メタヒューリスティクスを用いた最適化による地域間産業連関表の推計

平　松　　　燈[†1]
加　藤　康　彦[†1]
井　上　寛　規[†2]

本稿では，地域間産業連関表の推計法としてメタヒューリスティクスを用いた最適化法によって推計結果の精度を高める方法を提案する．性能評価として，日本の９地域間12部門の地域間産業連関の推計結果による比較実験を行う．比較実験ではRAS法単体の結果に加え，GAの代わりに焼きなまし法（Simulated annealing, SA）を組み合わせた場合との比較も行うことで，提案手法の有効性を示す．

キーワード：地域間産業連関表（interregional input-output table），反復手法（iterative technique），RAS法（RAS method），進化計算（evolutionary computation），遺伝的アルゴリズム（genetic algorithm），マトリクス・バランシング（Matrix Balancing）

1　はじめに

産業連関表はレオンチェフによって考案された地域や一国の産業間の取引関係をまとめた統計表である．経済構造の把握や経済波及効果など様々な分析に利用できることから世界各国で作成されるに至っている．

我が国においては都道府県ごとに産業連関表が作成されており，それらを取りまとめた地域間表が経済産業省によって公表されている．これは各都道府県を北海道・東北・関東・中部・近畿・中国・四国・九州・沖縄の９地域に統合した表であり，都道府県間表とはなっていない．よって，地域間よりも詳細な分析を行いたい場合には，各都道府県で公開されている表を利用して都道府県間表を作成する必要がある．都道府県間表については宮城・石川・由利・土谷（2003）や萩原（2011）などによって作成方法が述べられている．武者（2012）や中部圏社会経済研究所（2013）では同様の方法で各都道府県表から地域内表を作成し，地域内の経済構造を分析している．これらとは逆に山田（2013）のように各都道府県の産業連関表を利用して市町村レベルの分析を行う試みもある．

地域間産業連関表としてはIsard型とChenery-Moses型の２つのモデルがよく知られており，日本の経済産業省（Ministry of Economy, Trade and Industry, METI）で作成されている産業連関表はChenery-Moses型が採用されている．Chenery-Moses型では，地域間交易係数を用いて地域別の産業連関表を連結し，地域間産業連関表が作成される．そして，その地域間交易係数の推計に際して，初期値として地域間の取引に関するデータベースを利用するか推計式を利用するかによって，サーベイ法とノ

[†1] 熊本学園大学（Kumamoto Gakuen University）
[†2] 京都大学経済研究所（Institute of Economic Research, Kyoto University）

表１　地域（都道府県）別産業連関表（取引基本表）

		Intermediate demand			Total intermediate demand B	Total regional final demand F	International exports E	Exports G	International imports M	Imports H	Regional production (gross outputs) Y
		Sector 1	…	Sector k							
Intermediate input	Sector 1	x_{11}	…	x_{1k}	$\sum_{j=1}^{k} x_{1j}$	f_1	e_1	g_1	m_1	h_1	y_1
	⋮	⋮	⋱	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	Sector k	x_{k1}	…	x_{kk}	$\sum_{j=1}^{k} x_{kj}$	f_k	e_k	g_k	m_k	h_k	y_k
Total of intermediate sectors C		$\sum_{i=1}^{k} x_{i1}$	…	$\sum_{i=1}^{k} x_{ik}$	$\sum_{i=1}^{k}\sum_{j=1}^{k} x_{ij}$	$\sum_{i=1}^{k} f_i$	$\sum_{i=1}^{k} e_i$	$\sum_{i=1}^{k} g_i$	$\sum_{i=1}^{k} m_i$	$\sum_{i=1}^{k} h_i$	$\sum_{i=1}^{k} y_i$
Total of gross value added sectors V		v_1	…	v_k	$\sum_{j=1}^{k} v_j$						
Regional production (gross inputs) Y		$\sum_{i=1}^{k} x_{i1} + v_1$	…	$\sum_{i=1}^{k} x_{ik} + v_k$	$\sum_{j=1}^{k}\left(\sum_{i=1}^{k} x_{ij} + v_j\right)$						

ンサーベイ法に大きく分けられることになる．地域間の取引に関するデータベースを利用する方がより実情に即した地域間交易係数が得られるが，膨大な作成コストを要する．しかし，対象とする地域によっては相応のデータベースが存在しない場合もあり，完全に推計式に頼らざるを得ないこともある．

RAS 法では地域間交易係数の初期値のバランスは崩さずに調整を行うため，地域間交易係数の初期値として用いるデータや推計式の信頼性が，作成される地域間産業連関表の正確さに直接影響する．そこで，本稿では実数値 GA によって地域間交易係数自体を変化させる（改善する）ことで，地域間交易係数の初期値の誤りに由来する地域間産業連関表の推計誤差を緩和する．

本稿の構成は以下の通りである．第２節で地域別の産業連関表と地域間の産業連関表の構造について整理する．第３節で提案アルゴリズムの解説とその性能評価を行う．最後に，第５節を本稿の帰結とする．

2　産業連関表の構造

2.1　地域（都道府県）別産業連関表

各都道府県別の産業連関表の各行は中間需要，地域内最終需要，輸出と移出（または輸移出），輸入と移入（または輸移入），生産額に整理できる．つまり，k 部門（産業）から成る１地域の産業連関表は表１のようになる．需要面から見た各 i 財の生産額とその他の項目の関係は次式で書くことができる．

$$
\begin{aligned}
y_i &= \sum_{j=1}^{k} x_{ij} + f_i + e_i + g_i - m_i - h_i \\
&= b_i + f_i + e_i + g_i - m_i - h_i
\end{aligned}
\tag{1}
$$

y_i ：部門 i の生産額
x_{ij} ：部門 j の生産に利用される部門 i の財の中間需要額
f_i ：部門 i の地域内最終需要合計（除輸出）
e_i ：部門 i の輸出額
g_i ：部門 i の移出額
m_i ：部門 i の輸入額
h_i ：部門 i の移入額
b_i ：部門 i の中間需要合計

また，供給面から見た各 j 財の生産額と中間投入および粗付加価値額には次の関係が成り立つ．

$$
\begin{aligned}
y_j &= \sum_{i=1}^{k} x_{ij} + v_j \\
&= c_j + v_j
\end{aligned}
\tag{2}
$$

y_j ：部門 j の生産額
x_{ij} ：部門 i から部門 j へ供給される財の中間投入額
v_j ：部門 j の粗付加価値額
c_j ：部門 j から投入（供給）されるすべての

財の総額

したがって，各 i 財に対して以下の需給バランス式が成立する．

$$\sum_{j=1}^{k} x_{ij} + f_i + e_i + g_i - m_i - h_i = \sum_{j=1}^{k} x_{ji} + v_i \quad (3)$$

2.2　多地域（都道府県）間産業連関表

これに対し，作成目標である地域間非競争輸入型の都道府県間産業連関表は中間需要，県内最終需要，輸出，輸入から成り，移出と移入が中間需要に含まれた形となる．n 地域 k 部門から成る産業連関表において，地域 o の需要面から見た各 j 財の生産額は式(1)を多地域に拡張した式で表すことができる．(注1)

$$y_i^o = \sum_{d=1}^{n}\sum_{j=1}^{k} x_{ij}^{od} + f_i^o + e_i^o - m_i^o \quad (4)$$

y_i^o：地域 o の部門 i における生産額

x_{ij}^{od}：地域 d の部門 j における生産に利用される地域 o の部門 i の財の中間需要額

f_i^o：地域 o の部門 i における地域内最終需要合計

e_i^o：地域 o の部門 i における輸出額

m_i^o：地域 o の部門 i における輸入額

ここで

$$f_i^o = \sum_{d=1}^{n} \phi_i^{od} \quad (5)$$

ϕ_i^{od}：地域 o の部門 i で生産された財の地域 d の地域内最終需要合計

とすると，式(4)は以下のように書き換えることができる．

$$\begin{aligned} y_i^o &= \sum_{d=1}^{n}\sum_{j=1}^{k} x_{ij}^{od} + \sum_{d=1}^{n} \phi_i^{od} + e_i^o - m_i^o \\ &= \sum_{d=1}^{n}\left(\sum_{j=1}^{k} x_{ij}^{od} + \phi_i^{od}\right) + e_i^o - m_i^o \end{aligned} \quad (6)$$

同様に，地域 d の供給面から見た各 j 財の生産額は以下のように定式化される．

$$\begin{aligned} y_j^d &= \sum_{o=1}^{n}\sum_{i=1}^{k} x_{ij}^{od} + v_j^d \\ &= c_j^d + v_j^d \end{aligned} \quad (7)$$

y_j^d：地域 d の部門 j における生産額

x_{ij}^{od}：地域 o の部門 i から地域 d の部門 j へ供給される財の中間投入額

c_j^d：地域 d の部門 j における中間投入額合計

v_j^d：地域 d の部門 j における粗付加価値額

2.3　地域別産業連関表から多地域間産業連関表への変換

Chenery–Moses 型の地域間産業連関表の作成では，地域間交易係数を導入することによって，各都道府県別の産業連関表で得られるデータを都道府県間産業連関表のデータに変換することができる．地域間交易係数は財 i の地域 d での総需要量の内で地域 o からの移入量の割合として定義される．

$$t_i^{od} = \frac{q_i^{od}}{\sum_{p=1}^{n} q_i^{pd}} \quad (8)$$

t_i^{od}：地域 od 間の部門 i における地域間交易係数

q_i^{od}：部門 i 財の地域 o から地域 d への流動（移出）量

都道府県間産業連関表の中間需要は地域間交易係数，投入係数および生産額から次式で求められる．

$$x_{ij}^{od} = t_i^{od} a_{ij}^d y_j^d \quad (9)$$

x_{ij}^{od}：地域 o の部門 i から地域 d の部門 j への中間投入

a_{ij}^d：地域 d における部門 ij 間の投入係数

y_j^d：地域 d における部門 j の生産額

投入係数 a と生産額 y は都道府県別の産業連関表から得ることができるデータである．都道府県別の投入係数表も公開されているが，投入係数は中間投入と生産額から容易に求めることができる．

表２　地域間産業連関表（取引基本表）

destination / origin			Region 1 Intermediate demand Sector 1	…	Region 1 Intermediate demand Sector k	Region 1 Regional final demand	•	Region n Intermediate demand Sector 1	…	Region n Intermediate demand Sector k	Region n Regional final demand	Total intermediate demand B	Total regional final demand F	International exports E	International imports M	Regional production (gross outputs) Y
Intermediate input	Region 1	Sector 1	x_{11}^{11}	…	x_{1k}^{11}	φ_1^{11}	•	x_{11}^{1n}	…	x_{1k}^{1n}	φ_1^{11}	$\sum_{s=1}^{n}\sum_{j=1}^{k} x_{1j}^{1s}$	f_1^1	e_1^1	m_1^1	y_1^1
		⋮	⋮	⋱	⋮	⋮		⋮	⋱	⋮	⋮	⋮	⋮	⋮	⋮	⋮
		Sector k	x_{k1}^{11}	…	x_{kk}^{11}	φ_k^{11}		x_{k1}^{1n}	…	x_{kk}^{1n}	φ_k^{11}	$\sum_{s=1}^{n}\sum_{j=1}^{k} x_{kj}^{1s}$	f_k^1	e_k^1	m_k^1	y_k^1
	⋮	⋮	⋮			⋮	•	⋮			⋮	⋮	⋮	⋮	⋮	⋮
	Region n	Sector 1	x_{11}^{n1}	…	x_{1k}^{n1}	φ_1^{n1}		x_{11}^{nn}	…	x_{1k}^{nn}	φ_1^{n1}	$\sum_{s=1}^{n}\sum_{j=1}^{k} x_{1j}^{ns}$	f_1^n	e_1^n	m_1^n	y_1^n
		⋮	⋮	⋱	⋮	⋮	•	⋮	⋱	⋮	⋮	⋮	⋮	⋮	⋮	⋮
		Sector k	x_{k1}^{n1}	…	x_{kk}^{n1}	φ_k^{n1}		x_{k1}^{nn}	…	x_{kk}^{nn}	φ_k^{n1}	$\sum_{s=1}^{n}\sum_{j=1}^{k} x_{kj}^{ns}$	f_k^n	e_k^n	m_k^n	y_k^n
Total of intermediate sectors C			$\sum_{r=1}^{n}\sum_{i=1}^{k} x_{i1}^{r1}$	…	$\sum_{r=1}^{n}\sum_{i=1}^{k} x_{ik}^{r1}$	$\sum_{r=1}^{n}\sum_{i=1}^{k} \varphi_i^{r1}$	• •	$\sum_{r=1}^{n}\sum_{i=1}^{k} x_{i1}^{rn}$	…	$\sum_{r=1}^{n}\sum_{i=1}^{k} x_{ik}^{rn}$	$\sum_{r=1}^{n}\sum_{i=1}^{k} \varphi_i^{rn}$	$\sum_{r=1}^{n}\sum_{s=1}^{n}\sum_{i=1}^{k}\sum_{j=1}^{k} x_{ij}^{rs}$	$\sum_{r=1}^{n}\sum_{i=1}^{k} f_i^r$			
Total of gross value added sectors V			v_1^1	…	v_k^1		•	v_1^n	…	v_k^n		$\sum_{s=1}^{n}\sum_{j=1}^{k} v_j^s$				
Regional production (gross inputs) Y			y_1^1	…	y_k^1		• •	y_1^n	…	y_k^n		$\sum_{s=1}^{n}\sum_{j=1}^{k} y_j^s$				

$$a_{ij}^{d} = x_{ij}^{d} / y_{j}^{d} \tag{10}$$

a_{ij}^{d}：地域 d における部門 ij 間の投入係数
x_{ij}^{d}：地域 d における部門 i から j への中間投入（注2）
y_{j}^{d}：地域 d における部門 j の生産額

また，地域別・財別の最終需要φも地域間交易係数を利用して，都道府県別産業連関表の地域内最終需要額 f から得られる．

$$\phi_{i}^{od} = t_{i}^{od} f_{i}^{d} \tag{11}$$

f_{i}^{d}：地域 d における部門 i の地域内最終需要合計

式(9)と式(11)を式(6)に代入することで次式を導出できる．

$$y_{i}^{o} = \sum_{d=1}^{n} t_{i}^{od} \left(\sum_{j=1}^{k} a_{ij}^{d} y_{j}^{d} + f_{i}^{d} \right) + e_{i}^{o} - m_{i}^{o} \tag{12}$$

つまり，地域間交易係数さえ得られれば都道府県別の産業連関表から都道府県間産業連関表を作成することができるのである．

3　メタヒューリスティクスを用いた最適化による地域間産業連関表の推計

本節では，メタヒューリスティクスを用いた最適化による地域間産業連関表の推計方法及び推計精度について述べる．

具体的なメタヒューリスティクスとしてRAS法と実数値GAを組み合わせたハイブリッドアルゴリズムを提案する．RAS法は地域間産業連関表の推計として使われるポピュラーかつ簡便な方法である．RAS法による地域間産業連関表の推計では他のデータベースを利用して求めた交易係数から作成される地域間投入係数行列を所与として，そのバランスを保持しつつ産業連関表を作成する．しかしながら，交易係数は他のデータベースや推計式を使って求めた値であるため，真の値とはなっていない．そのため，推定された産業連関表の値にも誤差が生じる．

本稿ではRAS法と実数値GAを組み合わせたハイブリッドアルゴリズムを使って地域間産業連関表の作成，すなわち地域間投入係数行列 $\hat{T}A$ の最適化を行う．地域間産業連関表の行和と列和は都道府県別の産業連関表の行和と列和と同じであり，これらの誤差を最小化する地域間投入係数行列 $\hat{T}A$ をアルゴリズムで求める．RAS法では調整パラメータの行列RとSを使い，データに基づいて与えた地域間投入係数行列 $\hat{T}A$ のバランスを保持しつつ，$\hat{T}A$ の調整を行う．GAでは誤った交易係数 $\hat{T}$ による影響を取り除くため，$\hat{T}A$ の成分自体を変化させて $\hat{T}A$ の調整を行う．

4.1　RAS法と実数値GAのハイブリッドアルゴリズム

ここでは提案アルゴリズムの解説を行う．まず，最適化の対象である地域間投入係数行列について定義しておく．地域間交易係数の初期値（推定値）を分散配置した行列を地域間交易係数行列 $\hat{T}$ とする．以下では読者の理解を助けるため，2地域2部門の例を用いて構造が特殊な行列の中身を記述する．

$$\hat{T} = \begin{pmatrix} \hat{t}_1^{11} & 0 & \hat{t}_1^{12} & 0 \\ 0 & \hat{t}_2^{11} & 0 & \hat{t}_2^{12} \\ \hat{t}_1^{21} & 0 & \hat{t}_1^{22} & 0 \\ 0 & \hat{t}_2^{21} & 0 & \hat{t}_2^{22} \end{pmatrix} \tag{13}$$

各都道府県別の投入係数行列 A^d と零行列をブロックとする区分行列を投入係数行列 A とする．

$$\hat{T} = \begin{pmatrix} \hat{t}_1^{11} & 0 & \hat{t}_1^{12} & 0 \\ 0 & \hat{t}_2^{11} & 0 & \hat{t}_2^{12} \\ \hat{t}_1^{21} & 0 & \hat{t}_1^{22} & 0 \\ 0 & \hat{t}_2^{21} & 0 & \hat{t}_2^{22} \end{pmatrix} \tag{14}$$

地域間交易係数行列 $\hat{T}$ と投入係数行列 A の積を地域間投入係数行列 $\hat{T}A$ とする．地域間投入係数行列 $\hat{T}A$ は地域数が n，部門数が k ならば $nk \times nk$ の行列となる．

各都道府県表における投入係数とサーベイデータや推計式から求めた地域間投入係数行列 $\hat{T}A$ を初期値（starting point）として与え，RAS 法と実数値 GA を組み合わせたハイブリッドアルゴリズムを使ってその地域間投入係数行列 $\hat{T}A$ の最適化を行う．

Step1：RAS 法による地域間投入係数行列 $\hat{T}A$ の調整

RAS 法は基になる投入係数のバランスを保ちつつ，対角行列 R と S を使って地域間投入係数行列 $\hat{T}A$（1地域であれば投入係数行列 A）から計算される中間需要（投入）行列 X の行和と列和を目的の値に近づけていく手法である．R は行和を，S は列和を調整するためのものである．地域間産業連関表の作成において，投入係数行列 A は不変なので，推定値から成る交易係数行列 $\hat{T}$ を調整パラメータの対角行列 R と S を用いて真の T に近づけることに他ならない．

多地域間産業連関表では，地域間交易係数が地域内最終需要にも影響を与えるため，あらかじめ行和の目標値を設定しておくことができない．そこで，都道府県別の産業連関表から与えることが可能な生産額 Y を行和の目標値として利用する．つまり，地域間交易係数行列 $\hat{T}$ は最終需要合計にも影響するため，対角行列 R は中間需要と最終需要に対し同時に調整を行うことになる（注3）．

$$r_i^o = \begin{cases} \dfrac{y_i^o + m_i^o}{\hat{b}_i^o + \hat{f}_i^o + e_i^o - m_i^o + m_i^o} & if \quad \hat{b}_i^o + \hat{f}_i^o + e_i^o > 0 \\ 0 & otherwise \end{cases} \tag{15}$$

$$\begin{aligned} \frac{y_i^o + m_i^o}{\hat{b}_i^o + \hat{f}_i^o + e_i^o - m_i^o + m_i^o} &= \frac{y_i^o + m_i^o}{\sum_{d=1}^{n}\sum_{j=1}^{k}\hat{x}_{ij}^{od} + \hat{f}_i^o + e_i^o} \\ &= \frac{y_i^o + m_i^o}{\sum_{d=1}^{n}\hat{t}_i^{od}\left(\sum_{j=1}^{k} a_{ij}^d y_j^d + f_i^d\right) + e_i^o} \end{aligned} \tag{16}$$

$\hat{b}_i^o$：$\hat{T}_i^{od}$ から求められる地域 o の部門 i の中間需要合計

ここでは r が負の値をとらないよう移入 m の影響を取り除いている．

一方，列和の目標値を中間投入合計の行ベクトル C とする．これは都道府県別の産業連関表の中間投入額と等しく，あらかじめ与えておくことができる．

$$\begin{aligned} s_j^d &= \frac{c_j^d}{\sum_{o=1}^{n}\sum_{i=1}^{k}\hat{x}_{ij}^{od}} \\ &= \frac{c_j^d}{\sum_{o=1}^{n}\sum_{i=1}^{k}\hat{t}_i^{od} a_{ij}^d y_j^d} \end{aligned} \tag{17}$$

$\hat{x}_{ij}^{od}$：地域間交易係数の推定値 $\hat{t}_i^{od}$ から求められる中間投入 x_{ij}^{od}

RAS 法は反復手法であり，$R\hat{T}AS$ を新たな $\hat{T}A$ として置き換え，R と S による調整を何度も繰り返すことによって $\hat{T}A$ が最善されていく．通常, $\hat{T}A$ は収束することから，アルゴリズムの終了条件として収束判定が用いられることが一般的である．ただし，例外も考慮して本稿では繰り返し回数の上限を設定している．RAS 法のアルゴリズムの疑似コードを Appendix A.1として記載する．

$$R = \begin{pmatrix} r_1^1 & 0 & 0 & 0 \\ 0 & r_2^1 & 0 & 0 \\ 0 & 0 & r_1^2 & 0 \\ 0 & 0 & 0 & r_2^2 \end{pmatrix} \tag{18}$$

$$S = \begin{pmatrix} s_1^1 & 0 & 0 & 0 \\ 0 & s_2^1 & 0 & 0 \\ 0 & 0 & s_1^2 & 0 \\ 0 & 0 & 0 & s_2^2 \end{pmatrix} \tag{19}$$

ここでは提案アルゴリズムの基礎となる2つのアルゴリズムについて簡単に説明する．

Step2：実数値GAによる地域間投入係数の修正

遺伝的アルゴリズム（Genetic Algorithm, GA）は生物が環境に適応して進化するメカニズムを模倣して目的関数を最適化する解を見つけ出す工学的手法である．この手法は解析的に解くことができないNP–hardに属する問題の近似解を求める際に用いられる．GA は John Holland によって1975年に初めて提案されて以降，盛んに

研究されている．

GA では自然界の生物が世代を重ねて環境に適応していくように，交叉（交配）と突然変異によって目的関数を最大化する解を探索していく．GA では交叉（Crossover）と突然変異（Mutation）などの遺伝子（解）操作を遺伝オペレータと呼ぶ．最も基礎的な GA は特に単純遺伝的アルゴリズム（Simple Genetic Algorithm, SGA）と呼ばれ，0–1整数値のビットストリングから成る遺伝子（解表現）によって，問題を解く．

SGA は問題の設計変数が実数値であっても解を導出することができるが，より実数値問題に特化させた手法として実数値 GA（Real-coded GA）が存在する．実数値 GA では遺伝子を実数値で構成し，実数の連続性を生かした遺伝オペレータが採用される．本研究が対象とする地域間投入係数行列の最適化は実数値の問題であるため，実数値 GA を基礎にする．そこで以下では，実数値 GA の処理手順に沿って，アルゴリズムの簡単な説明を行う．実数値 GA のアルゴリズムの疑似コードを Appendix A.2として記載する．

遺伝子長が長すぎると優良解が発見される確率が下がるため，行毎に最適化を図る．具体的には，地域間投入係数行列 $\hat{TA}$ の第 i 行の各成分を遺伝子座に対応付け，実数値 GA で調整を行う．この際，行和の誤差の小さい順に処理を行うと性能が向上する．

Step2–1：初期集団生成

通常，実数値 GA では初期集団の生成にあたり，設計変数の値が取り得る定義域が必要となる．しかし，地域間投入係数の下限値は０と考えられるものの上限値は定かではない．そこで，本稿では RAS 法による反復（調整過程）で得られる地域間投入係数行列を実数値 GA の初期集団として利用する．

Step2–2：評価（evaluation）

各遺伝子（個体）の表す変数値を目的関数に代入した計算結果から適応度関数を定義して，各遺伝子の評価値を決定する．

各個体の評価値として以下の適合度関数を用いる．

$$f = 1 \Big/ \sum_{n}\sum_{k}\left(y_{nk} - \hat{y}_{nk}\right)^2 \tag{20}$$

ここでは y を目標値, $\hat{y}$ を個体 i を目的関数に代入した計算結果とする．これは目標値と推定値の残差平方和（Residual sum of squares, RSS）の逆数となっている．つまり，残差平方和が小さいほど個体の評価値は高くなる．

Step2–3：選択（selection）

次世代の親となる個体のペアを選択していく．遺伝オペレータによって変化させる個体の割合を *GAP*（注４）とすると，生成されるペアの数は *GAP/2* となる．ここではよく知られたいくつかの選択方法（selection strategy）を紹介する．これらの選択方法は SGA においても使われるごく一般的な方法である．

1．ルーレット選択

ルーレット戦略とは各個体の適合度が全体に占める割合を選択確率とし，その確率にしたがって親となる個体を選択していくというものである．選択確率は次式に従って計算される．

$$prob_i = f_i \Big/ \sum_{j=1}^{popSize} f_j \tag{21}$$

ここで, f_i は個体 i の適合度を意味する．

2．トーナメント選択

その名の通りトーナメントを行って最も高い適合度を持つ個体を親として選択する方法である．トーナメントサイズ *tournament_size* によって選択の強さを調整することができる．

3．切り捨て選択（truncate selection）

これはランキング法の中でもっとも単純

な方法で，適合度の高い上位の個体のみを次世代の親とする方法である．

Step2-4：交叉（crossover）

前 Step で選択した親個体を掛け合わせて子個体を生成する．実数値 GA のための交叉方法としては，算術交叉（arithmetical crossover），ブレンド交叉（blend crossover alpha, BLX-α），単峰性正規分布交叉（unimodal normal distribution crossover, UNDX）などがよく知られた手法である．

1．算術交叉（arithmetical crossover）

親となる個体のペア $P^1=\left(p_1^1,p_2^1,\cdots,p_{nbit}^1\right)$ と $P^2=\left(p_1^2,p_2^2,\cdots,p_{nbit}^2\right)$ をパラメータ $\lambda\in[0,1]$ で掛け合わせることで２つの子個体 $C^1=\left(c_1^1,c_2^1,\cdots,c_{nbit}^1\right)$ と $C^2=\left(c_1^2,c_2^2,\cdots,c_{nbit}^2\right)$ を生成する．パラメータ λ は一様乱数でその都度ランダムに決定される．

$$\begin{aligned}C_1&=\lambda P_1+(1-\lambda)P_2\\C_2&=(1-\lambda)P_1+\lambda P_2\\\lambda&=\mathrm{rand}_{\mathrm{uni}}\left(0,1\right)\end{aligned}\tag{22}$$

2．ブレンド交叉（BLX-α）

BLX-α では親個体が持つ実数ベクトルの各変数の区間 I_i を両側に αI_i だけ拡張した区間から一様乱数に従ってランダムに子個体を発生させる．

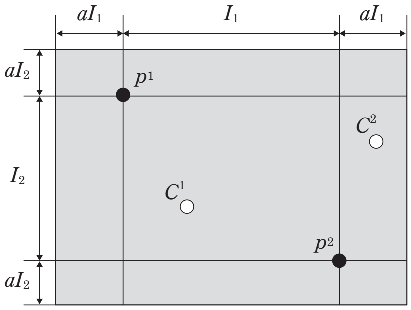

図１　BLX-α

$$\begin{aligned}c_i^1&=\mathrm{rand}_{\mathrm{uni}}\left(\min\left(p_i^1,p_i^2\right)-\alpha I_i,\max\left(p_i^1,p_i^2\right)+\alpha I_i\right)\\c_i^2&=\mathrm{rand}_{\mathrm{uni}}\left(\min\left(p_i^1,p_i^2\right)-\alpha I_i,\max\left(p_i^1,p_i^2\right)+\alpha I_i\right)\\I_i&=\left|p_i^1-p_i^2\right|\end{aligned}\tag{23}$$

ただし，パラメータ α はあらかじめ設定してやる必要がある．

3．単峰性正規分布交叉（UNDX）

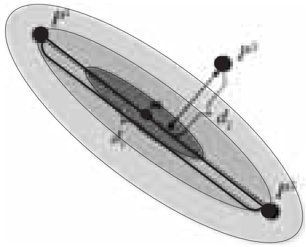

図２　UNDX

UNDX では２つの親個体に加えて補助的に３つ目の個体を利用して，２つの子個体を生成する．２つの親個体を結ぶ線分周辺に正規分布に従って子個体が生成される．第３の親は正規分布の標準偏差を決めるために利用される．

$$\begin{aligned}C_1&=mz_1e_1+\sum_{k=2}^{nbit}z_ke_k\\C_2&=mz_2e_2+\sum_{k=2}^{nbit}z_ke_k\\m&=(P_1+P_2)/2\end{aligned}\tag{24}$$

$z_1\sim N(0,\ \sigma_1^2),\ z_k\sim N(0,\ \sigma_2^2),(k=2,\cdots,nbit)$

$$\begin{aligned}&\sigma_1=\beta_1d_1,\sigma_2=\beta_2d_2/\sqrt{nbit}\\&e_1=(P_2-P_1)/\left|P_2-P_1\right|\\&e_i\perp e_j,(i\neq j),(i,j=1,\ldots,nbit)\end{aligned}$$

ここで，d_1 は両親間の距離 d_2 は第３の親と両親を結ぶ軸との距離，e_1 は両親を結ぶ軸方向の単位ベクトル，z_1 と z_2 は正規乱数を表す．β_1，β_2 は定数であり，これらにより子個体が生成される範囲が制御される．

Step2-5：エリート主義（Elitism）

これは必ずしも行われる処理ではないが，エリート解を確実に次世代に残す方法度導入されることがある．この処理では一定割合 *elite_rate*×*popSize* のエリート解をコピーして保持する．

Step2-6：終了条件が満たされるまで Step2-1～

表4　比較実験（2）

Algorithm			RSS（Residual sum of squares） Avg.	Best	Worst
RAS＋GA	Roulette selection	Arithmetical crossover	5.469E＋13	5.369E＋13	5.651E＋13
		BLX－α	2.329E＋12	1.605E＋12	**3.172E＋12**
		UNDX	5.146E＋13	4.558E＋13	5.394E＋13
	Tournament selection	Arithmetical crossover	5.504E＋13	5.503E＋13	5.506E＋13
		BLX－α	**1.427E＋12**	**1.078E＋11**	6.908E＋12
		UNDX	5.341E＋13	5.244E＋13	5.414E＋13
	Truncate selection	Arithmetical crossover	5.505E＋13	5.503E＋13	5.511E＋13
		BLX－α	4.897E＋12	1.220E＋12	1.019E＋13
		UNDX	5.302E＋13	5.259E＋13	5.343E＋13
RAS＋SA			6.986E＋13	5.812E＋13	7.994E＋13

Step2-5の処理を繰り返す

終了条件としてはあらかじめ設定した世代数*max_generation*の経過や解の収束判定などが用いられる．

また，

4.2　性能評価

性能評価実験は日本の地域間のデータをテスト問題として用いて行った．地域間交易係数の推定の初期値になるデータには物流センサス等のサーベイデータを利用している（注5）．47都道府県27部門で収集したデータを9地域12部門に統合してテスト問題とした．

実験で使用した設定およびパラメータは以下のとおりである．RAS法の終了条件には$\hat{TA}$が変化しなくなった場合または最大繰り返し回数*max_repeating*＝10000を超えた場合とした．実数値GAに関するパラメータは個体数*popSize*＝100，世代数*max_generation*＝100，交叉率*cross_rate*＝1，エリート保存率*elite_rate*＝0.1，突然変異確率*mut_rate*＝0.0，個体群のうち遺伝オペレータによって変化させる個体の割合*GAP*＝0.9とした．各選択方法および交叉法に関わるパラメータは，$\lambda = rand_{uni}(0,1)$，$\alpha = 1.0$，$\beta_1 = 0.5$，$\beta_2 = 0.35$，*tournament_size*＝*popSize*×0.1とした．

25回の試行における最良解の残差平方和（Residual sum of squares, RSS）と平均実行時間を表3に示す．加えて，表4には25回の試行のRSSの平均値，最良値，最悪値を示す．RAS法は数理解法であり，毎回必ず同じ解を出力するため，表4では除外している．実験環境はIntel® Core™ i7-2600 CPU @3.40Hz，実装メモリ（RAM）8.00GB，オペレーティングシステムWindows 7（64ビット）である．RAS法単体の結果と提案アルゴリズムの結果に加えてRAS法と焼きなまし法（Simulated Annealing, SA）を組み合わせた場合の結果を記載している．焼きなまし法は一定確率で改悪解への変化も許容することで局所解から脱出する術を持たせたアルゴリズムである．

比較実験の結果から，RAS法単体の結果に比べて大きな誤差の改善が見られる．また，焼きなまし法ではそれほどの改善は見られず，山登り法（Hill Climbing, HC）タイプのローカルサーチを組み合わせてもGAほどの効果は期待できないことがわかる．また，実数値GAの中でもBLX-αによる結果が選択方法に関わらず最も良い．これは子個体の生成される範囲が関係していると思われる．算術交叉は親同士を結ぶ線分上に子個体が生成され，UNDXでは親同士を結ぶ線分上の周囲に子個体が生成される．その

一方で，BLX−αでは２つの親個体の変数の区間を拡張した超直方立方体内に子個体が生成される．つまり，BLX−α，UNDX，算術交叉の順で子個体の生成範囲が広い．これは実験結果の性能順と一致していることがわかる．

表４では25回の試行のRSSの平均値，最良値，最悪値を比較している．地域数と部門数が多くなるほど，解の出力に要する時間が増加するため，対象の問題の規模に応じて適切な方法を選択する必要がある．時間的な制約により少ない回数の試行での結果を採用する場合，最悪値において最良の性能を示すルーレット選択とBLX−αの組み合わせが適している．十分な試行回数が重ねられる状況であれば，トーナメント選択とBLX−αの組み合わせが推奨される．

5　おわりに

本稿では，RAS法と実数値GAを組み合わせたメタヒューリスティクスによって，従来のRAS法よりも精度の高い地域間産業連関表の推計方法を提案した．従来のRAS法では交易係数の正確さがそのまま作成される地域間産業連関表の正しさにつながるため，交易係数の推定に大きな労力を費やす必要があった．それに対し，今回提案する手法では，地域間投入係数行列の成分の値を個別にGAで調整するため，交易係数の多少の誤差を許容することができる．市町村間連関表のように交易係数として利用可能なデータベースがほとんどなく，モデルによる推定に頼るしかないケースに対して，本アルゴリズムは特に有用であると思われる．

また，RAS法に組み合わせるアルゴリズムとして，実数値GAは山登り法（Hill Climbing, HC）タイプの探索法以上の性能を発揮することを示した．組み合わせた実数値GA自体は基本的なものであり，理解と導入が容易である．

地域別産業連関表を地域間交易係数を用いて結合するChenery–Moses型の地域間産業連関表生成であれば，RAS法以外の手法に対しても実数値GAを組み合わせることが可能であると考えられる．ただし，その場合には実数値GAの初期集団をどのように与えるかが一番の課題となるであろう．

謝辞

本稿は熊本学園大学付属産業経営研究所の研究助成（平成26年度～27年度）「多目的遺伝的アルゴリズム（GA）による地域間産業連関データを用いた物流ネットワークの最適化シミュレーション」の研究成果の一部です．

注

⑴　地域を表すインデックスとしてはrとsが使われるのか一般的だが，本稿では出発地（origin）と目的地（destination）から，あえてoとdを使用している．

⑵　地域間非競争輸入型の都道府県間産業連関表においては$x_{ij}^{d}=\sum_{o=1}^{n}x_{ij}^{od}$であるので，$x_{ij}^{d}\neq x_{ij}^{dd}$である点に注意されたい．

⑶　一般的にはRAS法の適用は内生部門に限られるため，都道府県間表において地域間投入係数を最終需要に対しても同時に調整する場合には注意が必要である．

⑷　*GAP*×*popSize*の個体はそのまま次世代に残る．これはSteady–State Selectionと呼ばれる．

⑸　利用データベースの詳細については宮城et al.(2003)に譲る．

参考文献

[1]　Ishikawa, Y. and Miyagi, T.: “The Construction of a 47-Region Inter-regional Input-Output Table, and Inter-regional Interdependence Analysis at Prefecture Level in Japan,” *ERSA conference papers*, No. ersa04p432, European Regional Science Association (2004)

[2]　宮城俊彦，石川良文，由利昌平，土谷和之：“地域内産業連関表を用いた都道府県間産業連関表の作成”，

土木計画学研究　論文集，Vol. 20，No. 1，pp. 87-95（2003）
[3] 萩原泰治："47都道府県間接続産業連関表の作成と分析"，神戸大學經濟學研究年報，Vol. 58，pp. 33-46（2011）
[4] 武者加苗："関西地域間産業連関表による関西経済の構造：2000年版および2005年版の比較分析"，経済学論究，Vol. 65，No. 4，pp. 199-222（2012）
[5] 公益財団法人 中部圏社会経済研究所："中部圏地域間産業連関表（2005年版）の活用～原表の活かし方と実証分析の例示～"（2013）
www.criser.jp/research/documents/renkan_2012.pdf
[6] 山田光男："グラビティー RAS 法による地域間交易の推計—愛知県内地域間産業連関表を事例として—"，Chukyo University Institute of Economics Discussion Paper Series,No.1301,(2013)
[7] 安藤朝夫，堺美智雄："産業連関表の都市圏への適用のためのノン･サーベイ改訂について"，土木学会論文集，第401号，Ⅳ-10（1989）

付録

A.1 Pseudo code of RAS method

```
begin
while not termination do
    for o = 1 to the number of prefectures do
        for i = 1 to the number of sectors do
            Calculate an adjacent parameter r_i^o
            using the equation
        end for
    end for
    for d = 1 to the number of prefectures do
        for j = 1 to the number of sectors do
        Calculate an adjacent parameter s_j^d
        using the equation
        end for
    end for
    Recalculate an Input Output table using new
    TA ← RTAS
end while
Output the best TA
end
```

A.2 Pseudo code of Real-coded GA

```
begin
Generate initial individuals
while not termination do
    Evaluate the fitness of individuals
    Store the best individual
    for pair = 1 to popSize/2*GAP do
        Create pair of two individuals (parents)
        according to selection strategy
        Generate new individuals (children)
        via crossover
    end for
    for i = 1 to popSize*GAP do
        if (uniform random number [0,1] <
        elite_rate) then
            Replace the individual i for the
            best individual in the generation
        end if
    end for
end while
Output the best individual
end
```

A.3 Pseudo code of our proposal hybrid algorithm of RAS method and Real-coded GA

```
begin
for i = 1 to the number of TA's rows do
    Store i^th row of TA as An initial individual
    [0] for i^th row
end for
while not termination do
    for o = 1 to the number of prefectures do
        for i = 1 to the number of sectors do
            Calculate an adjacent parameter r_i^o
            using the equation
        end for
    end for
    for d = 1 to the number of prefectures do
        for j = 1 to the number of sectors do
            Calculate an adjacent parameter s_j^d
            using the equation
        end for
    end for
    Recalculate an Input Output table using new
    TA ← RTAS
    for i = 1 to the number of TA's rows do
          Store i^th row of TA as An initial
          individual [ireration] for i^th row
    end for
end while
Ranking in an ascending order of row difference
sum of squares
for i = 1 to the number of  TA's rows do
    Give initial individuals for rank ith row
    while not termination do
        Evaluate the fitness of individuals
        Store the best individual (TA)
        for pair = 1 to popSize/2*GAP do
            Create pair of two individuals
            (parents) according to selection
            strategy
            Generate new individuals
            (children) via crossover
        end for
        for k = 1 to popSize*GAP do
            if (uniform random number [0,1]
            < elite_rate) then
                Replace the individual k for
                the best individual in the
                generation
            end if
        end while
    end for
end for
Output the best TA
end
```