研究レポート

マックスミニ型効用関数をもつプロジェクト選択問題に対する近似解法

福川忠昭・山口俊和・奈良雅子

1. 研究のねらい

企業体の経営意思決定問題の1つに，複数の資源制約のもとで複数の目標をバランスよく増大させるようにプロジェクトを選択する問題がある．この種の問題は決定変数が0-1型の整数計画問題として定式化されることが多い．多目標0-1計画問題に対する最適化法として[1][2]などがあるが，プロジェクト（変数）の数が多くなるにつれて，プロジェクトの可能な組合せの数は加速度的に増大し，計算量は膨大になる．いっぽう，投資計画などの現実問題では，問題を構成するデータに各種の測定誤差が含まれることが多いので，厳密な計算を行なって最適解を求めるよりも，簡便な計算で実用上十分な精度の解が得られるような近似解法が必要である．

0-1型の整数計画問題に対する近似解法としては単一目標複数制約の場合は空間法[3]が開発されている．この考え方は多目標の場合にも拡張されているが[4]複数目標のバランスよい達成をめざすという考え方にもとづく定式化は行なわれておらず，別の考え方が必要である．

本研究は，複数の目標達成値のそれぞれをバランスよく増大させることを目的とした0-1型の多目標多制約問題の定式化を行ない，その近似解を求めるための1つの方法を提案するものである．変数が0-1型で，複数目標のバランスよい達成をめざすタイプの代表例としては，資本予算配分問題があげられ[6]，最悪の目標達成値の最大化を行なうマックスミニ型効用関数が適していることが知られている[5]．そこで，本研究では0-1型の多目標多制約問題をマックスミニ型効用関数を用いて定式化し，近似解を求めるために空間法の考え方を応用したアルゴリズムを作成し数値実験を行ない，近似解の性質を多角的に検討する．

ふくかわ　ただあき　慶応義塾大学，やまぐち　としかず　東京理科大学，なら　まさこ　キヤノン㈱

2. 問題の定式化

互いに独立なm個のプロジェクトの中から，q個の資源制約のもとでr個の目標達成値のそれぞれをバランスよく増大させるようなプロジェクトの組合せを決定する問題を考える．定式化のため次のように記号を定義する．

i：プロジェクト名($i=1,2,\cdots,m$)

k：制約名($k=1,2,\cdots,q$)

j：目標名($j=1,2,\cdots,r$)

l_{ki}：プロジェクトiの制約kの使用量．そのベクトル表現を

$$\boldsymbol{L}_i=(l_{1i},l_{2i},\cdots,l_{qi})\geq\boldsymbol{0}$$

g_{ji}：プロジェクトiの目標jの達成値．そのベクトル表現を

$$\boldsymbol{G}_i=(g_{1i},g_{2i},\cdots,g_{ri})\geq\boldsymbol{0}$$

$\bar{l}_k$：制約kの使用可能上限値

$\boldsymbol{W}=(W_1,W_2,\cdots,W_r)\geq\boldsymbol{0}$：目標を増加させるのに望ましい方向を示すベクトル（目標ベクトル）

$\boldsymbol{w}=(w_1,w_2,\cdots,w_r)\geq\boldsymbol{0}$：$\boldsymbol{W}$上の単位ベクトル

x_i：決定変数（プロジェクトiを採択するとき1，採択しないとき0の値をとる）

以上の記号を用いて上述の問題を次のようなマックスミニ型効用関数をもつ0-1計画問題に定式化する．

目的関数：最大化 $\min_{j=1,2,\cdots,r}\{(\sum_{i=1}^{m}g_{ji}x_i)/w_j\}$

制約条件：$\sum_{i=1}^{m}l_{ki}x_i\leq\bar{l}_k\ (k=1,2,\cdots,q)$

$x_i=0$ または $1\ (i=1,2,\cdots,m)$

3. 解法の基本的な考え方

本研究で提案する近似解法の基本的な考え方は，各プロジェクトについて，(1) 複数の制約使用量を空間量に一元化し，(2) 複数の目標達成値を目標ベクトル上の長さに一元化し，(3) 目標の一元化指標を制約の一元化指標で割った効率指標を作成し，(4) それにもとづいてプ

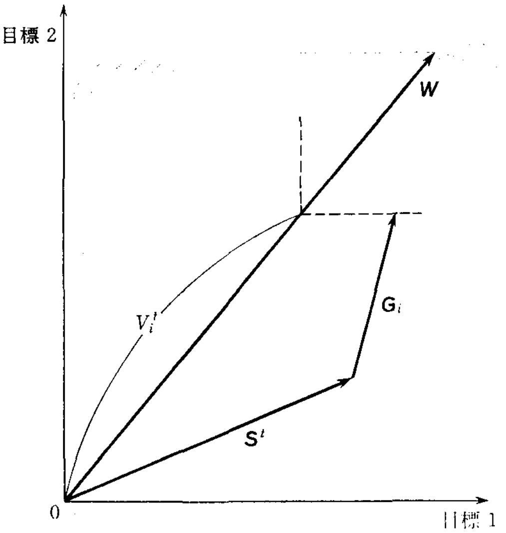

図 1　前進法における目標の一元化

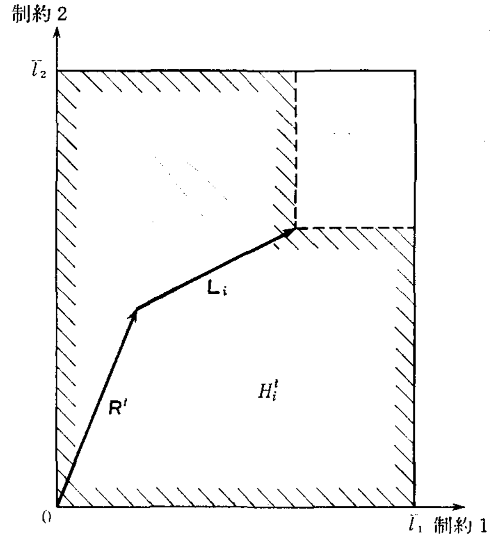

図 2　前進法における制約の一元化

ロジェクトに1つずつ順位をつけ制約条件を考慮して採択(または棄却)していく，というものである．効率のよいものから順次採択を決定していく方法を前進法，効率の悪いものから不採択を決定していく方法を後退法と呼ぶ．ただし，プロジェクトを1つ採択または棄却するごとに残ったプロジェクトの効率は計算し直す．

一元化指標の計算式を定義するために，次のような記号を定めておく．ここで，添字 t は逐次的にプロジェクトを採択(または棄却)していくときの解法の反復回数を示す．

I^t：既採択プロジェクトの集合

$\bar{I}^t$：未採択プロジェクトの集合

$\tilde{F}^t$：採択対象プロジェクトの集合

F^t：採択可能プロジェクトの集合

M^t：制約条件を満たしている制約の添字集合

$\bar{M}^t$：制約条件を満たしていない制約の添字集合

$$S^t=(S_1{}^t, S_2{}^t, \cdots, S_r{}^t)=\sum_{i\in I^t} G_i$$

$$R^t=(R_1{}^t, R_2{}^t, \cdots, R_q{}^t)=\sum_{i\in I^t} L_i$$

前進法の場合の目標の一元化指標 $V_i{}^t$，制約の一元化指標 $H_i{}^t$ をそれぞれ次のように定義する．

$$V_i{}^t=\min_j \{(S_j{}^t+g_{ji})/w_i\} \tag{1}$$

$$H_i{}^t=\prod_{k=1}^{q}\bar{l}_k-\prod_{k=1}^{q}\{\bar{l}_k-(R_k{}^t+l_{ki})\} \tag{2}$$

2目標2制約の場合で図解すると図1，図2のようになる．$V_i{}^t$ は目標ベクトル上の長さに対応し，$H_i{}^t$ は制約平面上の面積に対応する．

後退法の場合の目標の一元化指標 $V_i{}^t$，制約の一元化指標 $H_i{}^t$ を次のように定義する．

$$V_i{}^t=\min_j \{(\sum_{s=1}^{m} g_{js})/w_j\}-\min_j \{(S_j{}^t-g_{ji})/w_j\} \tag{3}$$

$$H_i{}^t=\prod_{k\in\bar{M}^t} \{\sum_{s=1}^{m} l_{ks}-(R_k{}^t-l_{ki})\} \tag{4}$$

2目標2制約の場合について図解すると図3，図4のようになる．

4.　解法の手順

m プロジェクト(変数)・q 制約・r 目標の問題に対する近似解法の手順を以下に示す．

〈前進法〉

(1) 制約の基準化

$l'_{ki}=l_{ki}/\bar{l}_k, \bar{l}_k'=1 \quad (\forall k)$

(2) 初期値の設定

$t=1, I^1=\phi, \tilde{F}^1=\{1, 2, \cdots, m\}$

(3) I^t の制約使用量と目標達成値の計算

$R_k{}^t=\sum_{i\in I^t} l'_{ki}(\forall k),\ S_j{}^t=\sum_{i\in I^t} g_{ji}(\forall j)$

(4) 選択可否の判定

$F^t=\tilde{F}^t \backslash \{i\in\tilde{F}^t \mid R_k{}^t+l'_{ki}>\bar{l}'_k, \exists k\}$

とし，$F^t=\phi$ のとき(8)へ．

(5) 効率指標の計算

$i\in F^t$ について，(1)式と(2)式により $V_i{}^t, H_i{}^t$ を計算し，次の効率指標を求める．

$U_i{}^t=V_i{}^t/H_i{}^t$

(6) 選択プロジェクトの決定

$U^t{}_{\max}=\max_{i\in F^t} U_i{}^t$

となる i を第 t 回目の選択プロジェクト $i^t{}_*$ とし，

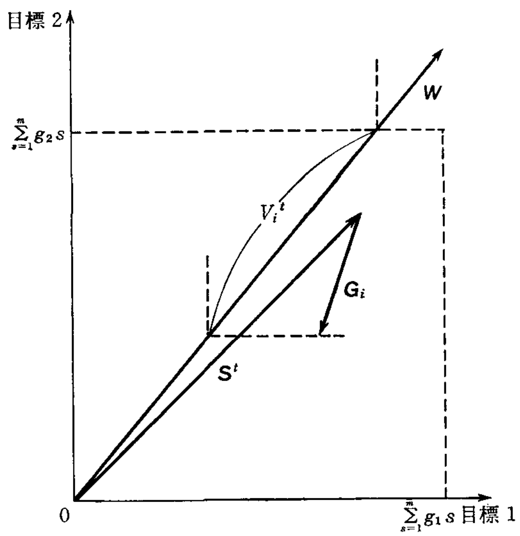

図 3　後退法における目標の一元化

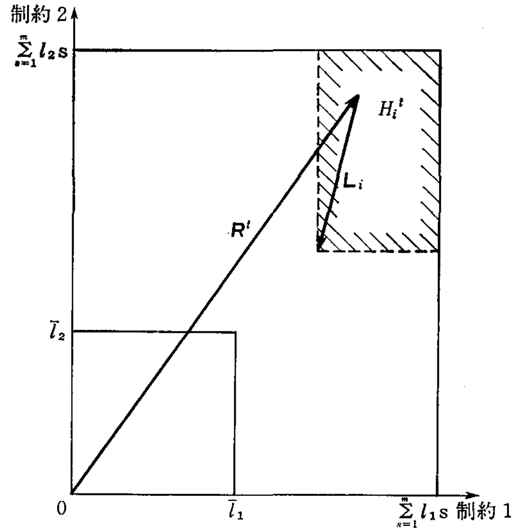

図 4　後退法における制約の一元化

$I^{t+1}=I^t \cup \{i^t_*\}$, $F^{t+1}=\tilde{F}^t \backslash \{i^t_*\}$

とする．

(7) 反復 $t=t+1$ として(3)へ戻る．

(8) 最終ステップへの後退

$t=t-1$ とし，$i \in F^t$ について(1)式の V_i^t の値を求め

$$V^t_{\max}=\max_{i \in F^t} V_i^t$$

となる i をあらためて第 t 回目の選択プロジェクト i^t_* とする．

(9) 手順終了

$I^{t+1}=I^t \cup \{i^t_*\}$

とし，I^{t+1} をこの問題の解として手順を終了する．

〈後退法〉

(1) 制約の基準化

$l'_{ki}=l_{ki}/\bar{l}_k$, $\bar{l}_k'=1(\forall k)$

(2) 初期値の設定

$t=1, I^1=\{1,2,\cdots,m\}$, $\bar{I}^1=\phi$

$M^0=\phi, \bar{M}^0=\{1,2,\cdots,q\}$

(3) I^t の制約使用量と目標達成値の計算

$$R_k^t=\sum_{i \in I^t} l'_{ki}(\forall k \in \bar{M}^{t-1}), S_j^t=\sum_{i \in I^t} g_{ji}(\forall j)$$

(4) 制約条件の満足の判断

$M^t=M^{t-1} \cup \{k \in \bar{M}^{t-1} | R_k^t \leq \bar{l}'_k\}$

$\bar{M}^t=\{1,2,\cdots,q\} \backslash M^t$

とし，$\bar{M}^t=\phi$ のとき(8)へ．

(5) 効率指標の計算

$i \in I^t$ について，(3)式と(4)式より V_i^t, H_i^t を計算し，次の効率指標を求める．

$U_i^t=V_i^t/H_i^t$

(6) 棄却プロジェクトの決定

$$U^t_{\min}=\min_{i \in I^t} U_i^t$$

となる i を第 t 回目の棄却プロジェクト i^t_* とし，

$I^{t+1}=I^t \backslash \{i^t_*\}, \bar{I}^{t+1}=\bar{I}^t \cup \{i^t_*\}$

とする．

(7) 反復 $t=t+1$ とし(3)へ戻る．

(8) 再選択の可能性の判定

$\tilde{F}^t=\bar{I}^t, F^t=\tilde{F}^t \backslash \{i \in \tilde{F}^t | R_k^t+l'_{ki}>\bar{l}'_k, \exists k\}$

とし，$F^t=\phi$ のとき，I^t をこの問題の解として手順を終了する．そうでない場合は(9)へ．

(9) 再選択プロジェクトの決定

$i \in F^t$ について，(1)式の V_i^t の値を計算し，

$$V^t_{\max}=\max_{i \in F^t} V_i^t$$

となる i を再選択プロジェクト i^t_* とする．

(10) 更新と反復

$I^{t+1}=I^t \cup \{i^t_*\}, \bar{I}^{t+1}=\bar{I}^t \backslash \{i^t_*\}$

とし，$t=t+1$ としてすべての k に対して制約使用量 R_k^t を計算し(8)へ戻る．

5.　近似解の評価

前進法および後退法によって得られる近似解の性質を調べるために，次のような数値実験を行なった．

(1) l_{ki}, g_{ji} の値を 0～99 の一様乱数によって発生させる．

(2) 制約のきつさ p(プロジェクト全体が必要とする総資源に比して利用可能な資源が少ないか否かを示す係数)を 3 通り($p=0.3, 0.5, 0.7$)に設定し，それによって各制約の上限 $\bar{l}_k$ を定める．すなわち

表 1　前進法による平均誤差率 (%)

	m	10			20	
p	r ＼ q	2	5	10	2	5
0.3	2	4.3	5.8	7.3	4.5	9.6
	5	6.7	14.4	10.2	7.0	14.6
	10	11.2	17.3	15.0	9.0	—
0.5	2	3.0	5.5	5.9	2.7	6.6
	5	4.5	8.6	10.3	4.2	8.9
	10	6.9	13.9	12.5	5.4	—
0.7	2	2.1	3.0	4.1	1.6	3.4
	5	3.4	5.4	8.2	2.2	4.9
	10	5.1	8.3	9.0	3.4	—

表 2　後退法による平均誤差率 (%)

	m	10			20	
p	r ＼ q	2	5	10	2	5
0.3	2	5.2	13.2	21.7	3.0	10.6
	5	7.0	13.4	20.4	4.7	12.2
	10	10.7	17.8	27.2	7.0	—
0.5	2	2.8	6.7	13.2	2.1	6.5
	5	4.1	8.8	15.0	3.3	6.5
	10	5.0	8.3	14.4	3.7	—
0.7	2	2.2	4.1	10.1	1.1	4.5
	5	2.6	5.2	11.4	1.4	3.5
	10	2.8	6.5	11.4	2.6	—

表 3　併用法による平均誤差率 (%)

	m	10			20	
p	r ＼ q	2	5	10	2	5
0.3	2	2.1	3.0	3.8	2.5	5.6
	5	2.7	6.0	4.7	3.1	9.1
	10	4.8	7.2	9.1	5.5	—
0.5	2	0.9	2.5	3.4	1.0	3.7
	5	2.1	4.0	6.6	2.0	5.3
	10	1.8	5.9	6.7	2.9	—
0.7	2	0.7	1.2	2.8	0.5	2.0
	5	1.6	2.6	5.3	0.9	2.3
	10	1.9	5.3	6.1	1.6	—

$$\bar{l}_k=(\sum_{i=1}^{m} l_{ki})\times p$$

(3) 目標ベクトルの要素を，$W_1=W_2=\cdots=W_r=1$ とする．したがって，$w_j=1/\sqrt{r}$ となる．

(4) プロジェクト数は $m=10$, 20 の場合を考え，$m=10$では，制約数 q と目標数 r をそれぞれ 2，5，10 の 3 通りに設定してすべての組合せについて各 100 問ずつのシミュレーションを行なう．$m=20$では，$(q,r)=(2,2),(2,5),(2,10),(5,2),(5,5)$ の 5 通りの組合せについて各50問ずつのシミュレーションを行なう．なお，最適解は陽的列挙法で求める．

(5) 近似解の誤差率を次のように定義する．

$$誤差率=(f^0-f^a)/f^0\times 100(\%)$$

f^0：最適解の目的関数値

f^a：近似解の目的関数値

表 4　併用法による正答率 (%)

	m	10			20	
p	r ＼ q	2	5	10	2	5
0.3	2	78	65	76	26	22
	5	70	60	65	38	8
	10	60	60	56	24	—
0.5	2	74	57	57	54	18
	5	63	55	38	20	6
	10	64	39	39	24	—
0.7	2	74	70	57	56	30
	5	64	56	37	52	22
	10	54	44	35	30	—

表 5　併用法による 5 %未満解率 (%)

	m	10			20	
p	r ＼ q	2	5	10	2	5
0.3	2	86	77	82	80	52
	5	85	60	74	70	36
	10	69	62	62	74	—
0.5	2	93	80	71	98	70
	5	80	70	59	90	52
	10	84	55	55	78	—
0.7	2	97	88	77	100	92
	5	87	79	61	98	92
	10	78	72	65	94	—

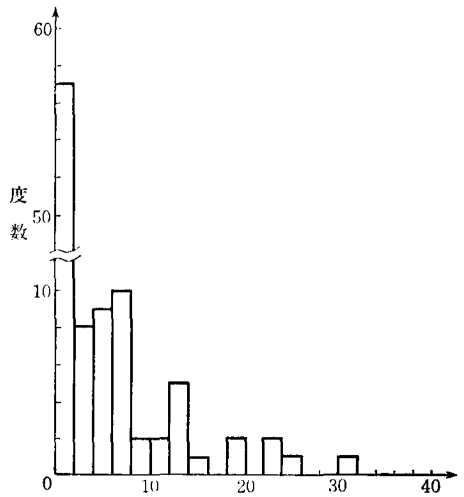

図 5　誤差率のヒストグラム

各条件の問題における前進法による近似解の平均誤差率(100問または50問の誤差率の平均値）を表 1 に，後退法による平均誤差率を表 2 に示す．

次に，同一の問題に両手法を別々に適用し，得られた解のうち目的関数値の大きな近似解のほうを採用する方法(これを併用法と呼ぶ)の平均誤差率を表 3 に示す．併用法については，正答率（各条件の問題において，近似解が最適解に一致した割合)を表 4 に，5 %未満解率(各条件の問題において，誤差率が 5 %未満であった割合)を表 5 に示す．また，誤差率の分布形の一例として，図 5 に $m=10, q=r=5, p=0.5$ における誤差率のヒストグラムを示す．

実験結果を分析すると次のような考察が得られる．

① 前進法と後退法の比較について

前進法，後退法に共通して，平均誤差率には次のような傾向がみられる．

(i) 制約がきつい($p=0.3$）場合は平均誤差率は悪くなり，制約がゆるい($p=0.7$）場合は良くなる．これは，最適解と比較して不適当なプロジェクトを採

択してしまった場合，$p=0.7$では他の採択プロジェクトでカバーできる可能性があるが，$p=0.3$では採択プロジェクト数が少ないので，誤差率が悪くなるためと考えられる．

(ii) 制約数・目標数が大きくなると，平均誤差率は少しずつ悪くなる．これは，多目標・多制約になるほど多次元での情報が一元化されてしまうために，効率指標の信頼度が落ちることによる影響と考えられる．

(iii) プロジェクト数が10から20に増加しても同様な傾向がみられる．

次に，前進法と後進法の比較を行なうと，$m=10$の場合には次のような傾向がみられる．

(i) 制約がきついときには前進法が，ゆるいときには後退法が有効である．

(ii) 制約数が少ないときには後退法が，多いときには前進法が有効である．

(iii) 目標数が少ないときには前進法が，多いときには後退法が有効である．

$m=20$の場合には，上述のような傾向はあまりはっきり見られない．

② 併用法について

平均誤差率についてみると，制約がゆるい場合，制約数・目標数が大きい場合に悪くなっているが，前進法，後退法をそれぞれ単独で適用した場合に比べると大きく改善されている(表3)．正答率は，$m=20$のほうが$m=10$の場合よりも悪くなっているが，5％未満解率はそれほど変化していない．これは，mが大きくなるほど実行可能な組合せが増加し，最適解に近い実行可能解が多くなって，アルゴリズムによってそれらの中の1つの解には到達するが，それが必ずしも最適解とは限らないからであると考えられる．

6. 結　論

本研究では，複数の目標をバランスよく達成させるようなプロジェクト選択問題をマックスミニ型効用関数を用いて定式化し，空間法の考え方にもとづいた近似解法を提案した．前進法，後退法を単独に用いるよりも，併用法を用いると平均誤差率は低く押さえられることがわかったが，より規模の大きな問題に対する検討や，他の近似解法との比較が今後の課題である．

参 考 文 献

[1] Bitran, G. R.: Linear Multiple Objective Programs with Zero-One Variables, Mathematical Programming, 13, 2, (1977), 121-139

[2] Bitran, G. R.: Theory and Algorithms for Linear Multiple Objective Programs with Zero-One Variables, Mathematical Programming, 17, 3, (1979), 362-390

[3] 福川忠昭，山口俊和，山梨哲哉：プロジェクト選択問題に対するヒューリスティック・アプローチ，日本経営工学会誌，30，1，(1979)，37-42

[4] 福川忠昭，山口俊和，福島悟：複数の目標をもつプロジェクト選択問題に対するヒューリスティック・アプローチ，日本経営工学会誌，32，2，(1981)，131-138

[5] 伏見多美雄，山口俊和：複数の目標をバランスよく達成するための数理計画的方法，日本オペレーションズ・リサーチ学会邦文機関誌，19，2，(1975)，88-102

[6] 伏見多美雄：多目標のバランスよい達成をねらいとする投資予算配分計画，慶応経営論集，3，1，(1981)，19-44