

ある種の非線形計画問題の代数的解法について

白石 啓一¹⁾

Kei-ichi Shiraishi

甲斐 博¹⁾

Hiroshi Kai

齋藤 友克²⁾

Tomokatsu Saito

野田 松太郎¹⁾

Matu-Tarow Noda

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

とすることができる。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。

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

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、

¹⁾ 愛媛大学 工学部

²⁾ 上智大学 理工学部

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、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、本方法のより広い分野の非線形計画問題への提供の可能性や今後の発展方向についても言及する。

2. 代数的解法

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 $x \in \mathbb{R}^n$ に対して定義される実数値関数であり、 $f(x), g_j(x), h_k(x) \in \mathbb{Q}[x]$ を満たす。ここで、制約集合

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

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

$$D = \{x \in \mathbb{R}^n \mid g_j(x) < 0 (j = 1, 2, \dots, m), h_k(x) = 0 (k = 1, 2, \dots, l)\} \quad (3)$$

$$C = C_1 \cup C_2 \cup \dots \cup C_m \quad (4)$$

である。ここで、 $C_i = \{x \in \mathbb{R}^n \mid g_i(x) = 0, g_j(x) \leq 0 (j = 1, 2, \dots, i-1, i+1, \dots, m), h_k(x) = 0 (k = 1, 2, \dots, l)\}$ である。最適解は D 内 (制約条件により定められる領域内部) に存在する場合と 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より、最適解を求める。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 $\lambda_i \geq 0, i = 1, 2, \dots, m$ 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。

$$\begin{array}{ll} \text{目的関数} & f(x) \rightarrow \text{最小} \\ \text{制約条件} & \begin{cases} g_j(x) + \lambda_j = 0 & (j = 1, 2, \dots, m) \\ h_k(x) = 0 & (k = 1, 2, \dots, l) \end{cases} \end{array} \quad (5)$$

従って、最適解の必要条件として、

$$\begin{cases} \frac{\partial f(\mathbf{x})}{\partial x_i} = 0 & (i = 1, 2, \dots, n) \\ g_j(\mathbf{x}) + \lambda_j = 0 & (j = 1, 2, \dots, m) \\ h_k(\mathbf{x}) = 0 & (k = 1, 2, \dots, l) \end{cases} \quad (6)$$

が得られる。(6) 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 $(\mathbf{x}^i, \boldsymbol{\lambda}^i)$ が得られる。この中で、 $\lambda_j \geq 0, j = 1, 2, \dots, m$ を満たすものが可能解となり、その中でも、 $f(\mathbf{x}^i)$ を最小にする $(\mathbf{x}^i, \boldsymbol{\lambda}^i)$ 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、(6) に対して、 $x_1, x_2, \dots, x_n, \lambda_1, \lambda_2, \dots, \lambda_m$ 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。得られた解 $(\mathbf{x}^i, \boldsymbol{\lambda}^i)$ の中で、 $\lambda_j \geq 0, j = 1, 2, \dots, m$ を満たすものが可能解となり、その中でも、 $f(\mathbf{x}^i)$ を最小にする $(\mathbf{x}^i, \boldsymbol{\lambda}^i)$ 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。ここでは、連立代数方程式 $f_1(\mathbf{x}) = 0, f_2(\mathbf{x}) = 0, \dots, f_m(\mathbf{x}) = 0$ を解くために以下の方法を用いている。

1. 全次数逆辞書式順序で $f_1(\mathbf{x}), \dots, f_m(\mathbf{x})$ のグレブナ基底を求める
2. グレブナ基底より、 $x_1, x_2, \dots, x_n$ それぞれの最小多項式を求める
3. 最小多項式を代数的に解く
4. 得られた解を組合せ、 $f_1(\mathbf{x}) = 0, \dots, f_m(\mathbf{x}) = 0$ なる解を選ぶ

この中で求められる最小多項式は、必ずしも代数的に解くことは出来ない。そこで、Sturm の定理により解が存在する範囲を求めることで近似解を得て、 $|f_i(\mathbf{x})| < \epsilon$ を条件に解を選ぶ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。

本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. 以上の各候補のうち、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。

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 では解法の概略を示し、例題 2 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。これらは、不等式制約条件のみを持つ問題を以上で述べた手法により解く手続きを数式処理システム Risa/Asir 上に実現して解いた。例題 3 は一定条件の下での図形を描く問題で、CAD に応用がある。等式制約条件を持つ問題であるが、現状の手続きでも解くことができる。

例題 1

$$\begin{aligned} \text{目的関数 } f &= \frac{(x-1)(x-2)(x-3)(x-6)(x-7)(x-8)}{100} \rightarrow \text{最小} \\ \text{制約条件 } &\begin{cases} g = x - 9 \leq 0 \\ x \geq 0 \end{cases} \end{aligned}$$

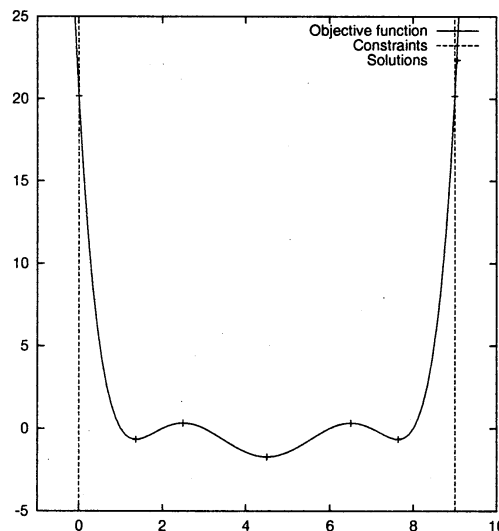


Fig. 1. 例題 1. 目的関数と制約集合

● 最適解が D に存在する場合

1. スラック変数 $\lambda \geq 0$ を導入

$$\begin{aligned} \text{目的関数 } f &= \frac{(x-1)(x-2)(x-3)(x-6)(x-7)(x-8)}{100} \rightarrow \text{最小} \\ \text{制約条件 } &\begin{cases} g + \lambda = x - 9 + \lambda = 0 \\ x \geq 0 \end{cases} \end{aligned}$$

		x	λ	f
領域内部		1.3659	7.6341	-0.65670
		2.4973	6.5027	0.32485
		4.5	4.5	-1.7227
		6.5027	2.4973	0.32485
		7.6341	1.3659	-0.65670
境	$x = 0$	0	9	20.16
界	$\lambda = 0$	9	0	20.16

Table 1. 例題 1. 最適解の候補

2. 最適解の必要条件

$$\begin{cases} \frac{\partial f}{\partial x} = \frac{6x^5 - 135x^4 + 1132x^3 - 4347x^2 + 7496x - 4572}{100} = 0 \\ g + \lambda = x - 9 + \lambda = 0 \end{cases}$$

3. グレブナ基底

$$\{x + a - 9, -6\lambda^5 + 135\lambda^4 - 1132\lambda^3 + 4347\lambda^2 - 7496\lambda + 4572\}$$

4. 解を求めた結果 $\rightarrow (x, \lambda) = (1.3659, 7.6341), (2.4973, 6.5027), (4.5, 4.5), (6.5027, 2.4973), (7.6341, 1.3659)$

5. 判定 $\rightarrow x = 4.5, \lambda = 4.5$ が最適解の候補

● 最適解が C (境界上) に存在する場合1. $x = 0$ の場合

(a) スラック変数 $\lambda \geq 0$ を導入

$$\begin{aligned} \text{目的関数 } f &= \frac{504}{25} \rightarrow \text{最小} \\ \text{制約条件 } \begin{cases} g + \lambda = -9 + \lambda = 0 \\ x_i \geq 0 \end{cases} \end{aligned}$$

(b) 最適解の必要条件

$$g + \lambda = -9 + \lambda = 0$$

(c) グレブナ基底

$$\{\lambda - 9\}$$

(d) 解を求めた結果 $\rightarrow x = 0, \lambda = 9$

(e) 判定 $\rightarrow x = 0, \lambda = 9$ が最適解の候補

以下、同様に

2. $\lambda = 0$ の場合

$$-20.16 \quad (x = 9, \lambda = 0)$$

- 最小値の数値比較より最適解は

$$-1.7227 \quad (x = 4.5, \lambda = 4.5)$$

例題 2

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

$$\text{制約条件} \quad \begin{cases} g_1 = -x_1 \leq 0 \\ g_2 = -x_2 \leq 0 \\ g_3 = -(1 - x_1)^3 + x_2 \leq 0 \end{cases}$$

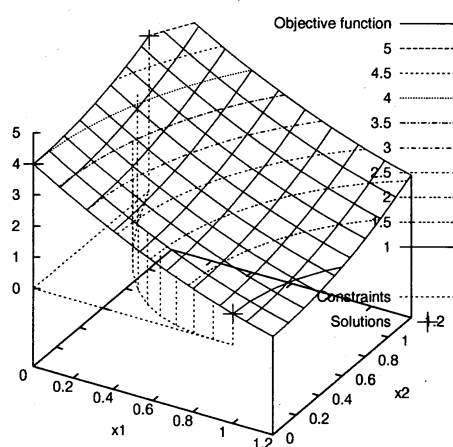


Fig. 2. 例題 2. 目的関数と制約集合

	x_1	x_2	λ	f
領域内部				
境	$x_1 = 0$	0	0	4
	$x_2 = 0$	2	-1	—
界	$\lambda = 0$	1.651	-0.2753	0
	$x_1 = x_2 = 0$	0	0	4
上	$x_1 = \lambda = 0$	0	0	5
	$x_2 = \lambda = 0$	1	0	1

Table 2. 例題 2. 最適解の候補

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

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

例題 3

楕円 $2x_1^2 + 3x_2^2 = 1$ に内接する円を描くことを考える。以下の定式化がなされる。

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

制約条件 $g = 2x_1^2 + 3x_2^2 - 1 = 0$

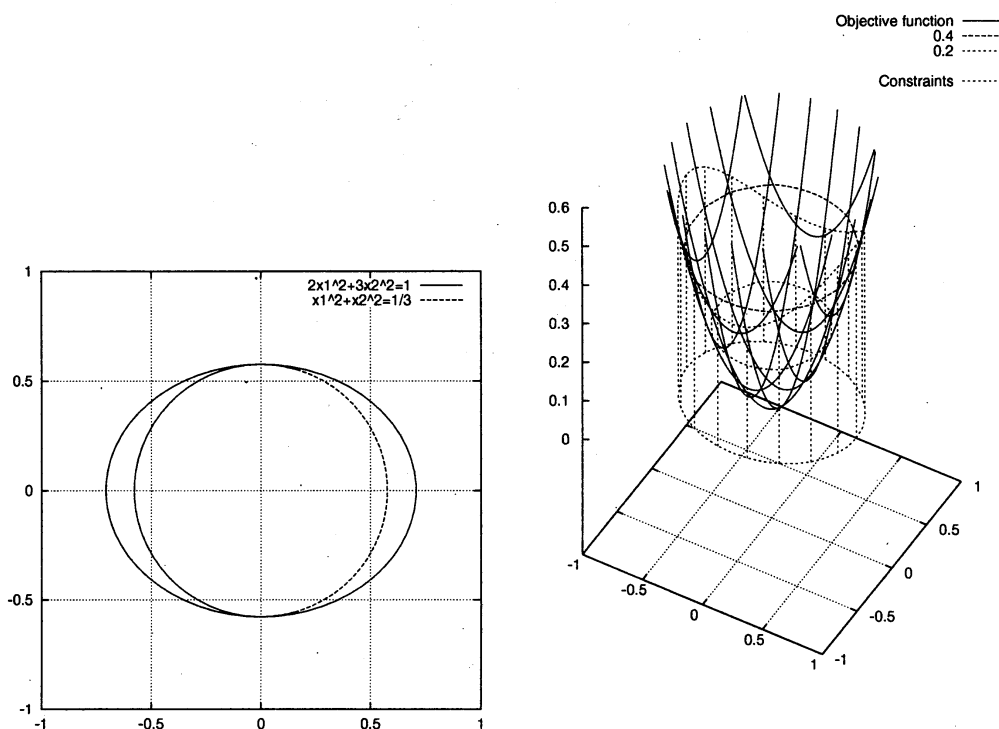


Fig. 3. 例題 3、問題の図示 (左)、目的関数と制約集合 (右)

	x_1	x_2	f
	0	0.5774	0.3333
領域内部	0 -0.5774	0.3333	
	0.7071	0	0.5
	-0.7071	0	0.5

Table 3. 例題 3. 最適解の候補

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

$$0.3333 \quad (x_1 = 0, x_2 = \pm 0.5774)$$

題意を満たす図形は

$$x_1^2 + x_2^2 = 0.3333$$

である。

3. 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、一部の問題を解く手続きを数式処理システム Risa/Asir 上に実現した。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。

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