

特殊なケースでの定式化技法

株式会社 数理システム

1. はじめに

本稿は、特殊な数理計画問題を線形計画問題（Linear Programming : LP）ないしは混合整数計画問題（Mixed Integer Programming : MIP）に置き換える為の、幾つかの代表的な手法についてまとめたものである。具体的には以下の話題を扱った。

- LP による定式化
 - 絶対値最小化問題
 - 最大値最小化問題
 - ノルム最小化問題
- MIP による定式化
 - インディケータ変数を用いた定式化
 - セパラブル・モデルを用いた非線形計画問題の近似

2. 絶対値最小化問題

本章では、絶対値最小化問題の LP による定式化について説明する。ここでは、絶対値最小化問題の例として、以下の問題を考える。

$$\begin{aligned} & \text{Minimize } \sum_{j=1}^n c_j |x_j| \quad \text{ただし, } c_j \geq 0 \quad (j = 1, \dots, n) \\ & \text{subject to } \sum_{j=1}^n a_{ij} x_j = b_i \quad (i = 1, \dots, m) \end{aligned}$$

本問題を、絶対値を用いず、LP で表現する方法を以下に説明する。

2.1. 自由変数

本節では、絶対値最小化問題の定式化の準備として、自由変数について説明する。自由変数とは、上下限値を持たない、制約の無い変数の事である。自由変数 x は、以下のような置き換えをすることが出来る。

$$\begin{aligned} x &= x^+ - x^- \\ x^+ &\geq 0, x^- \geq 0 \end{aligned}$$

ただし,

$$x \geq 0 \text{ の時 } x^+ = +x, x^- = 0$$

$$x < 0 \text{ の時 } x^+ = 0, x^- = -x$$

である. これは, x^+, x^- のどちらかが必ず 0 になる事を示している. この x^+, x^- を用いると, $|x|$ は $x^+ + x^-$ で表現できる.

2.2. 絶対値最小化問題の LP による定式化

上の絶対値最小化問題を, 2.1 節の自由変数の置き換えを用いて書き換えると以下の様になる.

$$\text{Minimize } \sum_{j=1}^n c_j (x_j^+ + x_j^-) \quad \text{ただし, } c_j \geq 0 \quad (j=1 \dots n)$$

$$\text{subject to } \sum_{j=1}^n a_{ij} (x_j^+ - x_j^-) = b_i \quad (i=1, \dots, m)$$

$$x_j^+ \geq 0, x_j^- \geq 0$$

$$x_j^+ x_j^- = 0 \quad (*)$$

ここで, (*) の制約は, x_j^+, x_j^- の少なくとも一方が必ず 0 になるというものである

が, 問題が最小化問題であることと, $c_j \geq 0 \quad (j=1, \dots, n)$ という仮定より, 省略する

ことが出来る.¹ このように, 絶対値最小化問題は LP で表現できる.

3. 最大値最小化問題

本章では, 最大値最小化問題を LP で表現する方法について説明する.

最大値最小化問題の例として, 以下の問題を考える.

$$\text{Minimize } \max\{z_1, z_2, \dots, z_k\}$$

$$\text{subject to } z_i = \sum_{j=1}^n c_j x_{ij} \quad (i=1, \dots, k)$$

これを, LP で表現する方法を以下に説明する.

3.1. 最大値最小化問題の LP による定式化

$$\text{Minimize } \max\{z_1, z_2, \dots, z_k\}$$

¹ この証明はここでは省略する. 最大化問題の場合は $c_j \leq 0 \quad (j=1, \dots, n)$ という仮定が必要.

は次の問題と等価である.

$$\begin{array}{ll} \text{Minimize} & z \\ \text{subject} & z \geq z_i \quad (i=1, \dots, k) \end{array}$$

よって, 最大値最小化問題を LP で表現すると以下のようなになる.

$$\begin{array}{ll} \text{Minimize} & z \\ \text{subject to} & z - \sum_{j=1}^n c_j x_{i,j} \geq 0 \end{array}$$

4. ノルム最小化問題

この章では, ノルム最小化問題を LP で表現する方法について説明する. 以下に標準的なノルムの定義をあげる.

$\mathbf{z} = (z_1, z_2, \dots, z_m)$ の時,

- 1-ノルム

$$\|\mathbf{z}\|_1 = \sum_{i=1}^m |z_i|$$

- 2-ノルム

$$\|\mathbf{z}\|_2 = \left(\sum_{i=1}^m z_i^2 \right)^{\frac{1}{2}}$$

- ∞ -ノルム

$$\|\mathbf{z}\|_\infty = \max_{1 \leq i \leq m} |z_i|$$

2章,3章で述べた2つの方法を応用すれば, 1-ノルム, ∞ -ノルムの最小化を LP で定式化することが出来る. ただし, 2-ノルムに関しては, LP で表現することは出来ない.

4.1. 1-ノルム最小化問題

この節では, 上で述べた1-ノルムの最小化問題, すなわち

$$\text{Minimize} \quad \sum_{i=1}^m |z_i|$$

を LP として, 表現する方法を以下に述べる.

これは, 2.1節で説明した絶対値の定式化を用い,

$$v_i^+ - v_i^- = z_i$$

$$v_i^+ \geq 0, v_i^- \geq 0$$

のように、絶対値の中身を置き換えればよい。よって、1-ノルム最小化の定式化は、以下の様になる。

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^m (v_i^+ + v_i^-) \\ \text{subject to} \quad & v_i^+ - v_i^- = z_i \quad (i = 1, \dots, m) \\ & v_i^+ \geq 0, v_i^- \geq 0 \quad (i = 1, \dots, m) \end{aligned}$$

4.2. ∞ -ノルム最小化問題

この節では、 ∞ -ノルムの最小化問題、すなわち

$$\text{Minimize} \quad \|\mathbf{z}\|_\infty = \max_{1 \leq i \leq m} |z_i|$$

を LP として、表現する方法を以下に述べる。

この問題も 4.1 節同様、絶対値の中身を、

$$v_i^+ - v_i^- = z_i$$

$$v_i^+ \geq 0, v_i^- \geq 0$$

のように、置き換える。

さらに、最大値最小化問題であるため、新しい変数 q を導入し

$$\begin{aligned} \text{Minimize} \quad & q \\ \text{subject to} \quad & q \geq v_i^+ + v_i^- \quad (i = 1, \dots, m) \\ & v_i^+ \geq 0, v_i^- \geq 0 \quad (i = 1, \dots, m) \end{aligned}$$

と置き換えられる。

5. インディケータ変数を含む問題

インディケータ変数とは、状態の on/off を表す変数である。これを用いることによって、以下に挙げる様な事例を混合整数計画問題 (Mixed Integer Programming Problem : MILP) で表現することが出来る。

5.1. 固定値の表現

この節では、以下のような問題を考える。

ある製品を x 単位生産するものとする、このとき、1 単位あたりの生産に要する費用は C_1 である。これに加え、もし、その製品が少しでも生産されるならば、立ち上げの為の初期費用 C_2 も発生する。

このことは次式で表現される。

$$x = 0 \text{ の時 } \quad \text{総費用} = 0$$

$$x > 0 \text{ の時 } \quad \text{総費用} = C_1x + C_2$$

これを表現するために、インディケータ変数 δ を導入した以下の式を考える。

$$\text{総費用} = C_1x + C_2\delta$$

$$0 \leq x \leq M \cdot \delta \quad (M \text{ は } x \text{ が取り得ることない、十分大きなパラメータ}) \quad (*)$$

この時(*)によって、 $\delta = 0$ の時には、 x の値は 0 に固定され、総費用 = 0 が表現出来る。また、 $\delta = 1$ の時には、 x は 0 ではない値を取ることができ、総費用 = $C_1x + C_2$ を表現することが出来る。

5.2. 不等式制約の切り替え

インディケータ変数 δ を用いると、不等式制約として $g(x) \geq 0$ と $g(x) < 0$ のどちらを用いるかを切り替えることが出来る。その為には以下の 2 つの式を連立させればよい。

$$g(x) \geq -(1 - \delta)M \quad (M : g(x) \text{ の絶対値より十分大きなパラメータ})$$

$$-g(x) \geq \varepsilon - \delta M \quad (\varepsilon : g(x) \text{ が } 0 \text{ とみなせる十分小さなパラメータ})$$

この時、 $\delta = 1$ ならば、 $g(x) \geq 0$ 、 $-g(x) \geq \varepsilon - M$ である。これらをまとめると、 $0 \leq g(x) \leq M - \varepsilon$ である。これは事実上、 $0 \leq g(x)$ という制約が適用される事を示している。また $\delta = 0$ ならば、 $g(x) \geq -M$ 、 $-g(x) \geq \varepsilon$ である。これはすなわち、 $-M \leq g(x) \leq -\varepsilon$ である。これは事実上、 $g(x) < 0$ という制約が適用される事を示している。

5.3. インディケータ変数の利用

インディケータ変数 δ を用いると、変数の値を固定したり、関数を差し替えることが出来る。

例えば δ が $0[1]$ の時、変数を 0 に固定するには、

$$0 \leq x \leq \delta \quad [0 \leq x \leq 1 - \delta]$$

とすればよい。

また、5.2 節の内容を用いると、 δ の値によって関数 f を差し替える事が出来る。その為には、十分大きなパラメータ M を用いて、

$$-M \cdot (1 - \delta) \leq f - a \cdot x \leq M(1 - \delta) \quad (1)$$

$$-M \cdot \delta \leq f - b \cdot x \leq M \cdot \delta \quad (2)$$

とすればよい。

この時、 $\delta = 1$ ならば、 f は (1) により $a \cdot x$ に固定され、(2) は事実上有効ではない。また $\delta = 0$ ならば、 f は (2) により $b \cdot x$ に固定され、(1) の制約は効いていない事がわかる。

5.4. 起動・停止コストの表現

この節では、機器が起動または停止する際にかかるコストを最小化の問題を考える。これもまた、インディケータ変数 δ で表現することが出来る。この表現方法を以下に説明する。

時系列の運転インディケータ変数 δ_t ($t=1, \dots, T$) と、起動・停止インディケータ γ_t, η_t ($t=1, \dots, T-1$) を導入する。 δ_t は機器が運転している時に 1、停止している時に 0 となるものとする。また γ_t は機器が運転を開始した時刻のみに 1 となり、残りの時刻では 0 となるものとする。同様に、 η_t は機器が停止する時刻のみに 1 をとるものとする。これらの一例を次の表に挙げる。

t	1	2	3	4	...	21	22	23	24
δ_t	0	0	1	1		1	1	0	0
γ_t	0	0	1	0		0	0	0	-
η_t	0	0	0	0		0	1	0	-

この時、起動・停止にかかるコストは

$$\sum_t (C \cdot \gamma_t + D \cdot \eta_t) \quad (※)$$

で表現できる。

さらに、 δ_t と γ_t, η_t の関係は、以下の 2 つの式を連立させる事によって表現できる。

$$\delta_t - \delta_{t-1} \leq \gamma_t \quad (\text{起動})$$

$$\delta_t - \delta_{t+1} \leq \eta_t \quad (\text{停止})$$

この式の左辺は、運転状態の変化を示している。すなわち、運転状態が変化すると 1 であり、変化がないと 0 である。これらの式だけでは、運転状態の変化がない時刻にも γ_t, η_t が 1 となる可能性があるが、 γ_t, η_t は最小化される目的関数 (※) に現れる事に注意すると、起動・停止時のみ 1、その他の時には 0 であることが言える。

5.5. 論理条件の表現

インディケータ変数を用いると、状態間の論理的な結びつけが可能となる。例として、論理和と論理積について表現すると、

$$\text{論理和} : \delta \leq \delta_1 + \delta_2 \leq 2\delta$$

$$\text{論理積} : 2 \cdot \delta \leq \delta_1 + \delta_2 \leq 1 + \delta$$

となる。

5.6. 折れ線関数の表現

インディケータ変数を用いると、折れ線関数を表現することも出来る。

折れ線関数 $f(x)$ が連続で、 x の定義域がその中において $f(x)$ が線形関数とみなせる n 個の区間に分割される場合を考える。この時 $f(x)$ は次のように表される。

$$y = f(x) = \sum_{i=1}^n s_i x_i + y_0$$

$$x = \sum_{i=1}^n x_i$$

$$0 \leq x_i \leq L_i \quad (i=1, \dots, n)$$

$$L_i z_i \leq x_i \leq L_i z_{i-1} \quad (i=1, \dots, n)$$

ここで、 z_i ($i=1, \dots, n$) はインディケータ変数であり、 s_i ($i=1, \dots, n$) は各折れ線の傾き、 y_0 は $x=0$ の時の y の値、 L_i ($i=1, \dots, n$) は各区間の幅である。

ただし、 z_0, z_n は $z_0=1, z_n=0$ と固定しても一般性を失わない。よって、必要なインディケータ変数の実際の総数は $n-1$ 個である。

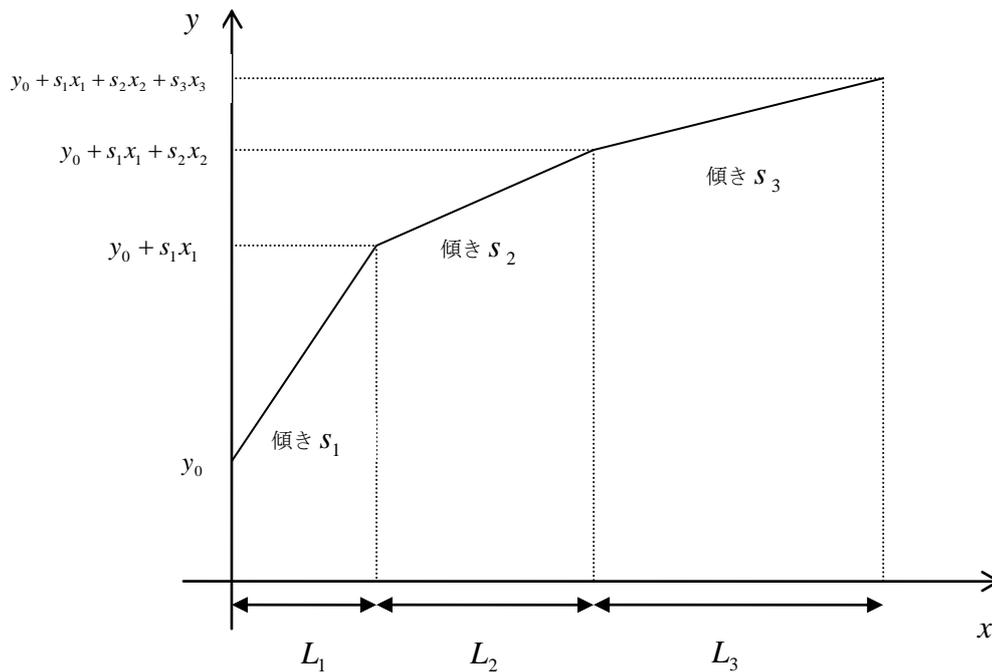


図 1 折れ線関数の例

具体的に、3 つの区間に分割されている、折れ線関数を定式化してみる。

$$y = s_1 x_1 + s_2 x_2 + s_3 x_3 + y_0$$

$$x = x_1 + x_2 + x_3$$

$$\begin{aligned}
0 &\leq x_1 \leq L_1 \\
0 &\leq x_2 \leq L_2 \\
0 &\leq x_3 \leq L_3 \\
L_1 z_1 &\leq x_1 \leq L_1 & (L_1 z_1 \leq x_1 \leq L_1 z_0) \\
L_2 z_2 &\leq x_2 \leq L_2 z_1 \\
0 &\leq x_3 \leq L_3 z_2 & (L_3 z_3 \leq x_3 \leq L_3 z_2)
\end{aligned}$$

本定式化により，インディケータ変数 z_1, z_2 の組み合わせで，各場合を表現できている事になる．以下，その事を確認してみる．

- $z_1 = 0, z_2 = 0$ の時

$$\begin{aligned}
0 &\leq x_1 \leq L_1 \\
0 &\leq x_2 \leq 0 \\
0 &\leq x_3 \leq 0
\end{aligned}$$

この時， x_2, x_3 は 0 に固定され， x_1 により L_1 の区間が表現されている．

- $z_1 = 1, z_2 = 0$ の時

$$\begin{aligned}
L_1 &\leq x_1 \leq L_1 \\
0 &\leq x_2 \leq L_2 \\
0 &\leq x_3 \leq 0
\end{aligned}$$

この時， x_1, x_3 は $x_1 = L_1, x_3 = 0$ に固定され， x_2 により， L_2 の区間が表現されている．

- $z_1 = 1, z_2 = 1$ の時

$$\begin{aligned}
L_1 &\leq x_1 \leq L_1 \\
L_2 &\leq x_2 \leq L_2 \\
0 &\leq x_3 \leq L_3
\end{aligned}$$

この時， x_1, x_2 は $x_1 = L_1, x_2 = L_2$ に固定され， x_3 により L_3 の区間が表現されている．

- $z_1 = 0, z_2 = 1$ の時

$L_2 z_2 \leq x_2 \leq L_2 z_1$ という制約があるため，選ばれることは無い．

6. セパラブル・モデル

最後にセパラブル・モデルの MIP による近似について説明する。

セパラブル・モデルについて説明するため、まずセパラブルな関数について説明する。セパラブルな関数とは、1 変数関数の和として表される関数の事である。つまり、1 つの項の中に含まれる変数の数は 1 つである。²

例) $x_1^2 + 2x_2 + e^{x_3}$

セパラブルな関数は、以下の性質を持つ。

- インディケータ変数を用いて折れ線近似が出来る
- 最小化問題で凸な問題に対して、折れ線近似した問題において、大域最適解を得ることが出来る

(ただし、非凸な問題に対しては、局所的最適解が得られる)

つまり、セパラブル・モデルとは、上で述べたセパラブルな関数をもつ、数理計画モデルの事である。

6.1. セパラブル・モデルの MIP による定式化

セパラブル・モデルを MIP として定式化する為には、セパラブルな関数の非線形項を線形近似する必要がある。ここでは、例として二次の非線形項がある、凸計画問題のセパラブル・モデルの定式化を以下に示す。

例) Minimize $x_1^2 - 4x_1 - 2x_2$

subject to $x_1 + x_2 \leq 4$
 $2x_1 + x_2 \leq 5$
 $-x_1 + 4x_2 \geq 2$
 $x_1, x_2 \geq 0$

² 一般には、これとは違った概念で「セパラブル」という言葉を定義する事もあるが、ここではその定義は割愛する。

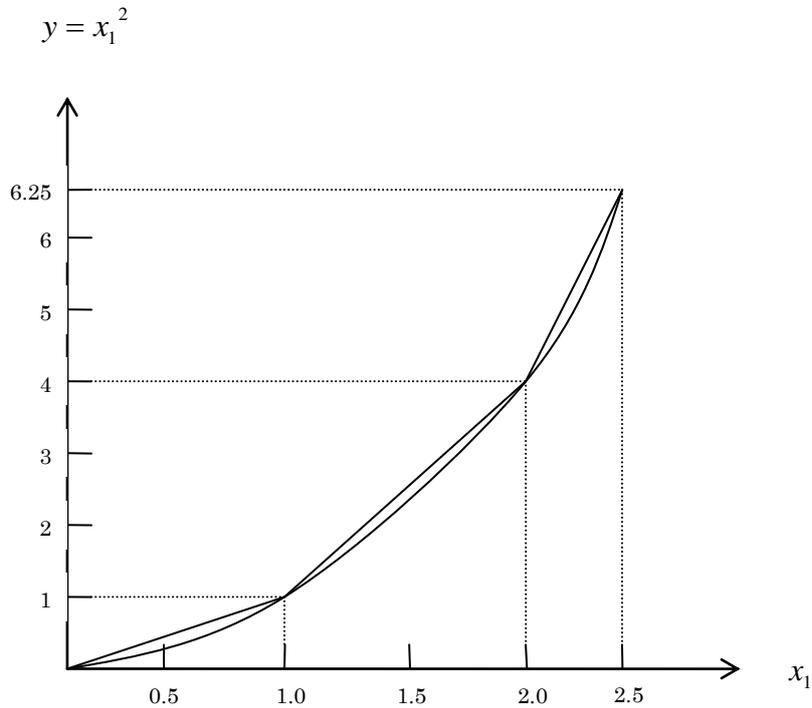


図 2 折れ線による近似の例

上の例で、非線形項 x_1^2 を取り除くことを考える．ここでは区間 $0 \leq x_1 \leq 2.5$ を $0 \leq x_1 \leq 1$ ， $1 \leq x_1 \leq 2$ ， $2 \leq x_1 \leq 2.5$ の 3 つの区間に分割し，図 2 のように折線近似を行う．

$$\begin{aligned} x_1 &= 0\lambda_1 + 1\lambda_2 + 2\lambda_3 + 2.5\lambda_4 \\ y &= 0\lambda_1 + 1\lambda_2 + 4\lambda_3 + 6.25\lambda_4 \\ \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 &= 1 \\ \lambda_i &\geq 0 \quad (i=1,\dots,4) \end{aligned}$$

ただし， λ_i はただか 2 つの隣り合った λ_i のみがゼロではない．(※1)

(※1) は y ， x_1 が折れ線上の値を取る事を保証するものである．この制約を定式化するには，インディケータ変数 δ_i ($i=1,\dots,3$) を導入して，

$$\begin{aligned} \lambda_1 - \delta_1 &\leq 0 \\ \lambda_2 - \delta_1 - \delta_2 &\leq 0 \\ \lambda_3 - \delta_2 - \delta_3 &\leq 0 \\ \lambda_4 - \delta_3 &\leq 0 \\ \delta_1 + \delta_2 + \delta_3 &= 1 \end{aligned} \quad \dots(\text{※2})$$

とすればよい.

上記を踏まえ, 上の例のセパラブル・モデルを MIP として定式化すると, 次のようになる.

$$\begin{aligned} \text{Minimize } & y - 4x_1 - 2x_2 \\ \text{subject to } & x_1 + 2x_2 \leq 4 \\ & 2x_1 + x_2 \leq 5 \\ & -x_1 + 4x_2 \geq 2 \qquad \dots(\ast 3) \\ & -x_1 + \lambda_2 + 2\lambda_3 + 2.5\lambda_4 = 0 \\ & -y + \lambda_2 + 4\lambda_3 + 6.25\lambda_4 = 0 \\ & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1 \\ & y, x_1, x_2, \lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0 \\ & \text{条件 } (\ast 2) \end{aligned}$$

さらに, 今回の問題は次の特徴を持つ.

- y は最小化される目的関数に含まれている
- y は凸関数である

これらの特徴により, 問題(\ast 3)から条件(\ast 2)をとり除く事が出来る. 従って問題は LP として表現できる.

本定式化には, 折れ線近似を用いているので, ある程度の不正確さは伴う. しかし, 不正確性は近似する直線の数を増やすことによって減らす事が出来る. また, 不正確性を取り除く事が非常に重要であると考えられる場合には, 本定式化で得られた, 最適解付近の分割を細かくし, 再度最適化を行うという方法もある.

6.2. 凸関数の線形近似の一般化

6.1.で示した凸関数の線形近似を一般化すると, 次のようになる.

非線形項 $y = f(x)$ を区間 i に分割する時, 新しい変数 λ_i を導入し,

$$y = \lambda_i f(x_i)$$

$$x = \sum_i \lambda_i x_i$$

$$\sum_i \lambda_i = 1$$

のように, 定式化できる.

6.3. セパラブルな関数への変換

セパラブル・モデルは, セパラブルな関数だけに適用可能であるが, セパラブルでな

い関数をセパラブルな関数に組み替える事が可能な場合がある。現実のモデルのセパ
ラブルでない関数は、2つ、もしくはそれ以上の変数の積である事が多い。この変換
の例を以下に示す。

例) x_1x_2 という項をセパラブルな形に変換する

u_1, u_2 という2つの新しい変数を以下の様に定める。

$$u_1 = (x_1 + x_2)/2$$

$$u_2 = (x_1 - x_2)/2$$

この時,

$$x_1x_2 = u_1^2 - u_2^2$$

となるので、 x_1x_2 をセパラブルな形に変換する事が出来た。

参考文献

- [1] H.P.ウイリアムス (前田栄次郎 監訳, 小林栄三 訳) 数理計画モデルの作成法,
産業図書, 1995
- [2] Gerge B.Dantzig, Mukund N. Thapa Linear Programming 1:Introduction
Springer-Verlag, 1997