Proceedings of the Risa Consortium
1997 pp. 101– pp. 112

非線形(多項式)計画問題の代数的解法

白石 啓一[*]　甲斐 博[†]　齋藤 友克[‡]　野田 松太郎[§]
愛媛大学 工学部　愛媛大学 工学部　上智大学 理工学部　愛媛大学 工学部

(Received 1997/7/23 Revised 1997/11/27)

1. はじめに

現在脚光を浴びている情報システム工学の主要な分野に、システムの最適化を行うための数理計画法がある。これは、

> 数式で与えられた制約のもとで、数式として表現された目的関数を最大(または最小)にするための数理的手法

ということができる。良く知られているように、数理計画法で取り扱う問題は変数が連続の場合と離散的である場合とに大別される。後者は組合せ最適化と呼ばれるもので、整数計画法や動的計画法が含まれる。一方、前者は制約関数や目的関数の形状によって、線形計画問題と非線形計画問題とに分けられる。連続変数の場合の計画問題は一般に次のように書ける。

$$\begin{array}{ll} \text{目的関数} & f(x_1, x_2, \ldots, x_n) = f(\boldsymbol{x}) \quad \rightarrow \quad \text{最大(小)} \\ \text{制約条件} & \left\{ \begin{array}{ll} g_j(\boldsymbol{x}) \leq 0 & (j = 1, 2, \ldots, m) \\ h_k(\boldsymbol{x}) = 0 & (k = 1, 2, \ldots, l) \end{array} \right. \end{array} \tag{1}$$

線形計画問題に対しては、シンプレックス法(単体法)、内点法等が発表されており、容易に最適解を求めることができる。一方、非線形計画問題の場合は、最急降下法、ニュートン法の活用によって、それなりの結果を得ることは可能であるが、これらで求まる最適解は局所的最適解であり、大域的最適解を得るためには、初期値を選び直して何度も計算を繰り返す

*shira@hpc.cs.ehime-u.ac.jp
†kai@cs.ehime-u.ac.jp
‡saito@mm.sophia.ac.jp
§noda@cs.ehime-u.ac.jp

必要がある。また、初期値の選び方によっては、最適解が求まらない場合もある。さらに数値的手法の特色として、大域的最適解に近い値をもつ局所的最適解をあたかも大域的最適解のようにみなす危険性も大きい。

これらの困難に対応し、安定に大域的最適解を得るためには、制約条件で定められる全領域を探索し、かつ厳密に大域的最適解を見出す手法が必要となる。このような場合には、一般に代数的手法を用いることをまず思い浮かべるが、代数的手法ではメモリを消費し、計算時間も大量に要するのが欠点であり、実用化はされてはいなかった。すでに、代数的手法を使って非線形計画問題を解こうとする試みは、従来から Quantifier Elimination (限量子消去) の立場からの接近があった [1] し、同じような考えを用いて不等式系を解析する試み [2] もある。しかし、いずれも数理計画問題を系統的に扱おうという視点はなかった。一方、連立代数方程式を 1 変数化するために用いられるグレブナ基底を高速に求める手法が提案されていること [3] を考えると、非線形計画問題を代数的手法を用いて安定にかつ比較的高速に解く手法が一般化されても不思議ではない。そこで、本論の主題は、非線形計画問題を代数的に解く新しい手法の確立を目指したものである。以下、非線形計画問題に焦点を絞って、従来の方法を概括し、ついで本論で提唱する代数的方法の有効性を具体的な計画問題の例とともに述べる。この方法のより広い分野の非線形計画問題への適用の可能性や今後の発展方向についても言及する。

2. 非線形計画問題に対する従来の解法

一般に数理計画問題は (1) のように書けるが、制約集合

$$\mathbb{X} = \{\boldsymbol{x} \in \mathbb{R}^n \mid g_j(\boldsymbol{x}) \leq 0 (j = 1, 2, \ldots, m), h_k(\boldsymbol{x}) = 0 (k = 1, 2, \ldots, l)\} \tag{2}$$

を用いれば、次のように略記できる。

$$\begin{array}{lccc} \text{目的関数} & f(\boldsymbol{x}) & \rightarrow & \text{最大 (小)} \\ \text{制約条件} & \boldsymbol{x} \in \mathbb{X} & & \end{array} \tag{3}$$

最大化問題は、目的関数の符号を変えるだけで、最小化問題に帰着できる。このため、本論では最小化問題のみを扱う。

まず、制約条件のない最小化問題

$$\begin{array}{lccc} \text{目的関数} & f(\boldsymbol{x}) & \rightarrow & \text{最小} \\ \text{制約条件} & \boldsymbol{x} \in \mathbb{R}^n & & \end{array} \tag{4}$$

の解法について述べる。制約条件のない最小化問題の解法には、降下法があり、勾配ベクト

ルの選び方により、最急降下法、ニュートン法、共役勾配法と呼ばれる。降下法のアルゴリズムを述べる。

降下法のアルゴリズム

手順 1: 初期点 $\boldsymbol{x}^1$ を選び、$l=1$ とする。

手順 2: 現在の点 $\boldsymbol{x}^l$ において停止規準を満たせば終了。そうでなければ降下方向 $\boldsymbol{d}^l$ を求める。

手順 3: 1 次元探索問題を解き、ステップ幅 α^lを求め、$\boldsymbol{x}^{l+1}=\boldsymbol{x}^l+\alpha^l\boldsymbol{d}^l, l=l+1$ として手順 2 へもどる。

ここで、$\boldsymbol{d}^l=-\nabla^T f(\boldsymbol{x})$ により、$\boldsymbol{x}^l$ の近傍で f を最も急速に減少させる最急降下方向に選ぶ手法は最急降下法と呼ばれる。また、Hesse 行列 $\nabla^2 f(\boldsymbol{x}^l)$ の逆行列を求め$\boldsymbol{d}^l=-[\nabla^2 f(\boldsymbol{x}^l)]^{-1}\nabla^T f(\boldsymbol{x}^l)$ とする手法はニュートン法と呼ばれる。

次に、制約条件のある最小化問題

$$\begin{array}{ll} \text{目的関数} & f(\boldsymbol{x}) \quad \rightarrow \quad \text{最小} \\ \text{制約条件} & \boldsymbol{x}\in\mathbb{X} \end{array} \tag{5}$$

の解法について述べる。制約条件のある最小化問題の解法には、変換法、一般縮小勾配法といった方法がある。ここでは、変換法の一つであるペナルティ法を述べる。

ペナルティ法は、制約条件のある問題 (5) に対して、制約を満たさない点 $\boldsymbol{x}\notin\mathbb{X}$ には非常に大きなペナルティを表す項 $rP(\boldsymbol{x})$ を目的関数 $f(\boldsymbol{x})$ に付け加えて得られる制約条件のない問題

$$\begin{array}{ll} \text{目的関数} & F(\boldsymbol{x},r)=f(\boldsymbol{x})+rP(\boldsymbol{x}) \\ \text{制約条件} & \boldsymbol{x}\in\mathbb{R}^n \end{array} \tag{6}$$

を考え、この問題をパラメータ r を変化させて繰り返し解くことにより、もとの制約条件のある問題 (5) の最適解を求めるという最適化手法である。ここで、$P(\boldsymbol{x})$ はペナルティ関数と呼ばれる連続関数で

$$P(\boldsymbol{x})\begin{cases} =0 & \boldsymbol{x}\in\mathbb{X} \\ >0 & \boldsymbol{x}\notin\mathbb{X} \end{cases} \tag{7}$$

を満たし、r は正のスカラーのパラメータである。

問題 (6) は $\boldsymbol{x}\in\mathbb{X}$ のとき $P(\boldsymbol{x})=0$ であるから問題 (5) と等価であり、また、$\boldsymbol{x}\notin\mathbb{X}$ のとき $P(\boldsymbol{x})>0$ であるから $r\to\infty$ に対して $F(\boldsymbol{x},r)$ は無限大のペナルティが課せられること

になり、 $\boldsymbol{x} \notin \mathbb{X}$ なる点は $F(\boldsymbol{x}, r)$ の最小点になり得ないことより、$r \to \infty$ のとき問題 (6) はもとの問題 (5) の解へ収束することが期待できる。

問題 (6) は制約条件のない問題となっているので、降下法と同様のアルゴリズムで解くことができる。そのアルゴリズムを述べる。

ペナルティ法のアルゴリズム

手順 1: 初期点 $\boldsymbol{x}^1$ と初期パラメータ r^1 を選び、 $l = 1$ とする。

手順 2: $F(\boldsymbol{x}, r^l)$ の最小点を求める。

手順 3: 現在の点 $\boldsymbol{x}^{l+1}$ において停止基準を満たせば終了。そうでなければ $r^{l+1} = cr^l, c > 1$ とし、$l = l + 1$ として手順 2 へもどる。

ここで手順 2 の $F(\boldsymbol{x}, r^l)$ の最小化は先に述べた制約条件のない問題の最適化手法を適用すればよい。

以上述べた通り、従来の解法は基本的にニュートン法の活用であり、極小値 (局所的最適解) を求める手法にすぎない。そのため、大域的最適解を得るためには、全ての極小値を求め、その最小値を選ぶ必要がある。これは、初期値を変更しての再計算の繰り返しを必要とし、大域的最適解を見出すことを困難にしている。

3. 代数的解法

再び、数理計画問題 (1) を考える。ただし、ここでは目的関数と制約条件はすべて多項式で記述されているとする。すなわち、 $f(x), g_j(x), h_k(x)$ は $\boldsymbol{x} \in \mathbb{R}^n$ に対して定義される実数値関数であり、 $f(\boldsymbol{x}), g_j(\boldsymbol{x}), h_k(\boldsymbol{x}) \in \mathbb{Q}[\boldsymbol{x}]$ を満たす。ここで、制約集合

$$\mathbb{X} = \{\boldsymbol{x} \in \mathbb{R}^n \mid g_j(\boldsymbol{x}) \leq 0 (j = 1, 2, \ldots, m), h_k(\boldsymbol{x}) = 0 (k = 1, 2, \ldots, l)\} \tag{8}$$

を制約集合の内部領域 D と境界 C とに分ける。 D, C は各々

$$D = \{\boldsymbol{x} \in \mathbb{R}^n \mid g_j(\boldsymbol{x}) < 0 (j = 1, 2, \ldots, m), h_k(\boldsymbol{x}) = 0 (k = 1, 2, \ldots, l)\} \tag{9}$$

$$C = C_1 \cup C_2 \cup \cdots \cup C_m \tag{10}$$

となる。ここで、

$$\begin{aligned} C_i \quad = \quad & \{\boldsymbol{x} \in \mathbb{R}^n | g_i(\boldsymbol{x}) = 0, g_j(\boldsymbol{x}) \leq 0 \\ & (j = 1, 2, \ldots, i-1, i+1, \ldots, m), \\ & h_k(\boldsymbol{x}) = 0 \ \ (k = 1, 2, \ldots, l)\} \end{aligned}$$

である。最適解が D 内 (制約条件により定められる領域内部) と C 内 (境界上) に存在する場合があるので、各々の場合に分けて最適解を代数的に求める。前者では、スラック変数を導入し、不等式制約条件を等式条件に変換し、目的関数とともに構成される連立代数方程式を解くことになる。連立代数方程式の解を安定に求めるためには代数的手法に頼る以外にない。簡単に示すと、グレブナ基底計算により 1 変数代数方程式に変換し、これを Sturm の定理を用いて解く。この場合に通常の辞書式順序でのグレブナ基底計算には大量の計算時間を要するが、対象とする変数のみを考慮する Shape Lemma [3] を数式処理システムにインプリメントすることによって、高速に**全次数逆辞書式順序**でグレブナ基底を求めることが出来る。一方、後者の場合は、スラック変数を含めた各変数に対して、次元が低い空間上で上と同様の操作を行う。すなわち

最適解が D に存在する場合

スラック変数 $\lambda_i \geq 0, i = 1, 2, \ldots, m$ を導入すると、数理計画問題 (1) を次のように書き直す事が出来る。

$$\begin{array}{ll} \text{目的関数} & f(\boldsymbol{x}) \quad \rightarrow \quad \text{最小} \\ \text{制約条件} & \left\{ \begin{array}{l} g_j(\boldsymbol{x}) + \lambda_j = 0 \quad (j = 1, 2, \ldots, m) \\ h_k(\boldsymbol{x}) = 0 \quad (k = 1, 2, \ldots, l) \end{array} \right. \end{array} \tag{11}$$

従って、最適解の必要条件として、

$$\left\{ \begin{array}{l} \frac{\partial f(\boldsymbol{x})}{\partial x_i} = 0 \quad (i = 1, 2, \ldots, n) \\ g_j(\boldsymbol{x}) + \lambda_j = 0 \quad (j = 1, 2, \ldots, m) \\ h_k(\boldsymbol{x}) = 0 \quad (k = 1, 2, \ldots, l) \end{array} \right. \tag{12}$$

が得られる。(12) に対して全次数逆辞書式順序のグレブナ基底、最小多項式を求め、求まった 1 変数代数方程式の解の存在する範囲を Sturm の定理によって確定する。このようにして、解 $(\boldsymbol{x}^i, \boldsymbol{\lambda}^i)$ が得られる。この中で、$\lambda_j \geq 0, j = 1, 2, \ldots, m$ を満たすものが可能解となり、その中でも、$f(\boldsymbol{x}^i)$ を最小にする $(\boldsymbol{x}^i, \boldsymbol{\lambda}^i)$ が最適解の候補になる。

最適解が C に存在する場合

この場合は、スラック変数を含めた各変数に対して、次元が低い空間上で上と同様の操作を行う。すなわち、(12) に対して、$x_1, x_2, \ldots, x_n, \lambda_1, \lambda_2, \ldots, \lambda_m$ のうち任意個の変数へ 0 を代入した後、全次数逆辞書式順序のグレブナ基底、最小多項式を求め、求まった 1 変数代数方程式の存在する範囲を Sturm の定理によっ

て確定する。解 $(\boldsymbol{x}^i, \boldsymbol{\lambda}^i)$ が得られる。この中で、$\lambda_j \geq 0, j = 1, 2, \ldots, m$ を満たすものが可能解となり、その中でも、 $f(\boldsymbol{x}^i)$ を最小にする $(\boldsymbol{x}^i, \boldsymbol{\lambda}^i)$ が最適解の候補になる。

以上の二つの場合に最適解を求め、最小になるものを選択すれば、与えられた最適化問題の解を得ることができる。制約条件や目的関数を多項式に限っているため、問題は単に連立方程式を解く問題に帰着する。多変数連立代数方程式の解を得るための安定した数値解法は十分に知られていないと言っても過言ではない。ニュートン法あるいはその拡張のような数値解法を使用して、たとえ結果を得てみても、それが必ずしも大域的な最適解を与え得ないことは自明である。しかし、この連立代数方程式を代数的に解くことによって、この困難は解消する。この解法の概略は以下のようにまとめられる。

1. 与えられた問題の多項式の係数を有理係数に変換
2. 領域 D 内の最適解をグレブナ基底の算法を用いて解く
3. 領域 C 内の最適解を求める
4. 以上の各候補の中、目的関数を最小にするものを選ぶ

現実の問題の多くは係数が実数であったり浮動小数であったりする。もちろん、浮動小数係数の代数方程式に対してグレブナ基底計算が容易であるなら、有理係数に変換する必要はない。また、グレブナ基底算法によって求められた 1 変数代数方程式は、必ずしも代数的に解くことは出来ない場合がある。そこで、 Sturm の定理を用いることを考えているが、一般的には数値解法 (DKA 法等) と結合して解く方が計算時間の面では優位になる。

次に、本論で提案した数理計画問題に対する代数的解法の適用例を示す。例題 1,2 では解法の概略を示し、例題 3 では非線形計画問題で解きにくい例として上げられる Kuhn–Tucker 条件を満たさない最適化問題を取り扱った。

例題 1

$$\text{目的関数} \quad f = -x_1 - 2x_2 + \tfrac{1}{2}x_1^2 + \tfrac{1}{2}x_2^2 \quad \rightarrow \quad \text{最小}$$

$$\text{制約条件} \quad \begin{cases} g_1 = 2x_1 + 3x_2 - 6 \leq 0 \\ g_2 = x_1 + 4x_2 - 5 \leq 0 \\ x_i \geq 0 \end{cases}$$

- 最適解が D に存在する場合

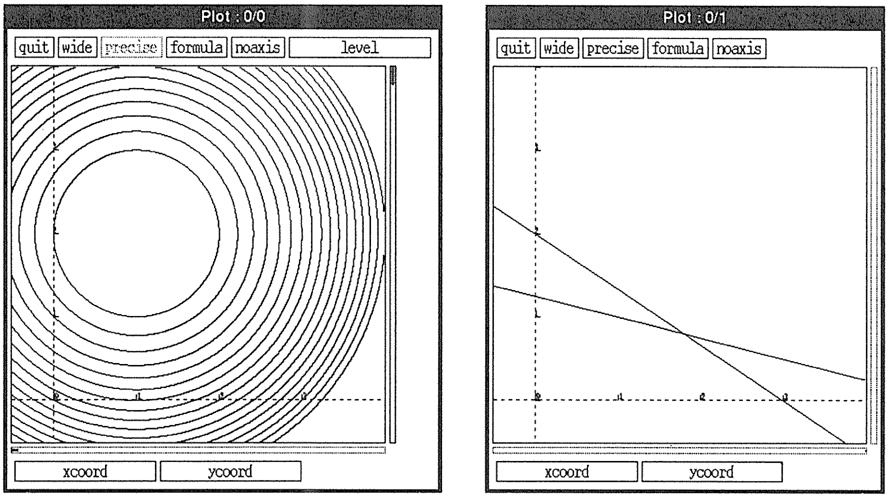

Fig. 1. 例題 1. 目的関数の等高線と制約集合

		x_1	x_2	λ_1	λ_2	f
領域内部		1	2	-2	-4	—
	$x_1 = 0$	0	2	0	-3	—
境	$x_2 = 0$	1	0	4	4	$-\frac{1}{2}$
	$\lambda_1 = 0$	$\frac{9}{13}$	$\frac{20}{13}$	0	$-\frac{24}{13}$	—
	$\lambda_2 = 0$	$\frac{13}{17}$	$\frac{18}{17}$	$\frac{22}{17}$	0	$-\frac{69}{34}$
界	$x_1 = x_2 = 0$	0	0	6	5	0
	$x_1 = \lambda_1 = 0$	0	2	0	-3	—
	$x_1 = \lambda_2 = 0$	0	$\frac{5}{4}$	$\frac{9}{4}$	0	$-\frac{55}{32}$
上	$x_2 = \lambda_1 = 0$	3	0	0	2	$\frac{3}{2}$
	$x_2 = \lambda_2 = 0$	5	0	-4	0	—
	$\lambda_1 = \lambda_2 = 0$	$\frac{9}{5}$	$\frac{4}{5}$	0	0	$-\frac{73}{50}$

Table 1. 例題 1. 最適解の候補

1. スラック変数 $\lambda_1, \lambda_2 \geq 0$ を導入

$$
\begin{array}{ll}
\text{目的関数} & f = -x_1 - 2x_2 + \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2 \quad \rightarrow \quad \text{最小} \\
\text{制約条件} & \begin{cases} g_1 + \lambda_1 = 2x_1 + 3x_2 - 6 + \lambda_1 = 0 \\ g_2 + \lambda_2 = x_1 + 4x_2 - 5 + \lambda_2 = 0 \\ x_i \geq 0 \end{cases}
\end{array}
$$

2. 最適解の必要条件

$$
\begin{cases}
\frac{\partial f}{x_1} = x_1 - 1 = 0 \\
\frac{\partial f}{x_2} = x_2 - 2 = 0 \\
g_1 + \lambda_1 = 2x_1 + 3x_2 + \lambda_1 - 6 = 0 \\
g_2 + \lambda_2 = x_1 + 4x_2 + \lambda_2 - 5 = 0
\end{cases}
$$

3. グレブナ基底

$$\{\lambda_2 + 4, \lambda_1 + 2, x_2 - 2, x_1 - 1\}$$

4. 代数的に解を求めた結果 $\rightarrow$ $x_1 = 1, x_2 = 2, \lambda_1 = -2, \lambda_2 = -4$
5. 判定 $\rightarrow$ $x_i \geq 0, i = 1, \ldots, 4$ と矛盾するので、制約集合内に解はない

- 最適解が C(境界上) に存在する場合
  1. $x_1 = 0$ の場合
     (a) スラック変数 $\lambda_1, \lambda_2 \geq 0$ を導入

$$
\begin{array}{ll}
\text{目的関数} & f = -2x_2 + \frac{1}{2}x_2^2 \quad \rightarrow \quad \text{最小} \\
\text{制約条件} & \begin{cases} g_1 + \lambda_1 = 3x_2 - 6 + \lambda_1 = 0 \\ g_2 + \lambda_2 = 4x_2 - 5 + \lambda_2 = 0 \\ x_i \geq 0 \end{cases}
\end{array}
$$

     (b) 最適解の必要条件

$$
\begin{cases}
\frac{\partial f}{x_2} = x_2 - 2 = 0 \\
g_1 + \lambda_1 = 2x_1 + 3x_2 + \lambda_1 - 6 = 0 \\
g_2 + \lambda_2 = x_1 + 4x_2 + \lambda_2 - 5 = 0
\end{cases}
$$

     (c) グレブナ基底

$$\{-\lambda_2 - 3, -\lambda_1, -x_2 + 2\}$$

     (d) 代数的に解を求めた結果 $\rightarrow$ $x_1 = 0, x_2 = 2, \lambda_1 = 0, \lambda_2 = -3$
     (e) 判定 $\rightarrow$ $x_i \geq 0, i = 1, \ldots, 4$ と矛盾するので、 $x_1 = 0$ の場合に解はない

以下、同様に

2. $x_2 = 0$ の場合

$$-\frac{1}{2} \quad (x_1 = 1, x_2 = 0, \lambda_1 = 4, \lambda_2 = 4)$$

3. $\lambda_1 = 0$ の場合 解なし

4. $\lambda_2 = 0$ の場合

$$-\frac{69}{34} \quad \left(x_1 = \frac{13}{17}, x_2 = \frac{18}{17}, \lambda_1 = \frac{22}{17}, \lambda_2 = 0\right)$$

5. $x_1 = x_2 = 0$ の場合

$$0 \quad (x_1 = 0, x_2 = 0, \lambda_1 = 6, \lambda_2 = 5)$$

6. $x_1 = \lambda_1 = 0$ の場合 解なし

7. $x_1 = \lambda_2 = 0$ の場合

$$-\frac{55}{32} \quad \left(x_1 = 0, x_2 = \frac{5}{4}, \lambda_1 = \frac{9}{4}, \lambda_2 = 0\right)$$

8. $x_2 = \lambda_1 = 0$ の場合

$$\frac{3}{2} \quad (x_1 = 3, x_2 = 0, \lambda_1 = 0, \lambda_2 = 2)$$

9. $x_2 = \lambda_2 = 0$ の場合 解なし

10. $\lambda_1 = \lambda_2 = 0$ の場合

$$-\frac{73}{50} \quad \left(x_1 = \frac{9}{5}, x_2 = \frac{4}{5}, \lambda_1 = 0, \lambda_2 = 0\right)$$

- 最小値の数値比較より最適解は

$$-\frac{69}{34} \quad \left(x_1 = \frac{13}{17}, x_2 = \frac{18}{17}, \lambda_1 = \frac{22}{17}, \lambda_2 = 0\right)$$

例題 2

$$\text{目的関数} \quad f = (x_1 - 1)x_1(x_1 - 2) + (x_2 - 1)^2 x_2 \quad \rightarrow \quad \text{最小}$$

$$\text{制約条件} \quad \begin{cases} g_1 = (x_1 - 1)(x_1 - 2)(x_1 - 3) - x_2 \leq 0 \\ g_2 = -\frac{x_1(x_1-4)(x_1-5)}{10} + x_2 \leq 0 \\ x_i \geq 0 \end{cases}$$

例題 2 の場合、例題 1 同様に解の候補を求めると、Table2 を得る。従って、最適解は

$$-0.384 \quad (x_1 = 1.577, x_2 = 1.000, \lambda_1 = 0.652, \lambda_2 = 0.307)$$

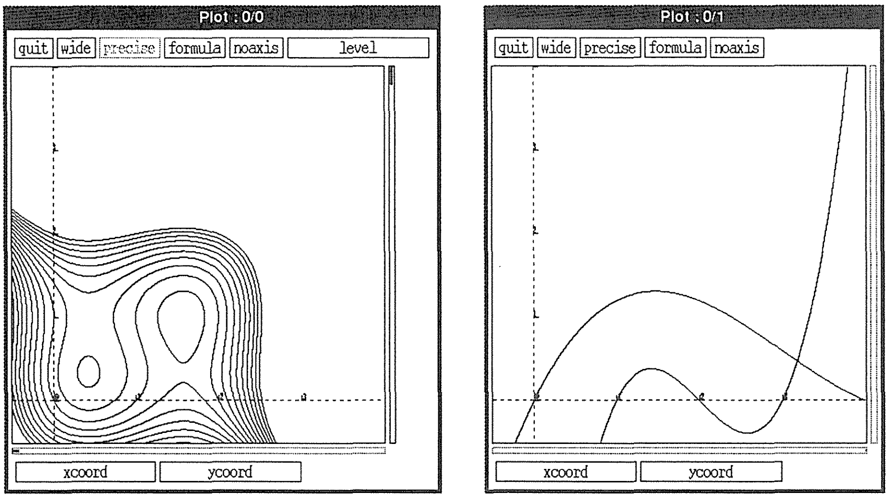

Fig. 2. 例題 2. 目的関数の等高線と制約集合

		x_1	x_2	λ_1	λ_2	f
領域内部		0.422	0.333	2.680	0.358	0.533
		1.577	1	0.652	0.307	−0.384
	$x_2 = 0$	0.422	0	2.347	0.692	0.384
	$\lambda_1 = 0$	1.059	0.109	0	1.118	0.386
境		1.573	0.348	0	0.959	−0.236
	$\lambda_2 = 0$	0.312	0.540	3.658	0	0.476
		1.606	1.304	0.972	0	−0.262
界	$x_1 = x_2 = 0$	0	0	6	0	0
	$x_1 = \lambda_2 = 0$	0	0	6	0	0
	$x_2 = \lambda_1 = 0$	1	0	0	1.2	0
上		2	0	0	1.2	0
		3	0	0	0.6	6
	$x_2 = \lambda_2 = 0$	0	0	6	0	0
	$\lambda_1 = \lambda_2 = 0$	3.183	0.472	0	0	8.352

Table 2. 例題 2. 最適解の候補

		x_1	x_2	λ	f
領域内部		2	0	−1	—
境界上	$x_1 = 0$	0	0	1	4
	$x_2 = 0$	2	0	−1	—
	$\lambda = 0$	1.651	−0.2753	0	—
	$x_1 = x_2 = 0$	0	0	1	4
	$x_1 = \lambda = 0$	0	1	0	5
	$x_2 = \lambda = 0$	1	0	0	1

Table 3. 例題 3. 最適解の候補

例題 3

$$\text{目的関数} \quad (x_1 - 2)^2 + x_2^2 \to \text{最小}$$

$$\text{制約条件} \quad \begin{cases} g_1 = -(1 - x_1)^3 + x_2 \leq 0 \\ x_1, x_2 \geq 0 \end{cases}$$

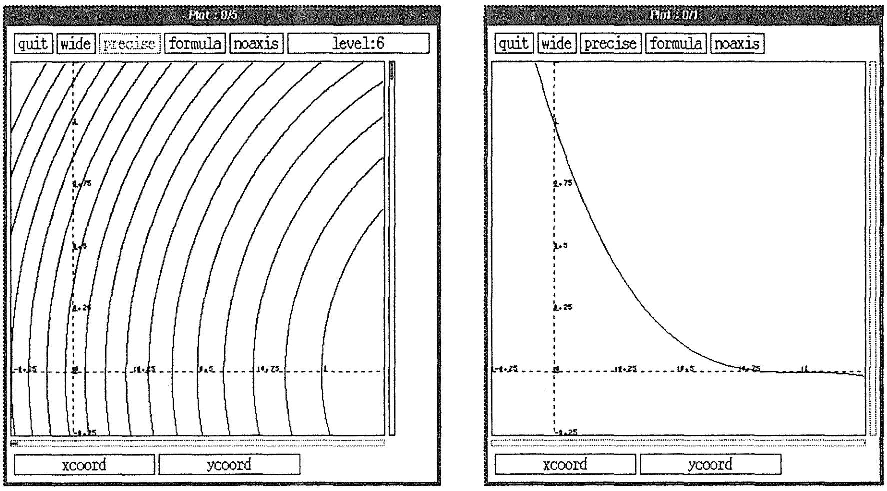

Fig. 3. 例題 3. 目的関数の等高線と制約集合

例題 3 でも同様に解の候補をを求めると、Table3 を得る。従って、最適解は

$$1 \quad (x_1 = 1, x_2 = 0)$$

4. むすび

本論では、制約条件と目的関数が多項式で表される場合の非線形計画問題の代数的解法を提案した。代数的に解くことにより、従来の解法での最適解を求められない場合を回避することが可能になり、正確な結果を得ることが期待できる。もちろん、本論に示した手法は、非線形計画問題等の数理計画問題への代数的解法の応用の可能性を示し、かついくつかの問題に有効であることの検討を行ったにとどまっている。より現実的な問題へ適用し、手法の有効性をみることは緊急の課題である。さらに、本手法の特徴を見るためにアルゴリズム面で

- 目的関数や制約条件が全て線形である場合、広く用いられているシンプレックス法 (単体法) とのアルゴリズム上の差異の明確化
- 特に 1 変数代数方程式を解く場合の各種の計算技法の比較と、それらを計画問題に用いる場合の長短所の解明
- Quantifier Elimination の立場から提唱されている諸手法 [1, 2] 等との比較検討

を行うことが必要である。また、より一般の計画問題に適用するためには、目的関数や制約条件を多項式に限定するのみでは不十分である。このような場合に簡単に考えられることは、任意の関数を十分な精度で多項式近似することである。さらに、最適解が境界 C 上にある場合に写像 ϕ をいかに求めるかも残された課題である。

参 考 文 献

[1] Weispfenning, V. : Simulation and Optimization by Quantifier Elimination, *J. Symbolic Computation*, **11**, 1996, 1–20

[2] 沢田 浩之：代数不等式制約評価アルゴリズム, 情報処理学会第 54 回 (平成 9 年前期) 全国大会, 1997, 1-253–1-254

[3] Winkler, F. :*Polynomial Algorithms in Computer Algebra*, Springer–Verlag, New York, 1996

[4] Gass, Saul I. : *LINEAR PROGRAMMING Methods and Applications Third Edition*, McGraw-Hill, 1969. (邦訳： 小山 昭雄： 線形計画法〈方法と応用〉原書第三版, 好学社, 東京, 1972.)

[5] 坂和 正敏：非線形システムの最適化〈一目的から多目的へ〉, 森北出版, 東京, 1986