

相補的プログラミング†

茨 木 俊 秀*

1. はじめに

相補的プログラミング問題 (Complementary Programming Problem, 略して, CP 問題) とは, 次の条件を満足するベクトル (x, u, v) を見いだす問題 P である [8][9].

$$(1) \quad P: \text{目標関数 } z = d^T x + e^T u + f^T v \longrightarrow \text{最小}$$

$$\text{拘束条件 } Ax + Bu + Cv \geq g$$

$$u^T v = 0$$

$$x, u, v \geq 0$$

ただし¹⁾, $x \in R^n$, $u, v \in R^m$ は変数ベクトル, $d \in R^n$, $e, f \in R^m$, $g \in R^p$, また, A, B, C は, それぞれ実数を要素とする $p \times n$, $p \times m$, $p \times m$ 行列である. なお, 本論文では, ベクトルはすべて列ベクトルとし, 行ベクトルは転置記号 T を付して示す.

CP 問題 P が, 通常の LP 問題と異なっている点は, 相補条件

$$(2) \quad u^T v = 0$$

にある. これは, $u, v \geq 0$ を考慮すると, u, v の各要素対 u_j, v_j の少なくとも一方が 0 であることに等しい. 以後, この変数 u, v を相補変数 (Complementary Variables) と呼ぶ. CP 問題の許容領域 (Feasible Region) は相補条件のために, 一般に凸領域ではなく, LP (線形計画法) とは本質的に異なる手法が必要とされる.

(2) のような相補的条件は, 実際の応用に際して, 数多く見受けられる. たとえば, 交差点 j における交通量を考えると, 縦方向の交通量 u_j が正のとき, 横方向の交通量 v_j は 0 でなければならない. また, その逆も成立する. つまり, u, v に関し, (2) 式が成立しなければならない. このような具体例を他にも見いだすのは困難ではないであろう. 2章では, より一般的な応用例をいくつか挙げる.

CP 問題(1)は, 筆者の知るかぎり, これまで考慮された事実はないようであるが, LP 問題, 2次計画問題, Bi-Matrix Game 等との関連において, (1) の特殊な場合である

$$(3) \quad u - Mv = g$$

$$u^T v = 0$$

† 1970年11月12日受理, 1971年9月7日再受理.

1) R^k は k 次元実数空間.

* 京都大学工学部・数理工学.

という系の解の存在に関しては、しばしば議論された [1] [13]. ただし, $u, v \in R^m$ は変数ベクトル, M は $m \times m$ 行列, $g \in R^m$ である. 容易にわかるように, (1) は (3) に比較し, 目標関数を持つこと, および, 拘束条件の形式において, より一般的であって, それだけ広い範囲の問題を含むことができる.

2. CP 問題の応用

以下に, CP 問題に帰着される典型的な問題をあげる. その他にも, たとえば, 2次計画問題が適当な仮定の下に, CP 問題の形式に帰着されることは周知である.

(i) 絶対値計画問題

次のような, 絶対値を含む計画問題を考えよう.

$$(4) \quad \text{目標関数 } d^T x + e^T |x| \longrightarrow \text{最小}$$

$$\text{拘束条件 } Ax + B|x| \geq g$$

ただし, $x, |x| = (|x_1|, |x_2|, \dots, |x_n|)^T \in R^n$ は変数ベクトル, $d, e \in R^n, g \in R^p, A, B$ は $p \times n$ 行列である. (4) において, $u_j, v_j \geq 0$ を用いて

$$x_j = u_j - v_j, \quad j=1, 2, \dots, n$$

と置けば, その絶対値 $|x_j|$ は

$$|x_j| = u_j + v_j$$

$$u_j v_j = 0, \quad j=1, 2, \dots, n$$

と書くことができ, 次の CP 問題に帰着される.

$$(5) \quad \text{目標関数 } d^T(u-v) + e^T(u+v) \longrightarrow \text{最小}$$

$$\text{拘束条件 } A(u-v) + B(u+v) \geq g$$

$$u^T v = 0$$

$$u, v \geq 0$$

このような絶対値を含む問題は, 実際の応用に際して, しばしば遭遇するものである. たとえば, ネットワーク・フロー, とくに, 多種フローの問題においては, 各フローの絶対値の和 (つまり, 方向を無視したフローの和) が, 各枝の容量を越えてはならないという条件を扱う.

(ii) 0-1 混合整数計画問題

0-1 混合整数計画問題は, 次の形式に書かれる.

$$(6) \quad \text{目標関数 } d^T x + e^T u \longrightarrow \text{最小}$$

$$\text{拘束条件 } Ax + Bu \geq g$$

$$x \geq 0$$

$$u_j = 0 \text{ or } 1, \quad j=1, 2, \dots, m$$

ただし, $x \in R^n, u \in R^m$ は変数ベクトル, $d \in R^n, e \in R^m, g \in R^p, A, B$ は $p \times n, p \times m$ 行列である.

0-1 変数 u_j を CP 問題に帰着させるために, スラック変数 v_j を導入し, $u_j = 0 \text{ or } 1$ とい

う条件を

$$u_j + v_j = 1$$

$$u_j v_j = 0, \quad j = 1, 2, \dots, m$$

と書く。(6)は次の CP 問題と等価である.

$$(7) \quad \text{目標関数} \quad d^T x + e^T u \longrightarrow \text{最小}$$

$$\text{拘束条件} \quad Ax + Bu \geq g$$

$$x, u, v \geq 0$$

$$u^T v = 0$$

$$u + v = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

(iii) OR-条件を持つ LP 問題

次の OR-条件を持つ LP 問題を考える.

$$(8) \quad \text{目標関数} \quad d^T x \longrightarrow \text{最小}$$

$$\text{拘束条件} \quad A_i x \geq g_i \text{ or } B_i x \geq h_i, \quad i = 1, 2, \dots, k$$

$$x \geq 0$$

ただし, $x \in \mathbb{R}^n$ は変数ベクトル, $g_i \in \mathbb{R}^{p_i}$, A_i は $p_i \times n$ 行列, $h_i \in \mathbb{R}^{q_i}$, B_i は $q_i \times n$ 行列, $i = 1, 2, \dots, k$, である. つまり, (8)は各 i に対して, $A_i x \geq g_i$ あるいは, $B_i x \geq h_i$ の少なくとも一方が満足されればよい, という条件を持つ. これは, 人為変数 $u_i, v_i, i = 1, 2, \dots, k$ を導入することによって, 次の CP 問題に書ける.

$$(9) \quad \text{目標関数} \quad d^T x \longrightarrow \text{最小}$$

$$\text{拘束条件} \quad A_i x + [u_i] \geq g_i$$

$$B_i x + [v_i] \geq h_i$$

$$u_i v_i = 0, \quad i = 1, 2, \dots, k$$

$$x \geq 0, \quad u, v \geq 0$$

ただし,

$$[u_i] = \begin{pmatrix} u_i \\ u_i \\ \vdots \\ u_i \end{pmatrix} \in \mathbb{R}^{p_i}$$

$$[v_i] = \begin{pmatrix} v_i \\ v_i \\ \vdots \\ v_i \end{pmatrix} \in \mathbb{R}^{q_i}, \quad i = 1, 2, \dots, k$$

3. CP 問題のアルゴリズム：分枝限定法

CP 問題(1)を解くために、分枝限定法 (Branch and Bound Method) に基づくアルゴリズムを述べる [8]. CP 問題のアルゴリズムとしては、他に、切除平面法 (Cutting Plane Method) がある [9]. しかし、筆者の経験から見るかぎり、分枝限定法が、計算能率の点から、よりすぐれていると思われるので、ここでは、切除平面法を、その特長を生かして、分枝限定法の補助手段に用いるという立場をとり、そのために必要な性質のみ説明する。

分枝限定法を構成するために、最初に、2, 3 の定義を導入する。(1) 式の P に対して、任意の $(x, u, v) \in R^{n+2m}$ を解という。解で、(1) の拘束条件: $Ax + Bu + Cv \geq g$, $x, u, v \geq 0$, $u^T v = 0$, を満足するものを許容解という。また、許容解のうちで、目標関数の値 z を最小にするものを最適解といい、そのときの z の値、すなわち、 P の値を z^0 と記す。 P が不能 (許容解を持たない) ならば、 $z^0 = \infty$ と考える。

部分解 S によって、 P の変数 u, v のうちで 0 に固定されているものの集合を示す。ただし、 u_j, v_j の両者を同時に 0 に固定することはない。ここで便宜上、 $u_j = 0$ を j , $v_j = 0$ を $-j$ によって示す。たとえば、 $S = \{-4, 1, -2\}$ は $v_4 = 0, u_1 = 0, v_2 = 0$ の意味である。 S において、 u_j, v_j のどちらも 0 に固定されていなければ、 u_j と v_j は自由変数と呼ばれる。 S の自由変数の添字 j の集合を F_S と記す。 $T \supset S$ を満足する部分解において、 $F_T = \phi$ (空集合) ならば、 T は S の Completion であるという。

さて、部分解 S に対して、部分問題 $P(S)$ を次のように定義する。

(10) $P(S)$: 目標関数 $z = d^T x + e^T u + f^T v \longrightarrow$ 最小

$$\text{拘束条件 } Ax + Bu + Cv \geq g$$

$$x, u, v \geq 0$$

$$u^T v = 0$$

$$u_j = 0 \text{ if } j \in S$$

$$v_j = 0 \text{ if } -j \in S$$

$P(S)$ の許容解、最適解等も、 P と同様に定義する。また、 $P(S)$ の最適解の値を $z^0(S)$ と記す。 $P(S)$ が不能ならば、 $z^0(S) = \infty$ と考える。この記法を用いると、 $P = P(\phi)$ である。

つぎに、 $\bar{P}, \bar{P}(S)$ を、それぞれ、 $P, P(S)$ から相補条件 $u^T v = 0$ を除いて得られる LP 問題とする。 $\bar{P}, \bar{P}(S)$ の許容解、最適解等は、やはり、 $P, P(S)$ に対するものから、条件 $u^T v = 0$ を除いて定義される。また、 $\bar{P}, \bar{P}(S)$ の最適解の値をそれぞれ $\bar{z}, \bar{z}(S)$ と記す。

分枝限定法は、よく知られているように、種々の計画、設計問題に適用され、成功している例も多く報告されている [12][15]. その基本的原理は、まず、与えられた問題 P に適当な細分を加えて、等価的に、いくつかの部分問題に変換することから始まる。この、細分を繰り返し適用すれば、 P は、数多くのやさしい小規模な部分問題に帰着されるであろう。この得られた部分問題全体のごく一部分だけを解いて、 P の最適解を得ようとするのが、分枝限定法の主眼点である。

そのためには、与えられた問題の固有の性質に応じて、種々の工夫を必要とするのが常である。

CP 問題においては、まず、ある変数対 u_j, v_j のそれぞれを 0 に固定することによって、 P を部分問題 $P(j)$ と $P(-j)$ に細分する。 $P(j), P(-j)$ の許容解は P の許容解でもあるから、 $P(j)$ と $P(-j)$ それぞれの最適解のうち、小さな値を持つほうが、 P の最適解であることは自明である。この議論は、生成された部分問題にも繰り返し適用でき、一般に $P(S)$ を $P(S, j)$ と $P(S, -j)$ にする細分を続けると、 P を多くの小さな部分問題に帰着できる。この様子は、いわゆる探索図を用いると理解しやすい。図 1 はその 1 例である。もちろん、得られた部分問題のすべてを解けば、 P の最適解が得られるが、できるだけ不要な部分問題を除くために、以下の方針をとる。

計算の途中、部分問題 $P(S)$ の最適解として、 $P(\phi)$ の許容解が得られることがあるが、その時点までに得られている最良の（最小の値を持つ） P の許容解を暫定解として記憶しておく。暫定解の値を z^* によって記す。

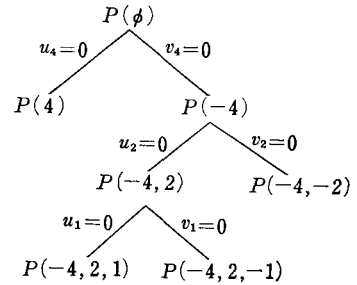


図 1 探索図の例

探索図において、ある部分問題 $P(S)$ が、次のいずれかを満足すれば、 $P(S)$ は終端されたという。終端された部分問題を、さらに調べる必要はない。

- (i) $P(S)$ の最適解が得られている。
- (ii) $P(S)$ は不能（許容解を持たない）であるか、 S のすべての Completion がその時点の暫定解の値 z^* より小さな値を与え得ないことが証明できる。

終端されてもいず、また、より小さな部分問題に細分されてもいない（つまり、探索図の末端の節点に対応する）部分問題は、活性であると呼ばれる。活性な問題が存在しなくなったとき、計算は終了し、そのときの暫定解が P の最適解を与えるのである。

以上の準備の下に、CP 問題の分枝限定法に基づくアルゴリズムは次のように書かれる。

手順 1 : $P(\phi)(=P)$ を活性な部分問題として持つ探索図から出発する。 $z^* = \infty$ 。

手順 2 (活性部分問題の選択) : 活性な部分問題がなければ、計算終了。暫定解が P の最適解を与える。このとき、 $z^* = \infty$ ならば、 P は不能である。活性な部分問題が存在すれば、そのうち 1 個を選んで $P(S)$ とし、手順 3 へ進む。

手順 3 (部分問題のテスト) :

- (i) $P(S)$ の最適解が得られたなら、 $P(S)$ を終端し、手順 4 へ進む。
- (ii) $P(S)$ が不能であるか、 S のすべての Completion が z^* より小さな値を与え得ないことがわかれば、 $P(S)$ を終端し、手順 2 へ戻る。
- (iii) 暫定解より小さな値を持つ P の許容解を S の Completion として得るには、 $u_j=0$ でなければならぬことがわかる。このときには、 $P(S, j)$ と $P(S, -j)$ を生成し、 $P(S, -j)$ を終端した後、手順 2 へ戻る。 $v_j=0$ に関しても同様。
- (iv) 上の (i) (ii) (iii) のどれも満足しないなら、手順 5 へ進む。

手順 4: 得られた $P(S)$ の最適解と、暫定解を比較し、小さな値を持つほうを改めて暫定解として、手順 2 へ戻る。

手順 5 (分枝変数の選択): 自由変数の中から、適当な変数対 u_j, v_j (分枝変数という) を選び、 $P(S)$ を $P(S, j)$, $P(S, -j)$ に細分し、手順 2 へ戻る. ($P(S, j)$, $P(S, -j)$ は活性)

部分問題の個数は、たかだか有限であり、同じ部分問題を繰り返し生成することはないから、各部分問題 $P(S)$ のテスト (手順 3) を有限回の操作で終わるようにしておけば、全手順も有限回の操作で終了する. しかし、実用上重要であるのは、その収束速度であって、これは活性問題の選定法、部分問題のテスト、分枝変数の決定法等を具体的にどう定めるかに依存して、大いに異なる. 以後、4, 5, 6 節において、各手順の細部を定めるために必要な性質を導出し、その結果を用いて、7 節以下で具体的な手順を構成する予定である.

4. CP 問題 P と LP 問題 \bar{P} の関係

本節では、5 節において切除平面法を導出する際必要となる CP 問題 P とその LP 問題 \bar{P} の関係を議論する.

定理 1: \bar{P} の基底解で P の最適解となるものが存在する.

証明: 2^m 個の自由変数を持たない部分解

$$S_1 = (-1, -2, \dots, -m)$$

$$S_2 = (-1, -2, \dots, -(m-1), m)$$

$$S_3 = (-1, -2, \dots, -(m-2), m-1, -m)$$

⋮

$$S_{2^m} = (1, 2, \dots, m)$$

を考える. P の値 z^0 は

$$z^0 = \min \{z^0(S_k) \mid k=1, 2, \dots, 2^m\}$$

によって与えられる. よって、各 $P(S_k)$ に対して、 \bar{P} の基底解となる最適解が存在することを示せば定理が証明される. さて、 $P(S_k)$ は自由変数を持たないから、 $u^T v = 0$ は自動的に満足している. 換言すれば、 $P(S_k)$ は 0 に固定されていない $n+m$ 個の変数、および、 $p+m+n$ 個の拘束条件 ($x, u, v \geq 0$ も含めているが、0 に固定されている変数は除外している) を持つ LP 問題 $\bar{P}(S_k)$ である. したがって、LP 問題の性質から、 $P(S_k)$ の基底解で最適解となるものが存在することがわかる. それを \bar{y} ($\bar{y} \in R^{n+m}$) と記すと、 \bar{y} は $P(S_k)$ の拘束条件の 1 次独立な $n+m$ 個を等式として満たす. つぎに、 \bar{y} に、 S_k において 0 に固定されている m 個の変数を追加して得られる $\bar{y}' \in R^{n+2m}$ を考えよう. \bar{y}' は、当然、上の $n+m$ 個の拘束条件を等式として満たし、また、固定されている m 個の変数に対する拘束条件 ($u_j \geq 0$ あるいは $v_j \geq 0$ 等) をも等式として満たす. 両者を合わせた、 $n+2m$ 個の拘束条件が 1 次独立²⁾ であることは自明であるから、 \bar{y}'

2) $a^i y \leq b_i$, $i=1, 2, \dots, N$, の係数ベクトル a^i が互いに 1 次独立ならば、この不等式系は 1 次独立であるという.

は \bar{P} の基底解である.

(証明終)

つぎに, CP 問題 P ((1)式)において, d, e, f, g, A, B, C の要素を, すべて整数と仮定しても, 実用上一般性を失うことはないことに注意しよう. なぜなら, 与えられた数を有理数によって任意の精度で近似でき, 有理数ならば, P の各式全体をある正整数倍することによって, 整数係数を持つ問題に変換できるからである.

定理 2: CP 問題 P の係数すべてが整数であると仮定する. このとき, P の係数のみに関係する $\Delta > 0$ が存在して, P の定理 1 を満たす最適解を

$$\begin{pmatrix} z \\ x \\ u \\ v \end{pmatrix} = \Delta \begin{pmatrix} z' \\ x' \\ u' \\ v' \end{pmatrix}$$

z', x', u', v' : 整数ベクトル

と書くことができる.

証明: よく知られているように, \bar{P} の任意の基底解は, そのときの基底行列 D に対し

$$D_D = \frac{1}{|\det D|}$$

と置けば,

$$\begin{pmatrix} z \\ x \\ u \\ v \end{pmatrix} = D_D \begin{pmatrix} z'' \\ x'' \\ u'' \\ v'' \end{pmatrix}$$

z'', x'', u'', v'' : 整数ベクトル

と書くことができる. したがって, 可能な基底行列すべてに対して, $|\det D|$ を求め (これは整数値), その最小公倍数を M とすれば, 定理 1 を考慮して,

$$\Delta = \frac{1}{M}$$

が定理の条件を満足する.

(証明終)

定理 2 は, P の最適解を得るには, Δ の整数倍という離散的な値のみを考えれば充分であることを示している. この性質は, 次節の切除平面法を導出するために用いられる.

5. 切除平面法

本節では, 切除平面法に基づく, CP 問題のアルゴリズムを述べる. これは, \bar{P} に, カットと呼ばれる拘束条件を次々と追加することによって, \bar{P} の許容領域を次第に狭くし, その結果得られる LP 問題の最適解として, P の最適解を求める手法である. \bar{P} に k 個のカットを追加して得られる問題を \bar{P}_k とする時, \bar{P}_k のカットとは, \bar{P}_k の最適解 (P の許容解ではないとする) は切除するが, P の最適解を含むある部分集合は決して切除しないという性質を持つ拘束条件である. 以下で, そのような性質を満足する 2 種のカットを導入する. 実は, そのカットを有限回追

加することによって、必ず P の最適解が得られることを証明できるが、ここでは、切除平面法を、もっぱら、分枝限定法に補助的に用いるという目的に沿って説明し、収束性に関する議論は省略する。

さて、(1)に対応する LP 問題

$$(11) \quad \bar{P} : \text{目標関数 } z = d^T x + e^T u + f^T v \longrightarrow \text{最小}$$

$$\text{拘束条件 } Ax + Bu + Cv \geq g$$

$$x, u, v \geq 0$$

を、(主あるいは双対) シンプレックス法で解こう。すなわち、まず、スラック変数 $w \in R^p$ を導入して、(11)を

$$(12) \quad z = d^T x + c^T u + f^T v$$

$$w = -g + Ax + Bu + Cv$$

$$x = x$$

$$u = u$$

$$v = v$$

$$w, x, u, v \geq 0$$

と書く。(12)は、変数 z, w, x, u, v を、非基底変数 x, u, v によって展開していると考えられることもできる。一般に、各変数を、1次独立な非基底変数(すなわち、(12)における係数列が1次独立) $t_1, t_2, \dots, t_N (N = n + 2m)$ によって展開するとき、

$$(13) \quad \begin{pmatrix} z \\ y \end{pmatrix} = \begin{pmatrix} \alpha_{00} \\ \alpha_{10} \\ \vdots \\ \alpha_{M0} \end{pmatrix} + \sum_{j=1}^N \begin{pmatrix} \alpha_{0j} \\ \alpha_{1j} \\ \vdots \\ \alpha_{Mj} \end{pmatrix} t_j$$

$$y \geq 0$$

$$t_j \geq 0, \quad j = 1, 2, \dots, N$$

と書かれる。ただし、 $M = n + 2m + p$ 、および $y^T = (w^T, x^T, u^T, v^T) \in R^M$ である。いうまでもなく、(13)の係数 α_{ij} が、LP の計算に用いられるシンプレックス・タブロの各要素に対応する。

シンプレックス法によって

$$(14) \quad (\alpha_{10}, \alpha_{20}, \dots, \alpha_{M0}) \geq 0 \quad (\text{主許容条件})$$

$$(15) \quad (\alpha_{01}, \alpha_{02}, \dots, \alpha_{0N}) \geq 0 \quad (\text{双対許容条件})$$

を満足するタブロが得られたとき

$$(16) \quad \begin{pmatrix} z \\ y \end{pmatrix} = \begin{pmatrix} \alpha_{00} \\ \vdots \\ \alpha_{M0} \end{pmatrix}$$

は \bar{P} の値およびその最適解を与える。

つぎに、 P^* を P の許容領域を含む許容領域を持つ LP 問題(たとえば \bar{P}) としよう。 P^* の最適解が、やはり、(16)によって与えられるとき、これが、条件 $u^T v = 0$ を満足すれば、(16)は

P の最適解でもある (P^* は P より緩い拘束条件を持つ問題だから). しかし, ある j に対して, $u_j v_j > 0$ ならば, (16) は P の許容解ではない. このとき, $u_j = y_p$, $v_j = y_q$ とすれば,

$$(17) \quad \alpha_{p0} \alpha_{q0} > 0$$

を満足する. (17) の条件の下で, タブロ (13) の第 p 行, 第 q 行から得られる次の拘束条件を考えよう.

$$(18) \quad s = -1 + \sum_{j=1}^N \beta_j t_j$$

$$s \geq 0$$

ただし,

$$(19) \quad \beta_j = \begin{cases} 0, & \text{if } \alpha_{lj} \geq 0, \quad l=p, q \\ \min \left\{ -\frac{\alpha_{lj}}{\alpha_{l0}} \mid -\frac{\alpha_{lj}}{\alpha_{l0}} > 0, \quad l=p, q \right\}, & \text{otherwise} \end{cases}$$

(18) 式を P^* の C-カット (Complementarity Cut) と呼ぶ.

C-カットの幾何学的意味は次のように説明される. 図 2 において, t^0 を現在の \bar{P} の最適解 (16), すなわち, $t^0 = (0, 0, \dots, 0)$ とする. $u_j v_j = 0$ の条件は, 解が平面 $u_j = 0$ 上か, あるいは $v_j = 0$ 上に位置することを要求するが, t^0 は条件を満たさない. ところで, t^0 は \bar{P} の基底解であるから, \bar{P} の拘束条件を等式に置いて得られる N 個の超平面の交点として与えられる. その N 個のうちの各 $N-1$ 個の超平面の交線として, t^0 から, N 本の線分が \bar{P} の許容領域側へ引かれるが, 交線が, 平面 $u_j = 0$ あるいは $v_j = 0$ に, 最初に交わる点を t^k , $k=1, 2, \dots, N$, としよう (交わらなければ, t^k は無限遠にとられる).

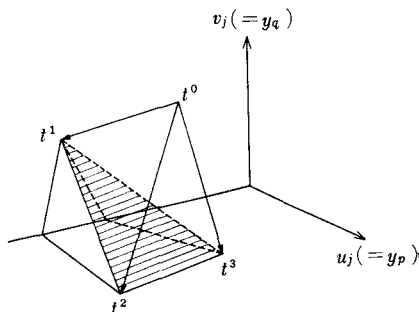


図 2 CP 問題の C-カットの図解

C-カットは, t^1, t^2, \dots, t^N を通る超平面の t^0 側, すなわち, N 単体 (t^0, t^1, \dots, t^N) を \bar{P} の許容領域から切除するものである (超平面上の点は除かない). 図 2 は, $N=3$ の場合であるが, 容易にわかるように, 切除された N 単体は, t^0 は含むが, P の許容解 ($u_j = 0$ あるいは $v_j = 0$ 上の点) を含まない. 次の定理は, 以上の事実を一般的に示すものである.

定理 3: P^* を P ((1) 式) の許容領域を含む許容領域を持つ LP 問題とし, (13) (16) によって, その最適解が与えられているとする. このとき, (16) が P の許容解ではないならば, (16) は拘束条件 (18) を満足しない ((18) は (16) を切除する) が, P の許容解は, すべて (18) を満足する.

証明: まず, \bar{P} の最適解 (16) は, (13) において $t_j = 0$, $j=1, 2, \dots, N$ と置くことによって得られる. ところが, (18) で $t_j = 0$, $j=1, 2, \dots, N$ と置けば, $s = -1 \not\geq 0$, つまり, (16) は (18) を満足しない. つぎに, (18) によって, 切除されるすべての $t = (t_1, t_2, \dots, t_N)$, $t_1 \geq 0, t_2 \geq 0, \dots, t_N \geq 0$, に対して $y_p y_q \neq 0$ であることを示す. これは, とりもなおさず, P の許容解 ($y_p y_q = 0$ を満たす) が (18) によって切除されないことを意味する. さて, (18) によって切除される t_j の値の組は,

$$(20) \quad \sum_{j=1}^N \beta_j t_j < 1$$

を満足する. ここで, 一般性を失うことなく,

$$\beta_1, \beta_2, \dots, \beta_r > 0$$

$$\beta_{r+1} = \beta_{r+2} = \dots = \beta_N = 0$$

と仮定する. $r=0$ ならば, (18) を満たす $t_j (\geq 0)$ は存在せず, P は不能である. したがって, 定理は成立している. $r > 0$ のとき

$$(21) \quad t^0 = (0, 0, \dots, 0)$$

$$t^1 = \left(-\frac{1}{\beta_1}, 0, \dots, 0 \right)$$

⋮

$$t^r = \left(0, \dots, 0, \frac{1}{\beta_r}, 0, \dots, 0 \right)$$

$$t^{r+1} = (0, \dots, 0, \mathbf{1}, 0, \dots, 0)$$

⋮

$$t^N = (0, \dots, 0, \mathbf{1})$$

と置く. このとき, $t_j \geq 0$, $j=1, 2, \dots, N$, でなければならぬことから, (20) を満たす任意の $t = (t_1, t_2, \dots, t_N)$ は, 半空間(20)と半空間 $t_j \geq 0$, $j=1, 2, \dots, N$, の共通部分 (N 単体) として

$$(22) \quad t = \sum_{k=0}^N \lambda_k t^k$$

$$\lambda_k \geq 0, \quad k=1, 2, \dots, N$$

$$\lambda_0 > 0$$

$$\sum_{k=0}^r \lambda_k = 1$$

と書くことができる ($\lambda_0 > 0$ は, t が超平面 $\sum \beta_j t_j = 1$ 上にはないことから得られる). ところで, (17) の仮定より

$$y_p^0 = \alpha_{p0} + \sum_{j=1}^N \alpha_{pj} t_j^0 = \alpha_{p0} > 0$$

$$y_q^0 = \alpha_{q0} + \sum_{j=1}^N \alpha_{qj} t_j^0 = \alpha_{q0} > 0$$

また, (19) における β_j の定め方より,

$$y_p^k = \alpha_{p0} + \sum_{j=1}^N \alpha_{pj} t_j^k \begin{cases} \geq \alpha_{p0} + \alpha_{pk} \left(-\frac{\alpha_{p0}}{\alpha_{pk}} \right) = 0 & \text{if } \alpha_{pk} < 0 \\ \geq \alpha_{p0} > 0 & \text{if } \alpha_{pk} \geq 0 \end{cases}$$

$$k=1, 2, \dots, r,$$

および

$$y_p^k = \sum_{j=1}^N \alpha_{pj} t_j^k = \alpha_{pk} \geq 0, \quad k=r+1, r+2, \dots, N$$

である。 y_q に対しても、同様の議論を適用できる。したがって、(22)の t を用いて、

$$\begin{aligned} y_p &= \alpha_{p_0} + \sum_{j=1}^N \alpha_{p_j} \sum_{k=0}^N \lambda_k t_j^k \\ &= \sum_{k=0}^r \lambda_k (\alpha_{p_0} + \sum_{j=1}^N \alpha_{p_j} t_j^k) + \sum_{k=r+1}^N \lambda_k \sum_{j=1}^N \alpha_{p_j} t_j^k \\ &= \sum_{k=0}^r \lambda_k y_p^k + \sum_{k=r+1}^N \lambda_k \alpha_{p_k} > 0 \end{aligned}$$

が証明される。同様にして、 $y_q > 0$ も証明できる。つまり、(18)によって切除される t は、必ず $y_p y_q > 0$ を満たす。 (証明終)

定理の P^* としては、たとえば、 \bar{P} 、あるいは、(18)式のカットを、次々と k 個追加して得られるLP問題 \bar{P}_k を考えることができる。

さて、C-カットを用いて、切除平面法を構成すると、次のようになる。

手順 1: $\bar{P}_0 = \bar{P}$ をシンプレックス法によって解く。 $k=0$ とし、手順 2 へ。

手順 2: \bar{P}_k の最適解が、 $u^T v = 0$ を満足すれば、計算終了、 P の最適解が得られた。 $u^T v \neq 0$ ならば、手順 3 へ進む。

手順 3: \bar{P}_k の C-カットを作り、(このとき(18)において $\beta_j = 0, j=1, 2, \dots, N$, ならば P は不能)

\bar{P}_k の最適解を与えるタブロに新しい行として追加する。このLP問題の最適解を双対シンプレックス法によって求め、タブロからカットの行を除いた後、 k を 1 増加し、手順 2 へ戻る。

(13)を \bar{P}_k の最終タブロとすると(このときの t_j は、(12)の w, x, u, v か、あるいは、(18)によって生成された変数 s のいずれか)、カット追加後の \bar{P}_{k+1} の初期タブロは、主許容条件

$$(23) \quad (\alpha_{10}, \alpha_{20}, \dots, \alpha_{M'0}) \geq 0$$

ただし、 $M' = M + 1$

は満足していないが ((18)より $\alpha_{M'0} = -1$ である)、双対許容条件

$$(24) \quad (\alpha_{01}, \alpha_{02}, \dots, \alpha_{0N}) \geq 0$$

は満足している。したがって、このタブロに、直接、双対シンプレックス法を適用し、 \bar{P}_{k+1} の最適解を得ることができるのである(手順 3)。

上の手順では、必ずしも有限回の収束を保証することはできないため(適当な仮定の下に、収束することは証明できるが)、次に、もう一種のカットを導入する。

\bar{P}_k のタブロにおいて、ある行 (z に対応する行でもよい)

$$(25) \quad y_p = \alpha_{p_0} + \sum_{j=1}^N \alpha_{p_j} t_j$$

の α_{p_0} が Δ (定理 2 参照) の整数倍ではないと仮定する。定理 2 によって、 α_{p_0} が Δ の整数倍となるような P の最適解が存在するから、 α_{p_0} を

$$(26)^3) \quad \left[\frac{\alpha_{p_0}}{\Delta} \right] \Delta$$

あるいは

3) $[\alpha]$ は α を越えない最大の整数。

$$\left[\frac{\alpha_{p0}}{\Delta} \right] \Delta + \Delta$$

に強制的に移すようなカットを追加することができる。そのようなカットの1例として、次のカットをあげる。これは、Gomory [5] が、混合整数計画問題に対して開発したカットの一般化とも考えることができる。

$$(27) \quad s = -1 + \sum_{j=1}^N \beta_j t_j$$

$$s \geq 0$$

ただし

$$\beta_j = \begin{cases} -\frac{\alpha_{pj}}{1-f_0} & \text{if } \alpha_{pj} \geq 0 \\ -\frac{\alpha_{pj}}{f_0} & \text{if } \alpha_{pj} < 0 \end{cases}$$

および

$$f_0 = \alpha_{p0} - \left[\frac{\alpha_{p0}}{\Delta} \right] \Delta (> 0)$$

(27)のカットを \bar{P}_k の Δ -カットと呼ぶ。

定理 4: P の許容解ではない \bar{P}_k の最適解は、拘束条件(27)を満足しない。また、 P の最適解で \bar{P} の基底解であるものは (定理 1, 2 参照) 必ず、(27)を満足する。

定理 4 の証明は省略する。先に述べた切除平面法において、C-カットと Δ -カットを混用すると、Gomory [6] が整数計画問題に用いたものと類似の辞書式順序を利用する方法によって、有限回で P の最適解を得ることが証明できる。しかし、 Δ -カットは、有限回収束を保証するための目的に導入されたものであって、 Δ の計算が必ずしも容易でないこと、および Δ が微量量であって、丸め誤差の観点からも問題があることを考えると、必ずしも有効なものではない。一般に、カットを分枝限定法の補助手段として用いる場合には、有限回収束が保証されていなくても、充分有用であるので、以後の議論では、一応 C-カットだけを想定しておく。実際の計算にも、C-カットのみを用いた。

6. 切除平面法の利用とペナルティ

4., 5 節の議論は、すべて P および \bar{P} に対して行なったが、 P を部分問題 $P(S)$ 、 \bar{P} を $\bar{P}(S)$ で置き換えても、同様の議論ができることは、明らかである。 \bar{P} の場合と同様、 $\bar{P}(S)$ に C-カットを k 個追加して得られる問題を $\bar{P}_k(S)$ と記す。さて、 $P(S)$ をテスト (3 節手順 3) する目的に、 $\bar{P}_k(S)$ を用いる (7 節参照) とき、もし $u^T v = 0$ を満足する $\bar{P}_k(S)$ の最適解が得られれば、これは $P(S)$ の最適解でもあるから、分枝限定法の手順 3(i) から手順 4 に進むことができる。しかし、 k をある程度大きくしても $P(S)$ の許容解が得られないならば、適当なところで打ち切って次に進むのが、計算速度の点から望ましい。そのとき、 $P(S)$ の値、最適解等に関して、 $\bar{P}_k(S)$ の最適解を与えるタブロから得られる情報を、できるだけ利用すれば、それだけ計算速

度を高めることができよう。その目的に、ここで、変数 $y_p (=u_j \text{ or } v_j)$ が目標関数に与える影響の尺度であるペナルティ $p(u_j), p(v_j)$ を導入する。これは、混合整数計画問題に対する、Driebeek の手法 [2] の拡張とも考えられる。すなわち、ペナルティ $p(u_j)$ は、 $\bar{P}_k(S)$ の最終タブロにおいて、 $u_j=0$ と固定したときの目標関数の値 z の増加分の下限值として定義される。すなわち、 $\bar{P}_k(S)$ の最適解の値を $\bar{z}_k(S)$ とすれば、 $P(S, j)$ の最適解の値 $z^0(S, j)$ は

$$(28) \quad z^0(S, j) \geq \bar{z}_k(S) + p(u_j)$$

を満足する。 $p(v_j)$ についても、同様に定義される。 $p(u_j), p(v_j)$ を計算する直接的な方法は、 $\bar{P}_k(S)$ のタブロにおいて、 u_j あるいは v_j を 0 に固定し、LP のパラメトリック・プログラミングの手法にしたがって、 $\bar{P}_k(S, j)$ あるいは $\bar{P}_k(S, -j)$ の最適解を求めることである。しかし、この方法は、一般に、計算時間の点から問題があるため、以下では、ピボット操作 1 回分に相当する量を用いる。

$\bar{P}_k(S)$ の最終タブロにおいて、 $y_p (=u_j \text{ or } v_j)$ が

$$(29) \quad y_p = \alpha_{p0} + \sum_{j=1}^N \alpha_{pj} t_j$$

と表わされ、 $\alpha_{p0} > 0$ とする。 $y_p = 0$ とするためには、ピボット操作を加え、 y_p を強制的に非基底変数とすればよい。その結果得られるタブロが、双対許容条件を満たすために、

$$(30) \quad \frac{\alpha_{0j_0}}{\alpha_{pj_0}} = \max \left\{ \frac{\alpha_{0j}}{\alpha_{pj}} \mid \alpha_{pj} < 0 \right\}$$

なる α_{pj_0} がピボットに選ばれる。ピボット操作の後の新しい $(0, 0)$ 要素、 α'_{00} は

$$(31) \quad \alpha'_{00} = \alpha_{00} - \frac{\alpha_{p0} \alpha_{0j_0}}{\alpha_{pj_0}}$$

である。これ以後、双対シンプレックス法を適用すれば、 $(0, 0)$ 要素の値は、 α'_{00} から単調に増加して、最適解に至る。したがって、

$$(32) \quad p(y_p) = -\frac{\alpha_{p0} \alpha_{0j_0}}{\alpha_{pj_0}} (\geq 0)$$

$$y_p = u_j \text{ or } v_j$$

と置くことができる。

7. 分枝限定法、手順 3：部分問題のテスト

$\bar{P}_k(S)$ が $P(S)$ に比べ、緩い拘束条件下の問題であるという事実 (定理 3) から、次の事実が容易に示され、分枝限定法の手順 3 に利用できる。

(i) $\bar{P}_k(S)$ の最適解が相補条件 $u^T v = 0$ を満足すれば、 $P(S)$ の最適解でもある。(手順 3, (i)).

(ii) $\bar{P}_k(S)$ が不能であれば、 $P(S)$ も不能。(手順 3, (ii))

また、同様の議論によって、

$$z^0 \geq z^0(S) \geq \bar{z}_k(S) \geq \bar{z}(S)$$

が得られる。ただし、 $z^0, z^0(S), \bar{z}_k(S), \bar{z}(S)$ は、それぞれ、 $P, P(S), \bar{P}_k(S), \bar{P}(S)$ の最適解の値である。したがって、次の性質を利用できる。

(iii) $\bar{z}_k(S) \geq z^*$ (暫定解の値) ならば、 $P(S)$ は暫定解より小さな値を与え得ない。(手順 3, (ii)).

ペナルティの持つ式(28)の性質を利用すれば、手順 3, (iii)を構成できる。

(iv) $\bar{z}_k(S) + p(u_j) \geq z^*$ ならば、 $P(S, j)$ は暫定解より小さな値を持たず、終端できる。

$\bar{z}_k(S) + p(v_j) \geq z^*$ ならば、 $P(S, -j)$ を終端できる。

$\bar{P}_k(S)$ を解くとき、 k を大きくするほど、上の (i) (ii) (iii) (iv) を得る可能性が大きくなり、それだけ生成される部分問題の個数も少なくてすむが、 $\bar{P}_k(S)$ の計算時間が大きくなる。適当な k を定めるのは難しい問題であって、実際の計算結果から決定しなければならないが、12節の計算実験では、与えられた $\varepsilon > 0$ に対して

$$(33) \quad \frac{\bar{z}_k(S) - \bar{z}_{k-1}(S)}{\bar{z}_k(S)} \leq \varepsilon$$

を最初に満たす k で打ち切っている。つまり、目標関数の値の変化が小さくなったところで、カット生成を打ち切るという方針である。

8. 分枝限定法, 手順 5: 分枝変数の決定

手順 5 の分枝変数の決定法としては、ペナルティに基づく、次の基準によった。これは、 $u_j = 0$ と $v_j = 0$ に固定して得られる 2 個の部分問題の値の差を、なるべく大きくしようとするものである。すなわち、

$$|p(u_{j_0}) - p(v_{j_0})| = \max \{ |p(u_j) - p(v_j)| \mid j \in F_S, u_j, v_j > 0 \}$$

を満たす自由変数 u_{j_0}, v_{j_0} を分枝変数とする。分枝限定法 (Branch and Bound Method) という名称を最初に用いた Little らの Traveling Salesman 問題に関する論文[14]も、使用するパラメータは異なるが、同様の方針によっている。

9. 分枝限定法, 手順 2: 活性問題の選定

手順 2 の活性問題の選定法には、一般の分枝限定法に用いられている基準から判断して、次の 2 種が有力であろう。

(i) 活性問題のうちで、最小の $\bar{z}_k(S)$ を持つ部分問題 $P(S)$ を選ぶ。

(ii) 最も新しく生成された問題を選ぶ。このとき、手順 5 においては、2 個の部分問題が同時に得られるが、小さな値を持つほうを選ぶ。

(i) は実際に生成される部分問題の個数を少なく保つ傾向を持つ。しかし、記憶容量の点で問題がある。一方(ii)の方法は、生成される部分問題の個数は多くなるが、その生成方法が簡単であること、記憶容量をあまり必要としない、という点に特長を持つ。10 節のアルゴリズムでは、簡単のために、(ii)の方法を用いている。

10. 分枝限定法に基づくアルゴリズム

以上をまとめて、CP 問題の分枝限定法を基本とするアルゴリズムを構成すると、図3に書かれるようなフローチャートを得る。図3の各部分はこれまでの説明から、容易に理解されるであらう。ただし、図3では部分解 S

を変数の固定順序を示す目的にも用いている。たとえば、 $S = \{-4, 1, -2\}$ は、まず、 v_4 が0に固定され、次に u_1 が0に、最後に、 v_2 が0に固定されたことを示す。また、 S の要素に下線を付して、 $S = \{S_1, \underline{j}, S_2\}$ (あるいは $S = \{S_1, -\underline{j}, S_2\}$) を $S' = \{S_1, -j\}$ (あるいは $S' = \{S_1, j\}$)、または S' から得られたすべての末端部分問題が終端されていることを示す目的に用いる。たとえば、 $S = \{-4, \underline{1}, -2\}$ は部分解 $S = \{-4, -1\}$ (あるいは、 $\{-4, -1\}$ から生成されたすべての末端部分問題) が、終端されていることを意味する。この方法の正当性等については、たとえば、Geoffrion [3] 等に詳しいので省略する。

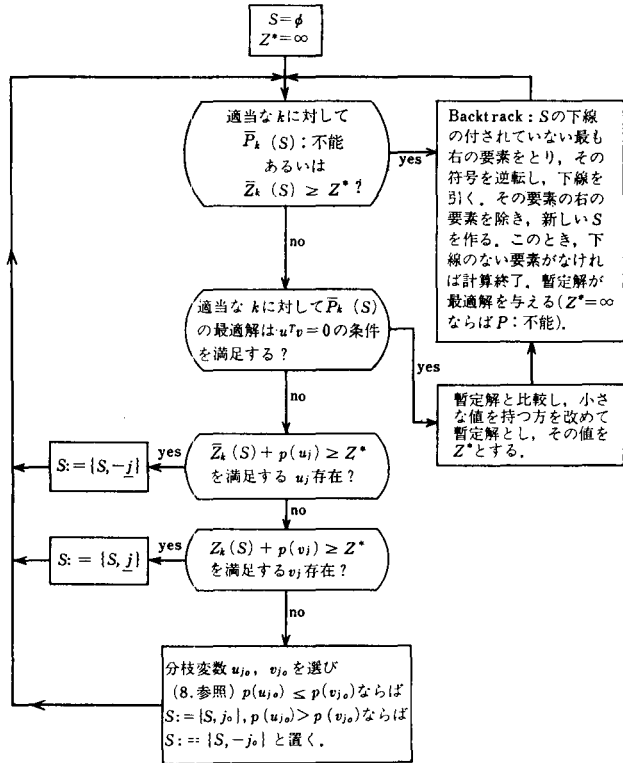


図3 CP 問題の分枝限定法に基づくアルゴリズム

11. 分枝限定法による例題

10 節に述べた図3のアルゴリズムを用いて、具体的な例題を解いてみよう。

P : 目標関数 $3x_1 + u_1 + 2u_2 + v_1 + 3v_2 \rightarrow$ 最小

- 拘束条件 $x_1 + 3u_1 + u_2 + v_1 + 2v_2 \geq 3$
- $3x_1 + u_1 + u_2 + 2v_1 + v_2 \geq 4$
- $3u_2 + v_1 + 3v_2 \geq 3$
- $2u_1 + 5u_2 + v_1 \geq 4$
- $u_1v_1 = 0, u_2v_2 = 0$
- $x_1, u_1, u_2, v_1, v_2 \geq 0$

スラック変数 w_1, w_2, w_3, w_4 を拘束条件の上の4式に対応させ、 \bar{P} のシンプレックス・タブロ

表 1 $\bar{P}(\phi)$ の初期タブロ

| | 1 | x_1 | u_1 | u_2 | v_1 | v_2 |
|---------|----|-------|-------|-------|-------|-------|
| $z =$ | 0 | 3 | 1 | 2 | 1 | 3 |
| $w_1 =$ | -3 | 1 | 3 | 1 | 1 | 2 |
| $w_2 =$ | -4 | 3 | 1 | 1 | 2 | 1 |
| $w_3 =$ | -3 | 0 | 0 | 3 | 1 | 3 |
| $w_4 =$ | -4 | 0 | 2 | 5 | 1 | 0 |

表 2 $\bar{P}(\phi)$ の最終タブロ

| | 1 | x_1 | w_1 | w_3 | w_2 | v_2 |
|---------|-------|--------|-------|-------|--------|--------|
| $z =$ | 37/13 | 32/13 | 4/13 | 7/13 | 1/13 | 9/13 |
| $w_4 =$ | 7/13 | 18/13 | 12/13 | 21/13 | -10/13 | -77/13 |
| $u_1 =$ | 4/13 | 1/13 | 5/13 | -1/13 | -2/13 | -5/13 |
| $u_2 =$ | 6/13 | 8/13 | 1/13 | 5/13 | -3/13 | -14/13 |
| $v_1 =$ | 21/13 | -24/13 | -3/13 | -2/13 | 9/13 | 3/13 |
| $s_1 =$ | -1 | 8/7 | 1/7 | 1/4 | 1/2 | 5/4 |

表 3 $\bar{P}_1(\phi)$ の最終タブロ

| | 1 | x_1 | w_4 | w_3 | s_1 | v_2 |
|---------|--------|---------|--------|-------|--------|-------|
| $z =$ | 163/52 | 339/182 | 14/104 | 3/13 | 61/169 | 27/26 |
| $w_1 =$ | 7/8 | -11/4 | 7/8 | -7/4 | 35/26 | 7/2 |
| $w_2 =$ | 7/4 | -3/2 | -1/4 | 0 | 21/13 | -7/2 |
| $u_1 =$ | 3/8 | -21/28 | 3/8 | -3/4* | 7/26 | 3/2 |
| $u_2 =$ | 1/8 | 3/4 | 1/8 | 1/4 | -7/26 | 0 |
| $v_1 =$ | 21/8 | -63/28 | -3/8 | 1/4 | 21/26 | -3 |

表 4 $\bar{P}(1)$ の最終タブロ

| | 1 | x_1 | w_4 | u_1 | s_1 | v_2 |
|---------|------|---------|-------|-------|--------|-------|
| $z =$ | 13/4 | 297/182 | 1/4 | -4/13 | 75/169 | 3/2 |
| $w_1 =$ | 0 | -1 | 0 | 7/3 | 28/39 | 0 |
| $w_2 =$ | 7/4 | -3/2 | -1/4 | 0 | 21/13 | -7/2 |
| $w_3 =$ | 1/2 | -1 | 1/2 | -4/3 | 14/39 | 2 |
| $u_2 =$ | 1/4 | 1/2 | 1/4 | -1/3 | -7/39 | -5/2 |
| $v_1 =$ | 11/4 | -5/2 | -1/4 | -1/3 | 35/39 | -5/2 |

$$(34) \quad p(u_1) = -\frac{3}{13} \cdot \frac{3}{8} \cdot \left(-\frac{4}{3}\right) = \frac{3}{26}$$

$$p(v_1) = -\frac{27}{26} \cdot \frac{21}{8} \cdot \left(-\frac{1}{3}\right) = \frac{189}{208}$$

である。分枝変数の選定を、8節の方法にしたがって行ない、 $p(u_1) < p(v_1)$ を考慮すると、図3のアルゴリズムにしたがって次に生成される部分問題は $P(1)$ である。

$u_1 = 0$ の下での最適解を得るには、表3の u_1 をピボット操作によって、非基底解とし、その後 u_1 列を無視して、双対シンプレックス法を適用すればよい。まず、(30)式を満たす、星印の

を書くと、表1を得る(非基底変数に対応する行は、単位ベクトルを含むだけであるので、省略してある)。表1は、双対許容条件： $(\alpha_{01}, \alpha_{02}, \dots, \alpha_{0N}) \geq 0$ は満足するが、主許容条件： $(\alpha_{10}, \alpha_{20}, \dots, \alpha_{M0}) \geq 0$ は満足しない。したがって、双対シンプレックス法によって、 $\bar{P}(\phi)$ の最適解を求めると、3回ピボット操作を加えて、表2を得る。表2は解 $(x_1, u_1, u_2, v_1, v_2) = (0, 4/13, 6/13, 21/13, 0)$ を表わしているが、 P の拘束条件 $u_1 v_1 = 0$ を満足しない。そこで、(18)にしたがって、 u_1 行、 v_1 行からC-カットを作ると、表2の s_1 行を得る。

以後、再び双対シンプレックス法に基づき、2回ピボット操作を加え、表3を得る。表3は $\bar{P}_1(\phi)$ の最適解 $(x_1, u_1, u_2, v_1, v_2) = (0, 3/8, 1/8, 21/8, 0)$ を与えるが、まだ、 P の条件 $u_1 v_1 = 0$ を満足しない。そこで、カットの生成を打ち切り(実は、もう1度カットを加えると、 $P(\phi)$ の最適解が得られるが)、分枝限定法の手順にしたがって、6節のペナルティを計算する。(32)式からわかるように

要素がピボットとして選ばれ、ピボット操作の後、表4を得る ($\bar{P}(1)$ の定義からは、表2から出発して得られる最適解を利用すべきであるが、カットが含まれているだけ、表3からのほうが精度の高い結果を与える)。表4は

$$(35) \quad z=13/4$$

$$(x_1, u_1, u_2, v_1, v_2) = (0, 0, 1/4, 11/4, 0)$$

を表わし、これは、 P の許容解である。したがって、(35)を暫定解として記憶する。 $z^*=13/4$ 。

図3における、次のステップは Backtrack であって、部分問題 $P(-1)$ が生成される。ところが、(34)式で求めたペナルティを用いると、 $P(-1)$ の最適解の値 $z^0(-1)$ は、

$$\begin{aligned} z^0(-1) &\geq \bar{z}_1(\phi) + p(v_1) \\ &= 163/52 + 189/208 \\ &> z^* = 13/4 \end{aligned}$$

である。したがって、 $P(-1)$ から P の最適解を得る可能性はない。再び、Backtrack を行なうと、活性問題の存在しないことが示され、計算終了に至る。結局、(35)が P の最適解である。

12. 計算結果

図3のアルゴリズムの能率を調べるために、いくつかの計算を試みた。その計算結果を次に示す。使用した問題は、0-1 問題 (全整数および混合整数) の代表的問題として、しばしば使用さ

表5 代表的な問題の計算結果 (注参照)

| 問題 | $n^{(a)}$ | $m^{(b)}$ | $n+2m^{(c)}$ | $p^{(d)}$ | 計算結果 (時間は秒) | | | | | | | | |
|---|-----------|-----------|--------------|-----------|-----------------------|-----|----------------------|-----|----------------------|-----|----------|--------------------|---------|
| | | | | | 分枝限定法 | | | | 切除平面法 ^(g) | | 他のアルゴリズム | | |
| | | | | | カット用いず ^(e) | | カット使用 ^(f) | | カット数 | 時間 | 時間 | コンピュータ | |
| | | | | | 部分問題数 ⁽¹⁾ | 時間 | 部分問題数 ⁽¹⁾ | 時間 | | | | | |
| Heldi Fixed Charge ^(h) | 1 | 3 | 2 | 7 | 6 | 6 | 0.1 | 5 | 0.1 | 8 | 0.4 | | |
| | 2 | 3 | 2 | 7 | 6 | 5 | 0.1 | 1 | 0.1 | 4 | 0.3 | | |
| | 3 | 3 | 2 | 7 | 6 | 5 | 0.1 | 1 | 0.1 | 3 | 0.3 | | |
| | 4 | 3 | 2 | 7 | 6 | 5 | 0.1 | 1 | 0.1 | 5 | 0.3 | | |
| | 7 | 3 | 2 | 7 | 6 | 6 | 0.1 | 4 | 0.1 | — | — | 1.8 ^(m) | 7044 |
| | 8 | 3 | 2 | 7 | 6 | 5 | 0.1 | 1 | 0.1 | — | — | 3.0 | 7044 |
| 10 | 6 | 6 | 18 | 16 | 7 | 0.8 | 1 | 0.5 | 10 | 1.9 | 3.6 | 7044 | |
| Petersen R and D ⁽ⁱ⁾ | 1 | 0 | 6 | 12 | 16 | 13 | 0.3 | 5 | 0.3 | 26 | 1.9 | | |
| | 2 | 0 | 10 | 20 | 20 | 31 | 1.4 | 7 | 0.7 | (o) | (o) | 2.4 ^(m) | 7044 |
| | 3 | 0 | 15 | 30 | 25 | 94 | 7.2 | 74 | 8.0 | | | | |
| | 4 | 0 | 20 | 40 | 30 | 59 | 7.5 | 27 | 3.6 | | | | |
| | 5 | 0 | 28 | 56 | 38 | 143 | 37.1 | 74 | 22.2 | | | 14.4 | 7044 |
| Threshold Gates ^(j) Network Design | 17 | 4 | 25 | 28 | 10 | 1.7 | 7 | 4.4 | | | | | |
| Graph ^(k) | 4 nodes | 0 | 6 | 12 | 14 | 1 | 0.1 | 1 | 0.1 | | | 0.9 ⁽ⁿ⁾ | CDC3600 |
| | 5 nodes | 0 | 10 | 20 | 30 | 20 | 1.4 | 14 | 1.1 | | | 3.4 | CDC3600 |

表 6 相補的変数の個数の効果 (注参照)

| | $n^{(a)}$ | $m^{(b)}$ | $n+2m^{(c)}$ | $p^{(d)}$ | 時間 ^(e) (秒) |
|----|-----------|-----------|--------------|-----------|-----------------------|
| 1 | 18 | 2 | 22 | 30 | 0.9 |
| 2 | 16 | 4 | 24 | 30 | 1.0 |
| 3 | 14 | 6 | 26 | 30 | 1.6 |
| 4 | 12 | 8 | 28 | 30 | 1.9 |
| 5 | 10 | 10 | 30 | 30 | 2.3 |
| 6 | 8 | 12 | 32 | 30 | 2.4 |
| 7 | 6 | 14 | 34 | 30 | 3.7 |
| 8 | 4 | 16 | 36 | 30 | 4.4 |
| 9 | 2 | 18 | 38 | 30 | 7.0 |
| 10 | 0 | 20 | 40 | 30 | 8.1 |

注 (表 5, 表 6 共通)

- (a) 連続変数の個数 ((1)式参照)
 (b) 相補的変数の個数 (")
 (c) 変数の総数 (")
 (d) 拘束条件の個数, ただし, $u^T v=0, x, u, v \geq 0$ は除く (")
 (e) 図 3 のアルゴリズムで $k=0$ と置いたもの
 (f) カットは $\frac{\bar{z}_k(S) - \bar{z}_{k-1}(S)}{\bar{z}_k(S)} > 0.001$ ならば生成される
 (g) C-カットのみによる切除平面法 (5.参照)
 (h) Haldi [7] の Fixed Charge 問題
 (i) Petersen [17] の R and D 選択問題
 (j) Muroga-Ibaraki [16] の Threshold Gates による論理回路設計問題
 (k) Trauth-Woolsey [18] によるグラフに関する問題
 (l) 分枝限定法において実際にテストされた部分問題の個数
 (m) Geoffrion [4] の計算結果
 (n) Trauth-Woolsey [18] の計算結果. ILP 2-2 使用
 (o) この問題は丸め誤差のため収束せず
 (p) アルゴリズムは分枝限定法. カット使用せず

が満足されるまで生成し, 得られた $\bar{P}_k(S)$ の結果を用いるもの, (iii) 切除平面法 (5節参照) のみ, の 3 種である. 切除平面法の場合のみ, 収束速度, 丸め誤差の影響等で, 大きな問題には適用できなかった. もちろん, 注意深くプログラムすれば, もう少し大きな問題に適用できると思われるが, いずれにしても, 計算速度は, 分枝限定法に劣るようである.

表 5 の結果は, 0-1 問題を, CP 問題として解いても, 0-1 問題のアルゴリズムに比し, 決して遜色のないことを示していると思われる.

表 6 は, Petersen [17] の R and D 選択問題 2 を用いて, 相補的変数の個数の効果を調べたものである. すなわち, 20 個の変数のうち, それぞれ, 2, 4, 6, ..., 20 個を,

$$v_j = 1 - u_j$$

$$u_j v_j = 0$$

の条件を入れることによって, 相補的変数とし, 得られた問題を, カットを用いない ($k=0$) 分枝限定法によって解いた. 各行とも, ($m=20$ を除いて), ランダムに相補的変数を選ぶことによ

れているものを, (7)にしたがって変換して用いた. 2 節に示したように, CP 問題は, 0-1 問題だけでなく, より広い範囲の問題を含むが, 既存のアルゴリズムとの比較を行なうために, 0-1 問題を利用したものである.

使用コンピュータは, 京都大学大型計算機センターの FACOM 230/60, コードは FORT-RAN によった.

表 5 は, 各問題に対する代表的結果を示したものであって, 整数計画アルゴリズムの代表的な結果も, あわせて記している. 使用したアルゴリズムは, (i) 図 3 の分枝限定法に基づくアルゴリズムにおいて, $k=0$ に固定 (つまり, カットを併用せず) したものの, (ii) C-カットを

$$\frac{\bar{z}_k(S) - \bar{z}_{k-1}(S)}{\bar{z}_k(S)} \leq 0.001$$

って得られた問題 2 個の平均である。表 6 の結果では、計算時間は相補的変数の個数 m とともに指数関数的に増大するようである。

む す び

本論文では、相補的プログラミングという概念を導入し、その応用、およびアルゴリズムについて議論した。ここで提案した分枝限定法アルゴリズムの能率は、0-1 問題等では、既存の整数計画アルゴリズムに遜色ないものの、実用的見地からは、まだ、今後の改良が望まれる。また、この種の問題の性質の理論的研究も興味ある分野である。この意味で、相補的プログラミング問題にある種の双対定理の成立することを示した[11]をあげておく。

おわりに、日頃、京都大学三根久教授にご指導いただいております、この論文に関しても有益なご助言をいただいたことを申し添え、謝意を示したい。

参 考 文 献

- [1] Cottle, R. W. and G. B. Dantzig, "Complementary Pivot Theory of Mathematical Programming," *J. of Linear Algebra and Its Applications*, **1** (1968), 103-125.
- [2] Driebeek, N. J., "An Algorithm for the Solution of Mixed Integer Programming Problems", *Management Science*, **12** (1966), 576-587.
- [3] Geoffrion, A. M., "Integer Programming by Implicit Enumeration and Balas' Method," *SIAM Review*, **9** (1967), 178-190.
- [4] Geoffrion, A. M., "An Improved Implicit Enumeration Approach for Integer Programming", *Operations Research*, **17** (1969), 437-454.
- [5] Gomory, R. E., *An Algorithm for the Mixed Integer Problem*, P-1885, The RAND Corp. Inc. 1960.
- [6] Gomory, R. E., "An Algorithm for Integer Solutions to Linear Program", in *Recent Advances in Mathematical Programming*, Graves, R. L. and P. Wolfe (eds.) McGraw-Hill, New York, 1963.
- [7] Haldi, J., "25 Integer Programming Test Problems," Working Paper, Graduate School of Business, Stanford Univ., 1964.
- [8] 茨木俊秀, "相補的プログラミング", OR 学会秋季研究発表会, 1969.
- [9] 茨木俊秀, "Complementary Programming 問題における Cutting Plane 法", OR 学会春季研究発表会, 1970.
- [10] 茨木俊秀, "相補的プログラミングの計算実験", OR 学会秋季研究発表会, 1970.
- [11] Kaji, N. and T. Ibaraki, "Duality in Complementary Programming," 4th Hawaii Int. Conf. on System Science, 1971.
- [12] Lawler, E. L. and D. E. Wood, "Branch-and-Bound Methods: A survey," *Operations Research*, **14** (1966), 699-719.
- [13] Lemke, C. E. and J. T. Howson, "Equilibrium Points of Bi-Matrix Games," *SIAM Journal*, **12** (1964), 413-423.
- [14] Little, J. D. C., K. G. Murty, D. W. Sweeney and C. Karel, "An Algorithm for the Traveling Salesman Problem", *Operations Research*, **11** (1963), 972-989.
- [15] Mitten, L. G., "Branch-and-Bound Methods: General Formulation and Properties," *Operations Research*, **18** (1970), 24-34.
- [16] Muroga, S. and T. Ibaraki, "Logical Design of an Optimum Network by Integer Linear Programming I", Rept. no. 264, Dept. of Computer Science, Univ. of Illinois, 1968.
- [17] Petersen, C. C., "Computational Experience with Variants of the Balas' Algorithm Applied to the Selection of R and D Projects," *Management Science*, **13** (1967), 736-750.
- [18] Trauth, C. A. and R. E. Woolsey, "Integer Programming: A Study in Computational Efficiency," *Management Science*, **15** (1969), 481-493.