 2002.