第 1 回 多目的最適化 ゼミ

ゼミ担当者　：　林　俊行，鈴木　和徳，山口　尚平
指導院生　：　福永　隆宏，實田　健，金　美和，永松　秀人
開催日　：　2003 年 4 月 25 日

1　はじめに

一般に最適化とはある 1 つの評価基準 (目的) に対する最適化、すなわち単一目的最適化のことを意味する．しかし，実社会に存在する様々な最適化問題は多くの場合、複数の評価基準を同時に考慮する必要がある．例えば，パソコンを購入する場合，その性能，価格，大きさ，外観，重量など複数の評価基準のもとで製品を選ぶ．しかし，通常全ての評価基準が最適であることは少なく，価格を抑えると性能が下がり, 性能を上げると価格が上がるといったように，各評価基準は互いにトレードオフの関係にあることが多い．このように，互いに競合する複数の目的関数のもとで最適解を求める問題を多目的最適化問題という．

2　多目的最適化問題

多目的最適化問題は**複数個の互いに競合する目的関数を与えられた制約条件の中で最小化 (最大化) するような設計変数を求める問題**[1)] と定義される．目的関数が互いに競合し合っているため，全ての目的関数値が最良となる最適解を求めることはできない．

そのため，多目的最適化では**ある目的関数の値を改善するためには，少なくとも他の 1 つの目的関数値を改悪せざるを得ないような解**を求める．このような解をパレート最適解と呼ぶ．以下，多目的最適化問題およびパレート最適解の定義を示す．

2.1　多目的最適化問題の定義

一般に多目的最適化問題は，n 個の設計変数を扱う k 個の互いに競合する目的関数

$$f_i(x_1, x_2, \ldots, x_n) \quad (i = 1, 2, \ldots, k) \tag{1}$$

を，m 個の不等式制約条件

$$v_i(x_1, x_2, \ldots, x_n) \leq 0 \quad (j = 1, 2, \ldots, m) \tag{2}$$

のもとで最小化 (最大化) する問題として定式化される．

2.1.1　パレート最適解

パレート最適解は，多目的最適解問題における解の優越関係により定義される．以下にその定義を示す．ただし，全ての目的関数が最小化であると仮定する．

2.2　定義 (優越関係)

$\mathrm{x}^1, \mathrm{x}^2 \in \Im(\mathrm{x} = (x_1, x_2, \ldots, x_n))$ とする．

a) $f_i(\mathrm{x}^1) \leq f_i(\mathrm{x}^2)(\forall_i = 1, \ldots, k)$ の時，x^1は x^2 に優越するという．

b) $f_i(\mathrm{x}^1) < f_i(\mathrm{x}^2)(\forall_i = 1, \ldots, k)$ の時，x^1は x^2 に強い意味で優越するという．

もし，x^1が x^2 に優越しているならば，x^1のほうが x^2 より優れた解である．そのため多目的最適化では，このような他のどの解にも優越されないような解の探索を行う[1]．次に解の優越関係に基づくパレート最適解の定義について以下に示す．

2.2.1　定義 (パレート最適解)

$\mathrm{x}^0 \in \Im$ とする．

a) x^0に強い意味で優越する $\mathrm{x} \in \Im$が存在しない時, x^0を弱パレート最適解という．

b) x^0に優越する $\mathrm{x} \in \Im$が存在しない時, x^0をパレート最適解という．

目的関数が 2 つの場合のパレート最適解の例を Fig. 1 に示す．一般に，パレート最適解集合が形成する面のことをパレート最適フロントと呼ぶ（Fig. 1 の実線の部分）．Fig. 1 に示すように，一般にパレート最適解は複数存在する．

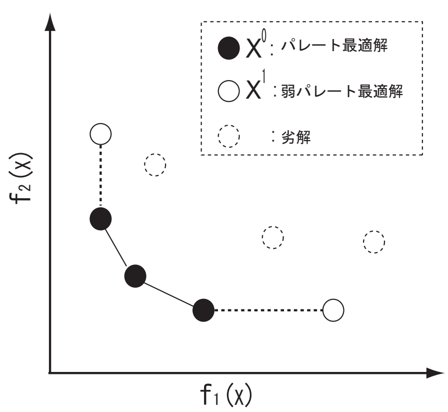

Fig. 1　パレート最適解の概念

[1] このようなほかのどの個体と比較しても劣っていない解を非劣解、劣っている解を劣解と呼ぶ．

2.3　代表的な多目的最適化問題

多目的最適化問題として定式化される多目的ナップザック問題と多目的巡回セールスマン問題を説明する．

2.3.1　多目的ナップザック問題

多目的ナップザック問題は，要素の集合 $E = \{e_1, e_2, \ldots, e_N\}$，要素の体積の集合 $V = \{v_1, v_2, \ldots, v_N\}$，要素の価値の集合 $C^{(1)} = \{c_1^{(1)}, c_2^{(1)}, \ldots, c_N^{(1)}\}, \ldots, C^{(p)} = \{c_1^{(p)}, c_2^{(p)}, \ldots, c_N^{(p)}\}, (v_i > 0, c_i^{(j)} > 0)$ とすると，式 (3)，(4) を満たす要素の部分集合 $E' \subseteq E$ を求める問題として，次のように定式化される．

$$maximize \quad \begin{cases} f_1(x) = \sum_{i=1}^{N} c_i^{(1)} x_i \\ \quad\vdots \\ f_p(x) = \sum_{i=1}^{N} c_i^{(p)} x_i \end{cases} \tag{3}$$

$$subject\ to \quad \sum_{i=1}^{N} v_i x_i \le K \tag{4}$$

ここで，K はナップザックの体積サイズであり，変数 x_i は次のように定義される．

$$x_i = \begin{cases} 1 & \text{if} \quad e_i \in E' \\ 0 & \text{if} \quad e_i \notin E' \end{cases} \tag{5}$$

Fig. 2 に多目的ナップザック問題を簡略に説明したものを示す．

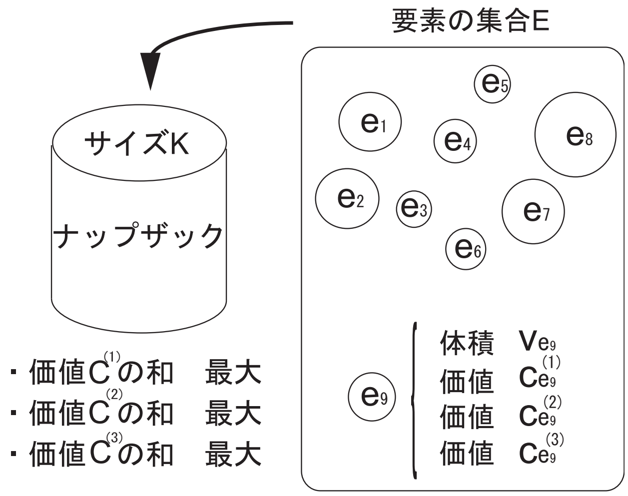

Fig. 2　多目的ナップザック問題

Fig. 2 を用いて多目的ナップザック問題の説明を行う．同図の円柱はナップザックで，その中に詰め込むことのできる要素の総体積を表すサイズは，K で表される．また，円で示した a から i はナップザックに詰め込むべき要素を表しており，各要素には属性として体積と複数の価値が存在する．多目的ナップザック問題はこれらの要素をある組み合わせでナップザックに詰め込むとき，その体積の和がサイズ K を越えないようにした上で，それぞれの価値の和が最大となるような要素の組み合わせを求める多目的最適化問題ということができる．従来よく知られている単目的ナップザック問題と異なる点は，1 つの要素に対し 2 つ以上の価値が存在することである．

2.3.2　多目的巡回セールスマン問題

多目的巡回セールスマン問題は，都市の集合を $U = 1, 2, \ldots, N$ とし，都市 (i, j) 間のコストを $c_{ij}^{(1)}, c_{ij}^{(2)}, \ldots, c_{ij}^{(p)}$ とすると，N 個の都市すべてをちょうど 1 度ずつ巡って出発した都市に戻ってくる巡回路の中から，通過した道のコストの総和を最小にする問題として，以下のように定式化される．

$$minimize \quad \begin{cases} f_1(x) = \sum_{i\in U}\sum_{j\in U} c_{ij}^{(1)} x_{ij} \\ \quad\vdots \\ f_p(x) = \sum_{i\in U}\sum_{j\in U} c_{ij}^{(p)} x_{ij} \end{cases} \tag{6}$$

$$subject\ to \quad \sum_{j\in U}^{N} x_{ij} = 1\ , \forall i \in U \tag{7}$$

$$\sum_{i\in U}^{N} x_{ij} = 1\ , \forall j \in U \tag{8}$$

$$\sum_{i\in V}\ \sum_{j\in U\setminus V} x_{ij} \ge 1, \forall V \subset U$$

$$\forall V \subset U (V \ne \emptyset, V \ne U) \tag{9}$$

$$x_{ij} \in \{0, 1\},\ \forall i, j \in U \tag{10}$$

ここで，$c_{ii}^{(q)} = +\infty, (q = 1, 2, \ldots, p), \forall i \in U$ とした．

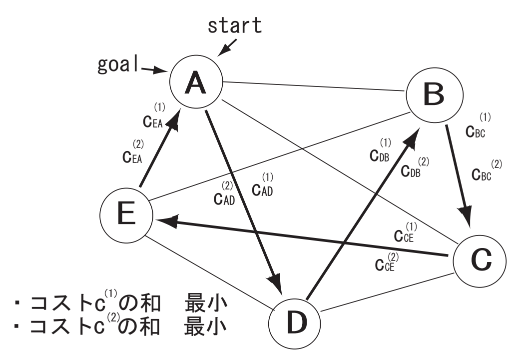

Fig. 3　多目的巡回セールスマン問題

Fig. 3 に多目的巡回セールスマン問題を簡略に説明したものを示す．同図において円で示す A から E は都市

を表している．また，任意の２都市間には，複数のコストが存在するものとする．多目的巡回セールスマン問題は，ある都市を出発し，すべての都市を１回ずつまわって最初の都市に戻ってくるような巡回路のうち，その巡回路の複数のコストのそれぞれが最小となるような巡回路を求める多目的最適化問題であるということができる．同図では，都市 A を出発して再び都市 A に戻ってくるような巡回路の例を示した．従来よく知られている巡回セールスマン問題と異なる点は，任意の２都市間に２つ以上のコストが定義されていることである．

3　多目的最適化手法

一般に多目的最適化問題にはただ１つ解が存在しない．そのため複数のパレート最適解集合の中から意思決定者の選好構造に基づいた解を求める必要がある．そこで，多目的最適化手法は大きく２つのアプローチがある．

- 意思決定者の嗜好を取り入れて多目的を単目的にし，１つの解を求める方法
- パレート最適解を集合として求め，最終的に意思決定者が解を選択する方法

前者の手法では１つの解を求めることはできるが，価値観を完全に定量化することは困難なため，意思決定者の嗜好を十分に反映した解探索を行えないという欠点がある．

一方後者の手法では，幅広いパレート最適解集合を示すことで，意思決定者の選択を助けることができるが，得られた解集合が膨大であった場合，意思決定が困難であるという欠点がある．

本ゼミでは，前者のアプローチとして価値関数を用いた手法．後者の手法としてシミュレーテッドアニーリング (SA)，遺伝的アルゴリズム (Genetic Algorithm:GA) を用いた手法を紹介する．

3.1　価値関数を用いる手法

価値関数を求める方法は意思決定者の選好順序に注目したものである．選好順序とは，意思決定者の価値観を，２つの代替案に対して「どちらの方をより好むか」という二項関係に基づいて考えたものである．このような二項関係が何らかの順序となっているときに，それを選好順序と呼ぶ．選好順序が存在し，かつ関係を保存する関数が明らかであれば，各解候補に一つの実数値を割り当てることが可能となり，この関数は価値関数と呼ばれる．探索以前に価値関数を定義しておくことは，多目的を単目的へと変換することに相当する．現実には，価値関数があらかじめ明らかになっていることは稀なため，意思決定者から選好関係についての情報を対話的に引き出すという方法や，ニューラルネットを用いて目的に対する重みの計算を行うなどの方法で，価値関数を同定する．価値関数を評価することで，ただ一つの最適解を求めることができる．

3.2　多目的 SA（MOSA）

SA は特に単目的組み合わせ最適化問題に対して適用され，大きな成果を上げてきた．多目的 SA は SA を多目的最適化問題に適用できるように一般化したものである．多目的最適化問題を解く上で多目的 SA に要求される条件として,「解品質の向上」と「解の広がりの向上」の２つがある．

- 解品質の向上

  真のパレート解により接近した解を見つけること

- 解の広がりの向上

  真のパレート面全域を幅広く網羅できるようなパレート最適解の集合を求めること

以下に多目的 SA のアルゴリズムを示す.

1. 初期設定
   - 各探索点に対応する各解 $x_l (\in S)$ の初期生成
$x_l := x_{l0}$,
ステップ数 $k := 0$, 温度 $T_k := T_{init}$
   - パラメータ付き受理確率関数のパラメータ λ の設定
   - 解集合のリスト $M := \emptyset$
   - 各$\boldsymbol{x_l}$のうち，他のどの解にも支配されない解を全て M に追加
2. Until 停止条件満足 Loop1
   (a) Until 温度 T_k で十分探索 Loop2
各 $x_l \in S$ について
      i. 解 x_l の近傍から解候補$\boldsymbol{y_l}$を生成
if $\boldsymbol{y_l}$が M の任意の解に対して支配されていない
then $\boldsymbol{y_l}$を M に追加
また，$\boldsymbol{y_l}$に支配される解が M にある場合は，それらを M から削除
      ii. パラメータ付き受理確率関数 $\mathbf{Pr} = (\boldsymbol{x_l}, \boldsymbol{y_l}, \boldsymbol{T_k}, \boldsymbol{\lambda})$の計算
      iii. 受理確率に従い，解候補の受理・棄却判定
if 受理 $\boldsymbol{x_l := y_l}$
else　解の更新は行わない
EndLoop2
   (b) $T_k = \gamma * T_k, (0 \leq \gamma \leq 1)$

(c) $k := k + 1$

EndLoop1

3. ReturnM

多目的 SA のアルゴリズムの説明をする．

1. 初期設定として，ランダムに複数の初期解 x_{l0} を生成する．温度パラメータを $T_k, (k = 0)$ を T_{init} に初期化する．解集合 M に各 x_{l0} のうち，他のどの解にも支配されない解をすべて M に追加する．

2. 解候補を生成し，受理確率関数によって受理棄却判定を行うところは SA と同じである．多目的 SA で異なる点としては解集合のリストの操作である．解候補が M の要素に支配されていないなら，その解候補を M に追加，逆に解候補に M の要素が支配される場合，その要素を削除する．

   温度を低くしつつこれを繰り返す．

3. M がパレート最適解の集合となる．

受理確率関数とは，温度と解候補の位置，重みによってその解候補を受理するかどうかの判定を行うものである．受理確率関数には Rule SL,Rule C,Rule W などが挙げられるが，最も代表的なものは Rule SL である．

それぞれ以下の式で与えられる．T は温度，w は重み，f は目的関数を表わす．添え字 j は目的関数を区別するものであり，$1 \leq j \leq p$ である．

- Rule SL
$$\Pr = \min\{1, \exp(\frac{\Sigma_{j=1}^{p} w_j \Delta f_j}{T})\} \tag{11}$$
- Rule C
$$\Pr = \min\{1, \min_j(\exp\{\frac{w_j \Delta f_j}{T}\})\} \tag{12}$$
- Rule W
$$\Pr = \min\{1, \max_j(\exp\{\frac{w_j \Delta f_j}{T}\})\} \tag{13}$$

Fig. 4 に 2 目的最大化問題において，受理関数により求められる受理確率 Pr の等高線の模式図を示す．同図は，原点に現在の解 x をとり，各座標軸を各目的関数値の差分 $\Delta f_j = f_j(y) - f_j(x)$ とした．また，灰色の領域は受理確率 $Pr = 1$ で受理される解候補 y の存在領域である．

SA の処理能力は，受理確率関数に大きく影響されると考えられる．つまり，解の改善される方向への探索を優先するような受理確率関数は，SA における解の探索を高速化するが，解が局所最適解に陥りやすくなる．逆に解の悪化する方向への探索を優先する受理確率関数は，大域的最適解を見つける可能性は高くなるが，解の探索速度を遅くする．このため，SA において，より高速により良質の解を得るためには，このバランスをうまく調整することが重要である．

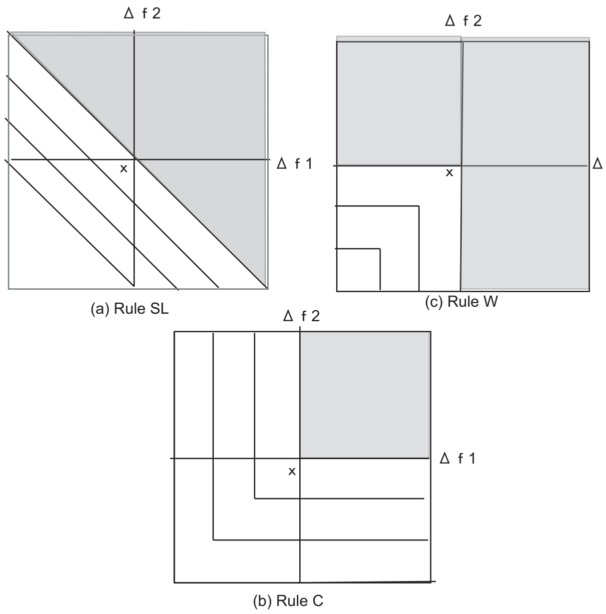

Fig. 4　多目的最適化問題に適用可能な受理確率関数

3.3　多目的 GA（MOGA）

遺伝的アルゴリズム (GA) を多目的最適化問題に適用した多目的 GA はもっとも数多く研究されている．

多目的 GA については次節で詳しく述べている．

4　多目的 GA

多目的最適化問題を解く手法として，近年 GA を用いた研究が数多く報告されている．GA を手法として用いる理由は，GA が多点探索であるため，直接的にパレート最適解集合を求めることが可能であるからである．その際の重要な観点として，多様性維持が挙げられる．つまり，個体をパレート最適解集合上により広範囲にかつ途切れることなく分布させることが重要となる．

多目的最適化問題における GA では，可能領域内に複数の遺伝子を生成し，交叉や突然変異といった遺伝的操作を用いて新たな遺伝子を発生させ，何らかの方法で選択することにより，各個体をパレート最適解集合に近づける．

概念としては Fig. 5 のように，世代が進むに従い個体の作り出すフロントは真のパレート最適曲線（最適解集合）に近づいていくものとして捉えることができる．

本節ではまず GA について解説を行い, パレート最適解を求めるための選択方法について幾つか手法を説明する．

4.1　GA の特徴

GA は自然界における生物の進化を模倣した，最適手法である．GA では世代を形成している個体の集合（個

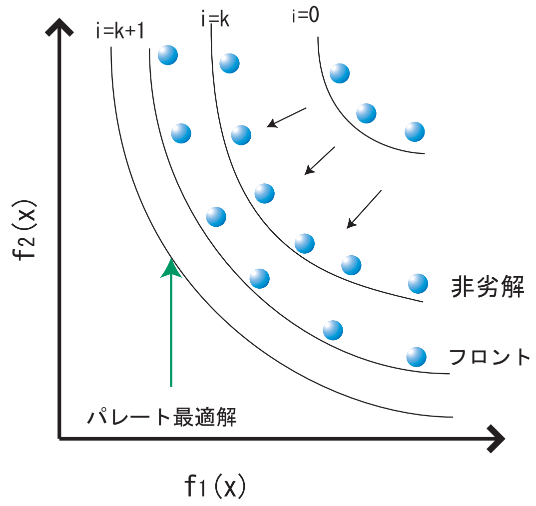

Fig. 5　GA の概念

体群）の中で，環境への適合度の高い個体が次世代により多く生き残り，また交叉および突然変異を起により，次の世代を形成していくといった操作を繰り返すことで，最適な解を探索していく．GA がこれまでの古典的な探索法と大きく異なる点を 2 つ挙げてみる．

- 一点探索ではなく，多点探索である．
- 決定的規則ではなく，確率的オペレータを用いる探査である．

上記における多点探索という特徴を持つ GA では，探索の各段階で個体評価における多目的性を直接扱うことが可能である．すなわち，それぞれの目的関数に対してある程度よい値をとる個体を同時に持ちながら探索を進めることができる．

4.2　GA のプロセス

GA の基本的なプロセスは次のようになっている．

1 （初期化）ランダムな染色体をもつ個体をN個生成して，初期世代の個体群を設定する．

2 （選択）各個体の適合度を計算して，適合度に依存した一定の規則で個体の再生を行う．ここで，適合度の低いいくつかの個体は消滅され，その個体だけ適合度の高い個体が増殖することになる．

3 （交叉）設定された交叉確率や交叉の方法により交叉を行い，新しい個体を生成する．

4 （突然変異）設定されたと突然変異の方法により突然変異を行い，新しい個体を生成する．この結果，新しい世代の個体群が生成される．

5 （終了判定）終了条件を満たせば，その時に得られている最良の個体を問題の準最適解とする．そうでなければ 2 へ戻る．

GA はこのように，選択により望ましい解を重点的に探索すると同時に，交叉と突然変異によって解の探索範囲を広げているので，これらの両方の手続きが有効に作用すれば，その威力を十分に発揮することになる．

5　GA による多目的最適化

GA を多目的最適化問題に適用する場合，非劣解に適切な適合度を与え，次世代に残していくことがキーポイントとなる．従来の「一つの最適解」を求める単目的の場合と異なり，多目的では，他の解に劣っていない解（非劣解）がすべて候補になるため，単純に単目的の適合度をそのまま適応させることはできない．多目的における個体の適合度割り当てに関して，パレート・ランキング法およびパレート・トーナメント法について述べる．

5.1　パレート・ランキング法

パレートランキングによる方法とは，解の優越関係に基づいて定められる個体間の順位を適合度に用いる方法である．個体の順位のつけ方としては，Golderg, Fonseca らによる方法がある．以下では，個体の順位付けがより明確に行える Fonseca によるランキング法を説明する．

Fonseca らのランキング法では，個体 X_i が n_i 個の個体に優越しているとき，X_i のランク $r(X_i)$ を

$$r(X_i) = 1 + n_i \tag{14}$$

のように定める．パレートランキング法の概念図を Fig. 6 に示す．以下，Fig. 6 を例にしてパレートランキング法における手続きを示す．

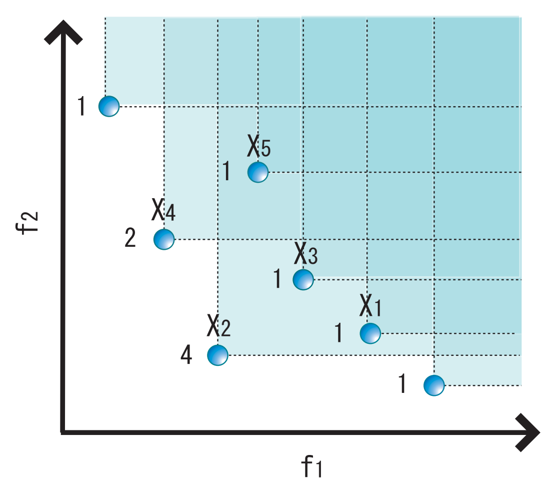

Fig. 6　パレート・ランキング法

(1) Fig. 6 のように，個体 X_1，X_3，X_5 はどの個体に対しても優越していないのでランクは１となる．

(2) 個体 X_2 は，X_3，X_4，X_5 の３個体に，X_4 は，X_5 に優越しているのでそれぞれのランク $r(X_2)$，$r(X_4)$ は，$r(X_2)$=1+3=4，$r(X_4)$=1+1=2 となる．

このランキング法を用いた選択手法としては，ランクの値を適合度に変換し用いるルーレット選択，各世代でパレート最適個体（ランク１の個体）のみ残すパレート保存戦略などがある．

5.2　パレート・トーナメント法

多目的最適化問題に対して，優越関係に基づくパレートトーナメント法を基本とし，これにシェアリングを組み合わせた手法として，以下のアルゴリズムが Horn らによって提案されている．

- 個体集合から競争関係にある二個体（「トーナメント個体」以下Ａとする）優越感関係のテスト用にあらかじめ定められた数の比較個体集合（Ｂとする）をランダムに取り出す．またこのときＢの数は優越関係パラメータ（t_{dom}）によって選出する．
- トーナメント個体のそれぞれについて，Ｂ内の個体との間で優越関係を調べる．この結果，次のようにして１つの個体を選択する．
  - もしＡ内の１個体がＢのすべての個体に優越して，他方の個体がそうでないならば，前者をトーナメントの勝者として選択し次世代への残す．
  - Ａがともに優越するかまたは優越してない（選択される個体が定まらない）ならば，トーナメント結果はシェアリングによって選択（ニッチ数がより小さい方の個体を選出）する．

この方法では，多様なパレート最適解が得られるが，探索が進み難い（探索が遅い）という問題点がある．

5.3　シェアリング

多目的最適化問題に対し GA を適用する際の重要な点は，解をパレート最適解集合上により広範囲にかつ均等に分布するような解を求めることである．その一つの有効な方法として Horn らによって提案されたシェアリングがある．以下では，シェアリングの概念を適応度に反映させる手法について述べる．

各個体についてその個体の近傍の混み具合をニッチ数という．ここではニッチ数を m_{x_i} を

$$m_{x_i} = \sum_{j=1}^{N} s(d(x_i, x_j)) \tag{15}$$

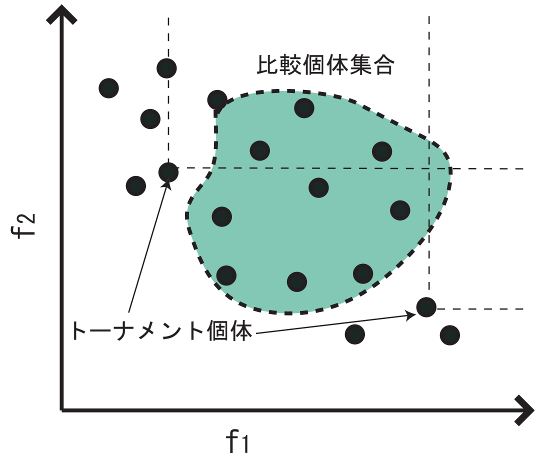

Fig. 7　パレート・トーナメント法

と定義する．$d(x_i,x_j)$ は個体 i,j 間の距離で，その定義としては幾つかの方法が提案されている．シェアリングの適用に関し，表現空間で行うことを提案しているもの，目的関数空間で行うことを提案しているものがあるが，ここでは，個体 i と j の表現型でのユークリッド距離を用いるものを適用する．

また，$s(d)$ はシェアリング関数と呼ばれ，これは，距離 $d(x_i,x_j)$ に応じて適合度を当てる．そして近傍を定めるパラメータ（シェアリング半径）σ>0 を与え，次式に用いる．

$$s(d) = max(1 - \frac{d}{\sigma}, 0) \tag{16}$$

このようにして算出したニッチ数 m_{x_i} でその個体の適合度 $g(i)$ を割り，それを新たな適合度 $g_s(i)$ とする

$$g_s(i) = \frac{g(i)}{m_{x_i}} = \frac{g(i)}{\sum_j s(d(x_i, x_j))} \tag{17}$$

上式により再計算された適合度は，個体間の集中度合いも考慮に入れているため，この適合度を用いた選択を行うことにより個体が均一に分散された状態で次世代に受け継がれるものと考えられる．

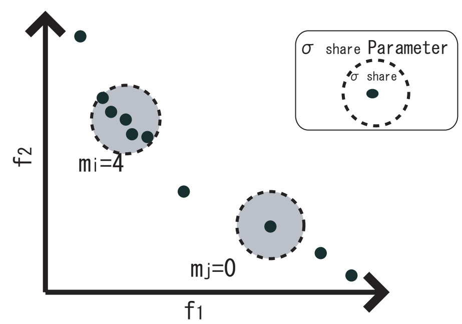

Fig. 8　シェアリング

参考文献

1) 坂和正敏. 離散システムの最適化. 森北出版, 2000.

2) 多目的最適化と再配置操作付き Pareto Simulated Annealing，久保谷寛行