The Science and Engineering Doshisha University, Vol.42, No.3 October 2001

Distributed Cooperation Model of Multi Objective Genetic Algorithms

Tomoyuki Hiroyasu* , Mitsunori Miki* , Tamaki Okuda** and Shinya Watanabe**

(Received June 30, 2001)

In this paper, a new algorithm of Genetic Algorithm for Multi objective Optimization Problems, called Distributed Cooperation model of MOGA (DCMOGA), is proposed. In the proposed algorithm, there are several sub populations. One of them is for finding a Pareto optimum set and the others are for each finding an optimum solution of one of the objectives. These sub populations sometimes exchange their searching information respectively. The proposed algorithm is applied to three types of knapsack test problems. Comparing to the conventional multi objective optimization methods, the proposed model found the better and much widespread Pareto solutions.

Key words ： multi objective optimization problems, distributed genetic algorithms, pareto opitimal solutions

キーワード ： 多目的最適化問題，遺伝的アルゴリズム，パレート最適解

多目的遺伝的アルゴリズムの分散協力型モデル

廣 安 知 之・三 木 光 範・奥 田 環・渡 邉 真 也

1. 序　論

構造最適化問題やレイアウト配置問題，ジョブショップスケジューリング問題といった実問題における意志決定においては，最小もしくは最大を目指す目的が複数存在する場合が多く，これらは多目的最適化問題と呼ばれる．これらの複数の目的がトレードオフの関係にある場合，一意の解を得ることは困難であり，パレート最適解 [1] と呼ばれる解集合の一部を求めることとなる．

多目的最適化問題問題において，パレート解集合を求めることは一つの目標となるが，近年，遺伝的アルゴリズム（GA）を適用する多目的 GA に関する研究が数多く行われている [2–7]．その理由は，GA が多点探索であり，一度の探索で複数のパレート解集合が求まることにある．多目的 GA における代表的な研究は，Schaffer らの VEGA [4] に始まり，パレート最適解集合のフロンティアを明示的に取り扱う Goldberg のランキング法 [2] や Fonseca らの MOGA [3] などがあげられる．また，その他の有効な手法としては，玉置らの並列選択と同時に得られているパレート最適個体を保存する手法の提案，村田らの多目的関数にそれぞれ重みを加え単一目的として解を求める方法 [8] ，Horn らのパレート最適性を考慮したパレート優越トーナメント選択 [6] ，Deb らの Goldberg のランキング法を用いた NSGA [7] などの提案があげられる．

パレート解集合を求める場合，得られた解が目的関数もしくは設計変数空間上の広範囲かつ真のパレート

* Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto
Telephone:+81-774-65-6932, Fax:+81-774-65-6780, E-mail:tomo@is.doshisha.ac.jp , mmiki@mail.doshisha.ac.jp
** Graduate Student, Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto
Telephone:+81-774-65-6716, Fax:+81-774-65-6780, E-mail:elmo@mikilab.doshisha.ac.jp

解付近に求まっていることは重要な要素といえる．広範囲に広がる解を求めるためには，次の 2 つの要素が必要である．すなわち，解が集中するのではなく均等に存在すること，各目的関数を単一目的とした際の最適解が得られていることである．前者においては，シェアリングを利用した Horn の手法 [9]，後者においては，VEGA [4] などの手法がそれにあたる．しかしながら，両社の要求を同時に改善する手法はこれまでに提案されていない．

そこで本研究では，解の広がりを持ったパレート最適解の探索を目的とし，各目的関数の最適解の探索とパレートフロントの前進を同時に行う新たな多目的分散 GA モデルを提案する．提案手法は，多目的 GA を行う個体群と各目的間数における最適解を探索する個体群を用いて解探索を行い，ある一定期間毎に最適解を交換することで協調して解探索を行う手法である．そのため，本提案手法を用いることで，より広範囲に分布し，かつ精度の高いパレート解の探索を期待することができる．また，各目的関数の最適解の探索とパレート解探索を別々に行うよりも計算回数が少なく同精度の解が得られることが期待できる．本研究では幾つかの数値実験例を通して，従来の手法（単一母集団 GA）との比較を行い，提案手法の有効性の検証を行っている．

2. 多目的最適化問題

2.1 多目的最適化問題

多目的最適化問題（Multiobjective Optimization problems: MOP）は，k 個の互いに競合する目的関数 $\boldsymbol{f}(\boldsymbol{x})$ を m 個の不等式制約条件のもとで最小化する問題と定式化される [10]．ベクトル最小化の形式で次のように定式化される．

$$\begin{cases} minimize \quad \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x}))^T \\ subject \quad to \quad \boldsymbol{x} \in X = \{\boldsymbol{x} \in R^n \\ \qquad\qquad\qquad \mid g_i(\boldsymbol{x} \leq 0, j = 1, \ldots, m\} \end{cases} \tag{1}$$

上式における $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$ は n 次元の決定変数のベクトルで，

$$\begin{cases} f_i(\boldsymbol{x}) = f_i(x_1, x_2, \ldots, x_n), i = 1, \ldots, k \\ g_j(\boldsymbol{x}) = g_j(x_1, x_2, \ldots, x_n), j = 1, \ldots, m \end{cases} \tag{2}$$

上式は与えられた n 変数 $x_1, x_2, \ldots, x_n$ の非線形実数値関数で，X は実行可能領域を表す．

多目的最適化問題では，各目的関数がトレードオフの関係にある場合，単一の解を得ることは難しい．そのため，最適解の概念の代わりにパレート最適解の概念が導入されている．

2.2 パレート最適解

$\boldsymbol{x}_1, \boldsymbol{x}_2 \in X$ に対して，$f_i(\boldsymbol{x}_1) \leq f_i(\boldsymbol{x}_2), i = 1, \ldots, k$ で，しかも，ある j について $f_i(\boldsymbol{x}_1) < f_i(\boldsymbol{x}_2)$ であれば，$\boldsymbol{x}_1$ は $\boldsymbol{x}_2$ に支配されないという．さらにある $\boldsymbol{x}^*$ に支配されないような $\boldsymbol{x} \in X$ が存在しないとき，$\boldsymbol{x}^*$ をパレート最適解と呼ぶ．

Fig. 1 に，2 目的 ($k = 2$) の場合のパレート最適解の例を示す．図中の太線がパレート最適解を，破線が弱パレート最適解を示している．

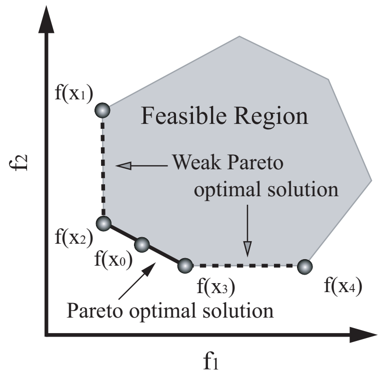

Fig. 1. Pareto Optimal Solution and Feasible Region.

定義により，最適解が存在するときには，それがパレート最適解であり，それ以外のパレート最適解は存在しない．したがって，パレート最適解は多目的最適化問題に対する最も合理的な解（集合）であるといえる．

3. 多目的遺伝的アルゴリズム

3.1 多目的遺伝的アルゴリズム

一般に，多目的最適化問題において求めるパレート最適解は 1 つではなく集合となる．遺伝的アルゴリズム (Genetic Algorithum:GA) を適用することによってパレート最適集合を一度の探索で求めることができる．これは，GA が複数の個体を用いて解探索を進めるため，複数の目的関数を明示的に取り扱うことができ，目的間のトレードオフ関係を明示的にしながらパレート最適解の探索を行えるためである．このことから，GA を多目的最適化問題に適用する多目的 GA の研究が数多くなされている [2–5]．多目的最適化問題（2 目的）における GA による解探索の様子を Fig. 2 に示す．

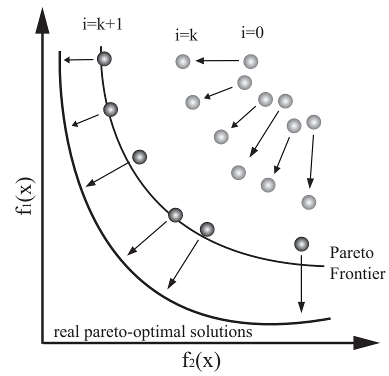

Fig. 2. The concept of searching mechanism.

Fonseca らは種々提案されている手法を以下のように分類した [11]．

- 非パレート的アプローチ
- パレート的アプローチ

前者は，それぞれの目的関数について独立に選択を行う．代表的な手法としてベクトル評価遺伝的アルゴリズム (Vector Evaluated Genetic Algorthms: VEGA) がある [4]．それに対し後者は，パレート最適性を明示的に扱う．代表的な手法には，ランキング選択やトーナメント選択によるものなどがある．

3.2 非パレート的アプローチ -VEGA-

Schaffer は，ベクトル評価遺伝的アルゴリズム (Vector Evaluated Genetic Algorthms: VEGA) と呼ばれる手法を提案した [4]．

VEGA は，Fig. 3 に示すように，個体集合を目的関数の数に等しい部分個体集合に分割し，各目的関数値に応じて独立に個体を選択してそれぞれの部分個体集合を生成する．そして，生成された部分個体集合をすべて合わせて一つの個体集合としたものに対して交叉，突然変異を行う手法である．

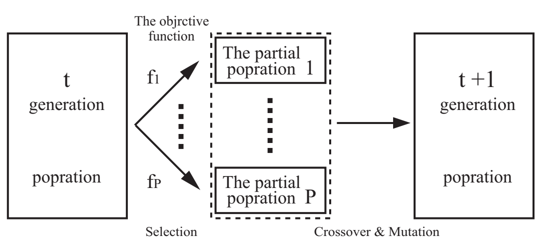

Fig. 3. VEGA.

しかし，ある一つの目的関数値のみが極端に良い解が生成されやすく，すべての目的関数がバランスよく達成されているという意味での妥協解が得られ難いという問題がある．

3.3 パレート的アプローチ -ランキング法-

解の優劣関係に基づいて定められるランクとして適応度関数を作り，これにより選択を行う手法である．この手法はパレート的アプローチであり，Goldberg により提案された．

各個体のランクは次のように決定される．個体集合中で他に優越されない個体をパレート最適個体とし，個体集合の中からパレート最適個体を求め，これらのランクを $r = 1$ とする．次に，得られたパレート最適個体を個体集合から除き，$r = r+1$ とする．この手続きを個体すべてのランクが決定されるまで繰り返し，ランクが決定される．

この手法に類似した手法として Fonseca らは，個体 X が n_X 個の個体に優越してるときに，X のランク r_x を

$$r_X = 1 + n_X \tag{3}$$

のように定める，ランクの決定法を提案している．このランキング法の適用例を図 4 に示す．この手法では，Goldberg のランキング法で区別できなかった個体のランキングが可能になる．

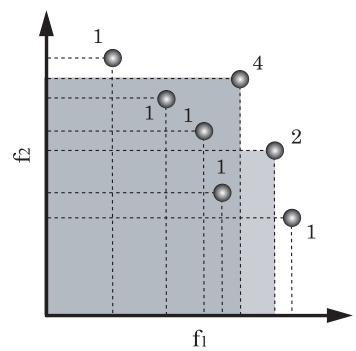

Fig. 4. Ranking Method (Fonseca).

本論文では，Fonseca らの提案するランキング法により個体にランク付けを行い，このランクを用いたルーレット選択によって個体を選択している．

3.4 パレート保存戦略

パレート保存戦略では，原則として個体集合中のパレート最適個体を選択し，次世代へ残す手法であり [12]

これは，単目的の問題におけるエリート保存戦略 [2] に対応するものである．

3.5 シェアリング

多目的最適化問題では，多様性を維持し，より広範囲にかつ均等に分布するパレート最適解集合を求めることは重要である．Horn らは，シェアリングを利用した有効な手法を提案している [6, 9]．

まず，各個体 x について，その個体の近傍がどの程度込み合っているかをニッチ数 (niche count)m_{x_i} として計算する．ニッチ数については一般的に

$$m_{x_i} = \sum_{j=1}^{N} s(d(x_i, x_j)) \tag{4}$$

と定義される．ここで，$d(x_i, y_i)$ は，個体 x_i と x_j との距離で，その定義としては幾つかの方法が提案されている．本研究では，個体 x_i と x_j の表現型でのユークリッド距離を用いるものとする．

また，$s(d)$ は，シェアリング関数 (sharing function) と呼ばれ，距離 d について単調減少関数である．$s(d)$ としてニッチの大きさを表すパラメータ σ_{share} をあらかじめ与えておくものとし，次式を用いる．

$$s(d) = \max\{0, 1 - \frac{d}{\sigma_{share}}\} \tag{5}$$

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．

$$g_s(i) = \frac{g(i)}{m_{x_i}} = \frac{g(i)}{\sum_{j} s(d(x_i, x_j)} \tag{6}$$

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．

$$D = \sqrt{(x_i^{max} - x_i^{min})^2 + (x_j^{max} - x_j^{min})^2} \tag{7}$$

シェアリングレンジというパラメータを導入し，求めた最大距離 D をこのパラメータで割ることにより，シェアリングパラメータ σ_{share} を求める．

本研究においてシェアリングは，個体数の膨張を防ぐ目的で用いる．そのため，シェアリングは毎世代行うのではなく，個体数がある一定数（limit population size）を越えた場合のみ適用する．具体的には，シェアリングを用いて計算した適合度より，ルーレット選択を用いて任意の選択個体数（select population size）を選択するというものである．Fig. 5 に 2 次元空間におけるシェアリングの概念図を示す．

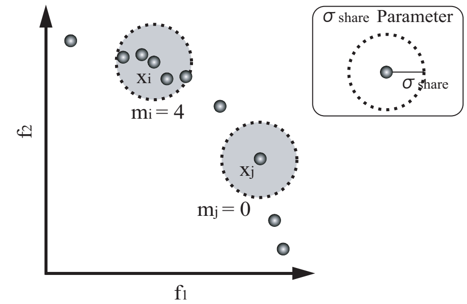

Fig. 5. Sharing.

3.6 パレート解の評価方法

得られたパレート最適解に対する評価方法は，適用したモデルの性能を評価する上でも不可欠である．しかし，多目的最適化の分野において，幾つかの評価方法は提案されている [13, 14] ものの未だ確立には至っていない．これは，単一目的の場合と異なり，多目的では解の評価のポイントとして，

- 真のパレート解へどれだけ近づいているか（精度）
- 真のパレート解のどの程度を被覆しているか（解の広がり）

の両方を考慮して判断する必要があり，この 2 つのポイントを適切にかつ定量的に判断するような評価方法は非常に難しいためである．特に，真のパレート最適解が未知であった場合にはこれらの評価項目を適切に評価することは非常に困難である．

本研究では，比屋根 [13] が提案している定量的な評価方法を参考に，幾つかの評価項目を設けて総合的に解の評価を行った．具体的な評価項目について以下で説明する．

3.6.1 被覆率 (Cover Rate)

パレート解を探索する場合，得られた解が真のパレート解上の 1 点に集中していては解集合が十分とは言えない．そのため，解の広がりを示す指標が必要となる．被覆率とは，いかに真のパレート解を隙間なく詳細に求めているかを評価する基準である．被覆率の概念図を Fig. 6 に示す．

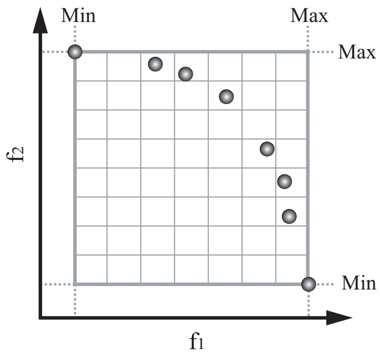

Fig. 6.　Cover Rate.

被覆率は，各目的関数の最大値および最小値を検索し，その領域を任意の数で分割する．それぞれ分割された領域の中に解が存在する場合は 1，存在しない場合には 0 とする．本研究では，これらの数値を合計し，領域の数で割ったものを被覆率として用いている．そのため，この被覆率が 1 に近い程すべての領域に解が存在していることになり，解が集中することなく全体に行き渡っていると解釈することができる．本研究の数値計算では分割された領域の数を 100 としている．また，真のパレート解が既知でない場合には，経験的に得られたパレート解を基準に各目的関数値の最大値，最小値を用いて被覆率を計算している．

4.　多目的 GA と単一目的 GA の分散協力型モデル

4.1　DCMOGA

本研究では，分散協力型モデル（Distributed Cooperation model of MOGA and SGA：DCMOGA）の提案を行う．DCMOGA では多目的 GA を行う従来の個体群（MOGA 個体群）とは別に，各目的間数における最適値を探索する個体群（SGA 個体群）を用いてパレート最適解の探索を行う．すなわち，目的関数数+1 の個体群が存在することとなる．以下に提案手法のアルゴリズムを示す．

1. 各個体群でそれぞれの個体数分の個体をランダムに発生させる．
2. 個体情報の交換（移住）間隔として，一定の評価回数を定め，その評価計算回数を各個体群に均一に分配する．
3. 与えられた評価計算回数まで，各個体群が独立して探索を行う．
4. MOGA 個体群が各 SGA 個体群と個体情報を交換する．(移住)

   $$MOGA \text{個体群} \quad \longleftrightarrow \quad SGA \text{個体群} \ (F_i)$$

   - 目的関数 F_i の探索を行う SGA 個体群が，個体群内の最適解 I_S を MOGA 個体群に送信する．
   - MOGA 個体群は，群内で F_i の最良値を持つ最適解 I_M を送信する．
5. 目的関数 F_i において，各個体群の最適解である I_S と I_M を比較し，次回の移住までの各個体群の評価計算回数を決定する．

   $I_S > I_M$　SGA 個体群の評価計算回数を減少させ，MOGA 個体群の評価計算回数を増加させる．

   $I_M \geq I_S$　上記を逆の操作を行う．
6. 現段階までの各個体群の評価計算回数を合計し，終了評価計算回数と比較する．終了評価計算回数に満たなければ，3. に戻る．

この手法の 2 目的の場合の概念図を Fig. 7 に示す．

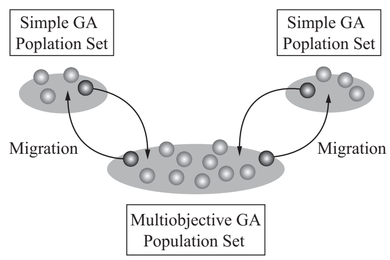

Fig. 7.　Migration in DCMOGA.

以上のように DCMOGA では，SGA 個体群を用いて各目的関数における最適解を探索を行っている．そのため，探索の各段階で，それぞれの目的関数の最良解を得ることができる．この最良解は，多目的 GA では問題が複雑になるにつれて得られにくいものであり，パレートフロンティアにおける両端部分が得られるということは重要である．

さらに，この最良解を探索途中で MOGA 個体群に移住させることで，MOGA 個体群は各目的関数における最良解を知ことができる．この最良解がパレート最適解の探索に大きく影響し，その結果，広範囲に分布するパレート最適解を得られることができると考えられる．

また，MOGA 個体群における各目的関数についての最良個体を SGA 個体群に移住させる．それより，MOGA 個体群が SGA 個体群より良い最良解を見つけている場合，SGA 個体群がさらに良い最良解を知り，探索がより早く進んでいくと考えられる．

このように，それぞれの個体群が最良解を交換することで，各目的関数における最良解がパレート最適解の多様性を維持し，広範囲に及ぶパレート最適解を得ることできると考える．

5. 数値実験

5.1 対象問題

本研究では，提案する DCMOGA を実際に幾つかの対象問題に適用し，従来手法との比較を通じて DC-MOGA の有効性の検証を行う．対象問題として，荷物数の異なる 3 つの多目的 0/1 ナップサック問題を用いた [14, 15]．

5.1.1 多目的 0/1 ナップサック問題

0/1 ナップサック問題は，ナップサックと荷物（item）のセットから成り立っている．各荷物には重さと利益が付随し，上限制約としてナップサックの容量がある．この問題の目的は，ナップサックの許容量内で，荷物全体を総和した利益が最大になるような荷物の組み合わせを見つけることである．

この単一目的問題は，ナップサックの数および付随する荷物のセットを複数にすることによって多目的問題へ拡張することができる．多目的 0/1 ナップサック問題は，多目的における多くの研究に用いられている代表的なテスト関数の 1 つであり，特に離散的な問題における良質なテスト関数として知られている．[14, 15]

多目的 0/1 ナップサック問題は，次のような制約条件を満たし，

$$\forall \in 1, 2, \ldots, n : \sum_{j=1}^{m} w_{i,j} \cdot x_j \leq c_i \qquad (8)$$

次式で表される $\boldsymbol{f}(\boldsymbol{x})$ の最大化するような設計変数 $\boldsymbol{x} = (x_1, x_2, \cdots, x_m) \in 0, 1^m$（$m$ は任意の正の整数）を求める．

$$\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x})) \qquad (9)$$

$$f_i(\boldsymbol{x}) = \sum_{j=1}^{m} p_{i,j} \cdot x_j \qquad (10)$$

（尚，上式において $x_j = 1$ ならば荷物 j は選択された荷物であることを意味する）

各変数は，次のように定義されている．

$$p_{i,j} = \text{ナップサック } i \text{ の荷物 } j \text{ の利益} \qquad (11)$$

$$w_{i,j} = \text{ナップサック } i \text{ の荷物 } j \text{ の荷物の重さ} \qquad (12)$$

$$c_i = \text{ナップサック } i \text{ の許容重量} \qquad (13)$$

本実験では，対象問題として 100 荷物 2 目的，250 荷物 2 目的，750 荷物 2 目的の 3 種類の多目的 0/1 ナップサック問題を用いている．

5.2 多目的 GA の構成

5.2.1 制約外への個体の対応

ナップサック問題では重量の制約があるため，この重量を超えるような目的関数値を持つ個体に対して何らかの対処が必要となる．

本数値実験では，初期個体発生時において定義域外に存在する個体は認めず，制約を満たす個体数が必要個体数となるまで，個体の生成を行うものとする．しかし，交叉もしくは突然変異といった GA オペレータにより個体が定義域を越える場合もある．このような場合，制約を満たしていない個体の詰め込むべき荷物を，ランダムに 1 つずつ減らすという方法を用いた．

5.2.2 GA パラメータ

数値計算で使用したそれぞれの手法における GA パラメータは，交叉率を 1.0，突然変異率を 0.01 とする．提案手法における多目的 GA 個体群でのパラメータは従来の手法でのパラメータと同様の値を用い，SGA 個体群におけるパラメータは，交叉率 1.0，突然変異率 1/L（L:染色体長）とする．

多目的 GA において，用いる個体数，シェアリング半径や終了条件は，得られるパレート最適解の精度に影響すると考えられる．これらのパラメータは多くの手法で解に大きく影響することが報告されている [16]．さらに，最適な値を求めることは非常に難しい．そこで，本数値実験では，各 SGA 個体群の個体数を 4 に固定し，MOGA 個体群の個体数を 100，200，400 とした．また，シェアリングレンジとして各個体数を使用し，終了条件として評価回数を 180，360，720 万回と設定する．

5.3 数値実験結果

従来の手法（以下：MOGA）として，パレートランキング法で各個体にランクを付け，パレート保存戦略

Table 1. Cover Rate in Knapsack Problems.

items	Population size	DCMOGA	MOGA
100	100	0.37	0.41
	200	0.50	0.60
	400	0.54	0.61
250	100	0.36	0.37
	200	0.58	0.54
	400	0.65	0.68
750	100	0.33	0.29
	200	0.56	0.45
	400	0.77	0.56

で選択する手法を用いている．この MOGA と提案手法である DCMOGA を用いて実験を行った．

10 試行平均の被覆率を Table 1 と Fig. 14 に示す．また，10 試行のうちパレート最適解の被覆率が 10 試行平均に最も近い値を持つパレート最適解集合のプロット図を問題別に Fig. 8，Fig. 9，Fig. 10 に示す．

DCMOGA では，各個体群の評価回数が移住間隔毎に変化する．個体の探索における MOGA 個体群の評価回数が 10 試行の平均値に最も近い結果を持つ評価回数推移のプロット図を Fig. 11，Fig. 12，Fig. 13 に示す．

5.4 提案モデルと従来のモデルとの比較

得られたパレート最適解の広がりとパレート最適解の進行状況について考察を行う．

- Fig. 8 からわかるように，item 数が 100 の場合では，問題があまり複雑ではないため，どちらの手法においても同等の精度のパレート最適解集合を得られている．しかし，個体数が少ない場合には，DCMOGA を用いた場合に，より広範囲に分布したパレート最適解を得ている．

- 少し問題が複雑になった item 数 250 の場合，Fig. 9 からわかるように，MOGA により得たパレート最適解は，パレートフロンティアの中央部分に偏っている．個体数の増加に伴い，パレート最適解がカバーする領域も増加するが，DCMOGA を用いた場合には，より広範囲に分布したパレート最適解を得ることができ，さらに，パレートフロンティアの全体を把握することができる．パレート最適解の精度を比較すると，ほぼ同等の結果と

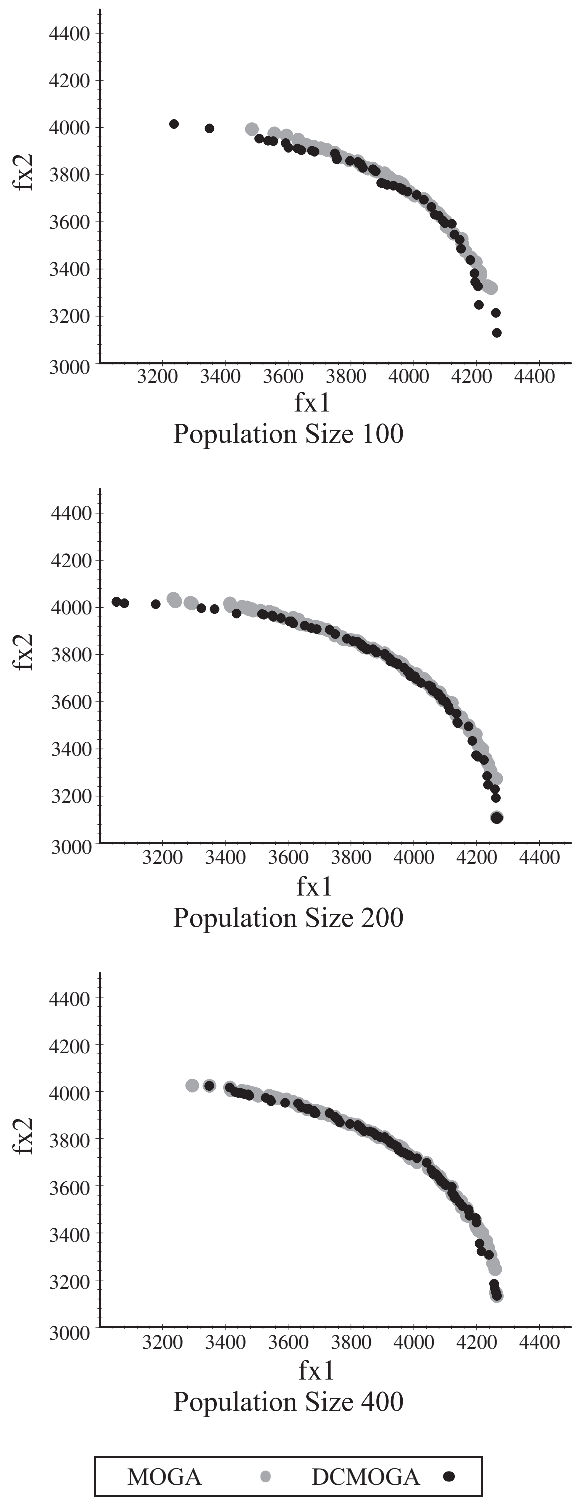

Fig. 8. Pareto Solutions (100 items).

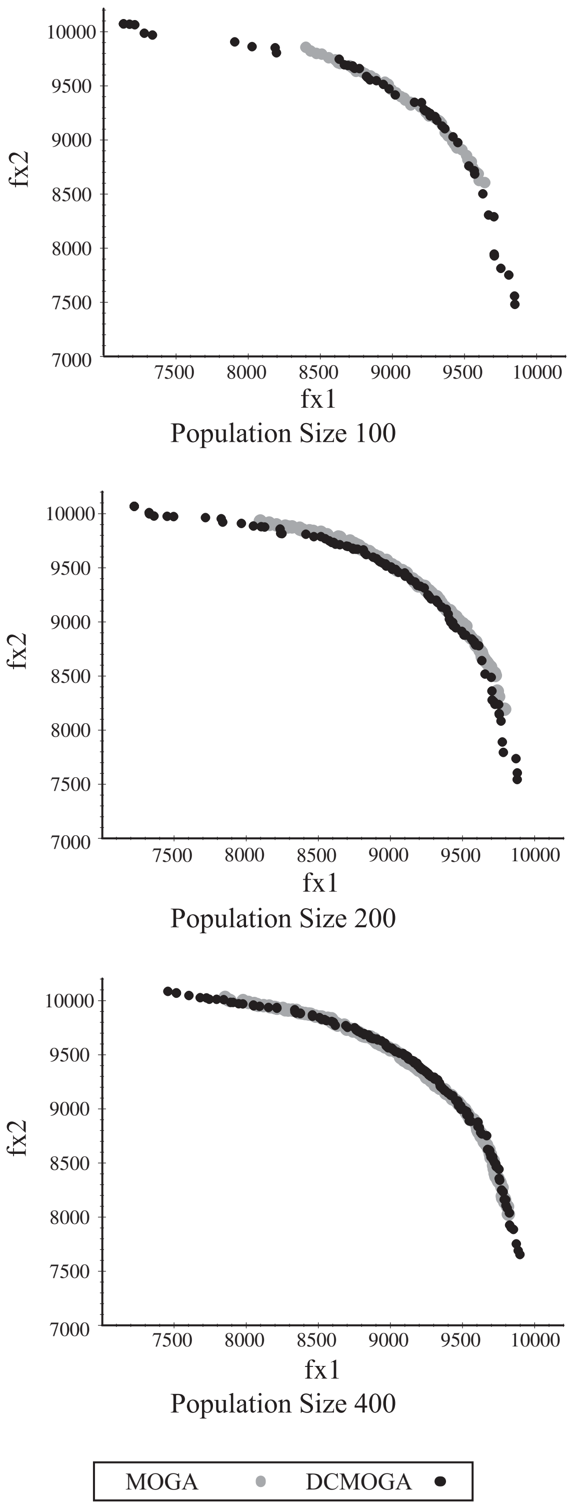

Fig. 9. Pareto Solutions (250 items).

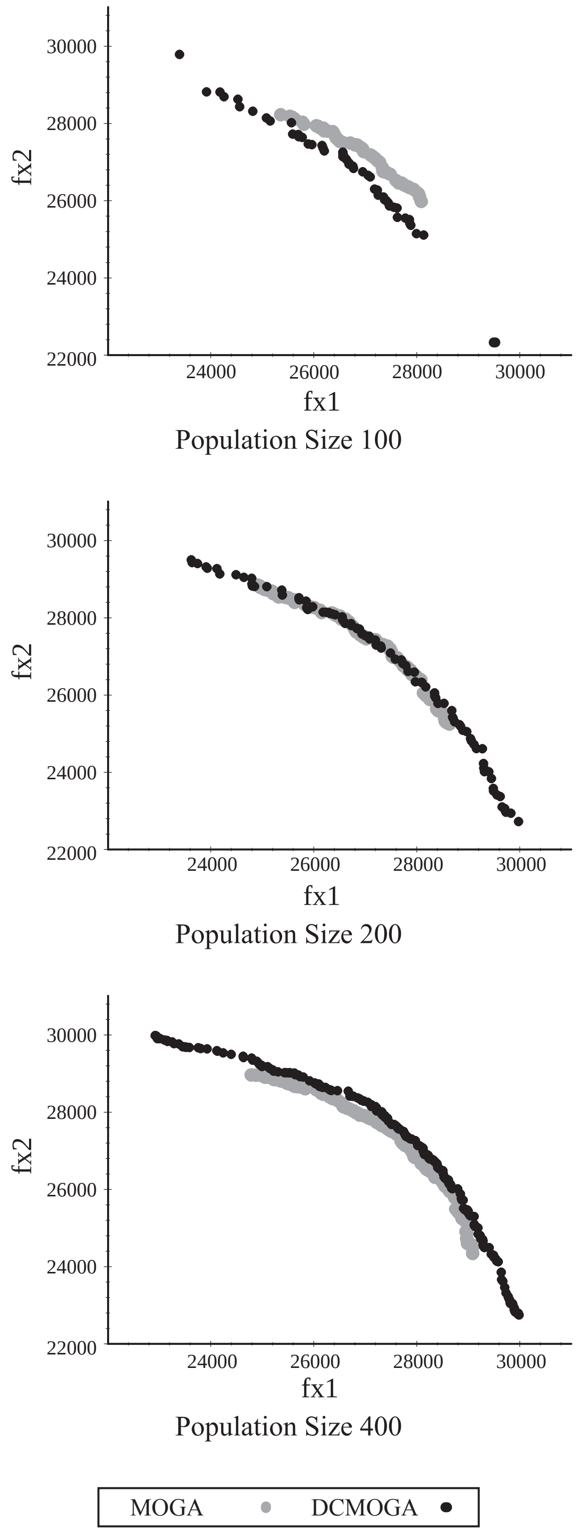

Fig. 10. Pareto Solutions (750 items).

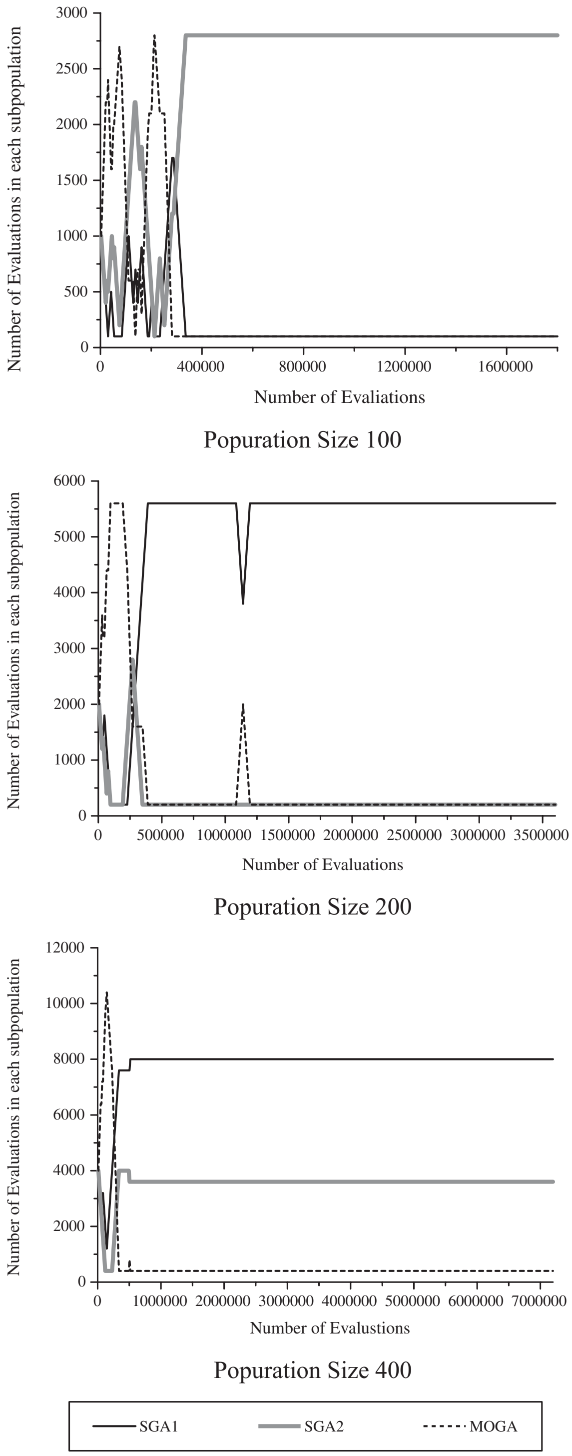

Fig. 11.　Number of Evaluations (100 items).

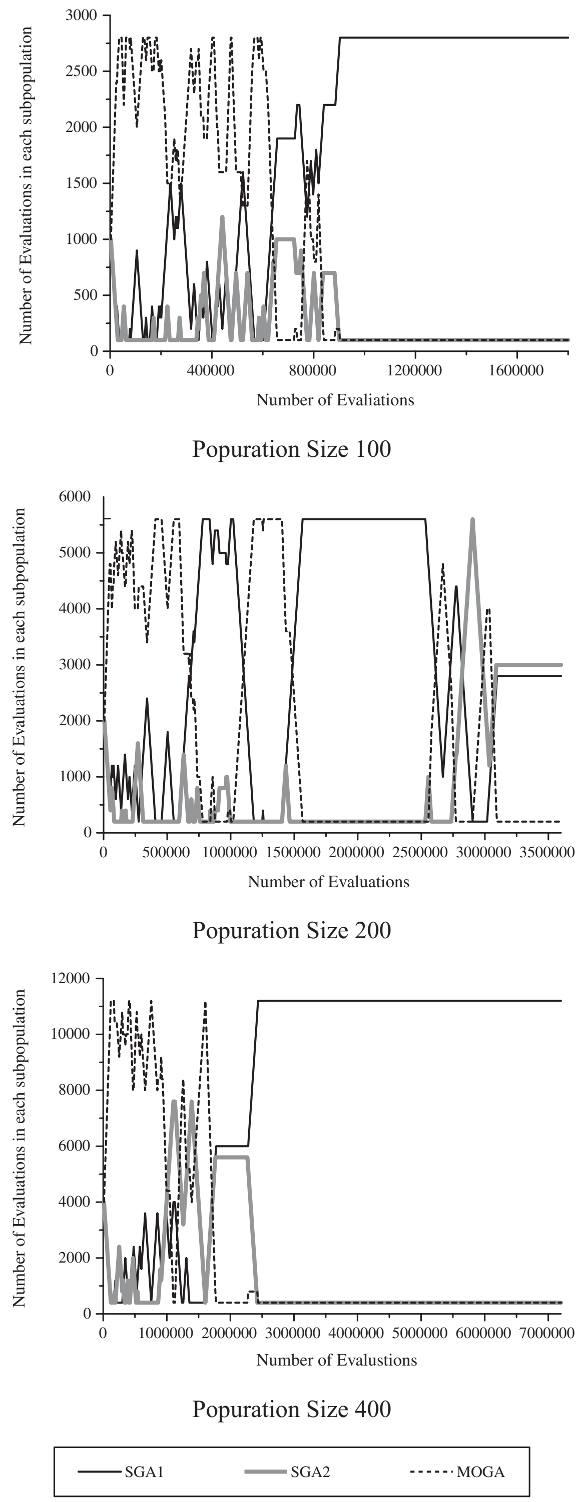

Fig. 12.　Number of Evaluations (250 items).

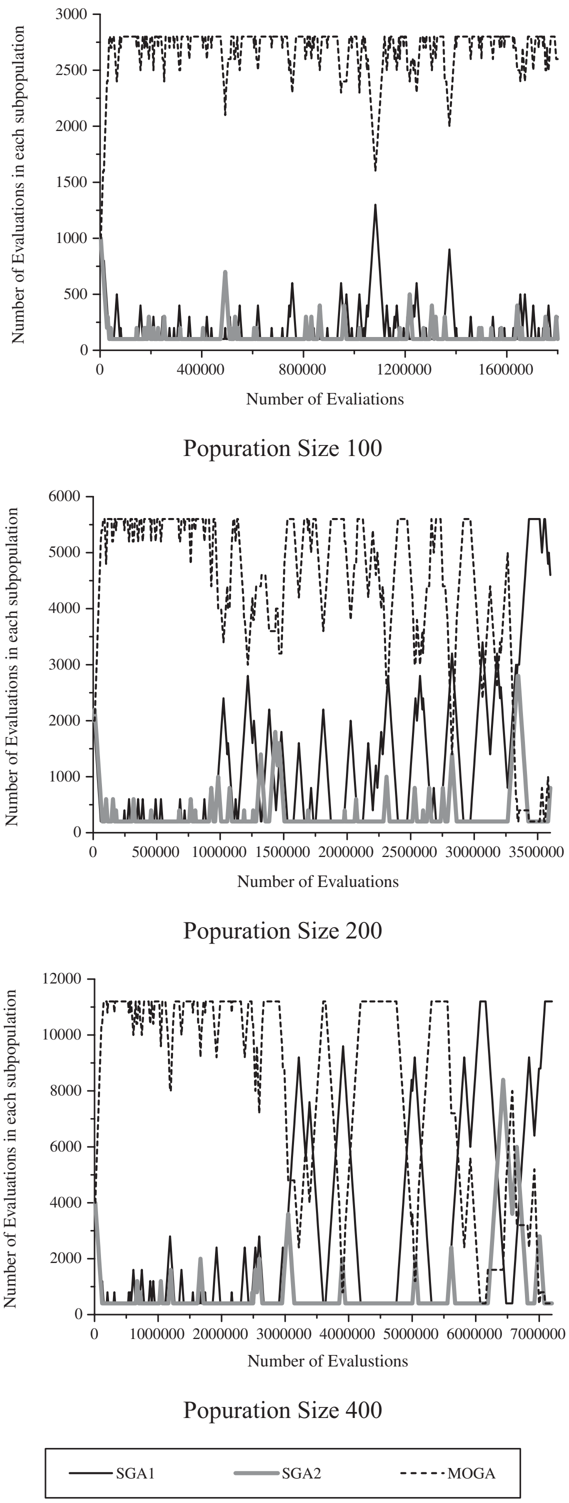

Fig. 13.　Number of Evaluations (750 items).

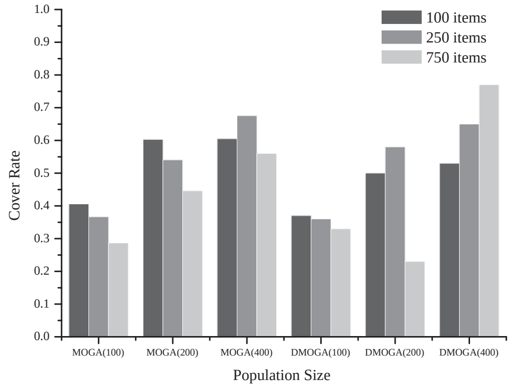

Fig. 14.　Cover Rate in Knapsack Problems.

なっている．

- かなり複雑な問題である，item 数が 750 の場合には，精度においても差が見られることが，Fig. 10 からわかる．個体数が少ない場合，MOGA を用いた方がより精度が高いパレート最適解を得ることができる．DCMOGA では，パレートフロンティアの両端部分を把握することはできるが，パレートフロンティア全体を把握できる程度には，解探索が進んでいない．個体数が増加することで，MOGA では item 数 250 の場合と同様に，パレート最適解がカバーするパレートフロンティアの領域は増加しているが，DCMOGA と比較するとかなりの差が見られる．また，DCMOGA では，パレートフロンティアをある程度推測することができる．

MOGA による探索では，1 つの目的関数の最適解の探索が，問題の難度が上がるにつれて困難になっている．これは可能領域が広範囲に及ぶため，多様性を維持した解探索が懇談であると考えられる．

それに対し DCMOGA では，SGA 個体群を用いて各目的関数の最適解を探索を行い，この結果を MOGA 個体群の探索に取り入れているため，広範囲に分布するパレート最適解を得ることが可能になる．

精度においても，DCMOGA は MOGA とほぼ同等の結果を得ている．一定の評価計算回数を終了条件としているため，両手法での合計評価計算回数は同じである．しかし，DCMOGA では，SGA 個体群を用いて探索を行っているため，評価計算回数の一部が SGA 個体群の評価計算として使用される．このため，

MOGA 個体群での行われている評価計算回数は，従来の MOGA と比較すると少なくなる．それにもかかわらず，精度面で同等の結果が得られたということは，個体群における移住が，パレート最適解の進行（解の精度）にも良い影響を及ぼしていると考えられる．

5.5 各個体群における評価回数の推移

提案手法における各個体群の評価回数の推移について考察を行う，

- Fig. 11，Fig. 12，Fig. 13 からわかるように，探索の初期段階においては，評価計算のほとんどが MOGA 個体群で行われている．これは，探索の初期段階では，各 SGA 個体群で得られている各目的関数の最良値が，MOGA 個体群の値よりも良いためである．
- 問題がかなり複雑になった item 数 750 の場合には，Fig. 13 からわかるように，解探索の終了時まで，MOGA 個体群での評価計算の多い状態が続いている．特に個体数が少ない場合には，この傾向が顕著に現れている．Fig. 10 を見ると，個体数 100 の場合には，まだ解探索が十分に進んでいないことがわかる．すなわち，MOGA 個体群である程度の解探索が進むと，評価計算は，SGA 個体群へと移行されていくが，問題が複雑になり，MOGA 個体群での解探索が進まないと，SGA 個体群へと評価計算が移行されない．
- 解探索後半に，SGA 個体群の評価計算が減少し，MOGA 個体群の評価計算が増加することが，Fig. 11，Fig. 12 の個体数 200 の場合からわかる．これは，SGA 個体群が探索後半で，より良い最適解を得ると，MOGA 個体群でさらに良いパレート最適解を探索するため，MOGA 個体群の評価計算が増加する．

DCMOGA では，探索の初期段階では MOGA 個体群における評価計算回数が増加している．その後，MOGA 個体群である程度まで，パレート最適解の探索が進むと，ほとんどの評価計算が SGA 個体群で行われる．このことから，DCMOGA ではこの切り替わる時点で，MOGA 個体群の探索がある程度進んでいると考えられる．

6. 結　論

本論文では，多目的 GA の新たな手法として，多目的 GA と単一目的 GA の分散協力型モデル（DCMOGA）を提案し，その有効性を検証した．

本手法は，広範囲に分布するパレート最適解の探索を目的とし，パレートフロントの前進と各最適解の更新とを同時に行う分散モデルである．0/1 多目的ナップサック問題に対して従来の MOGA と DCMOGA を適用した結果，以下の結論が得られた．

- 従来の MOGA では，問題が複雑な場合，また用いる個体数が少ない場合に，広範囲に分布するパレート最適解を得ることが困難であった．これは，探索領域が広範囲に及ぶため，探索途中に多様性が失われているからである．
- 提案した DCMOGA では，従来の手法と比較し，精度においてはほぼ同等，解の幅広さではそれ以上の結果を得ることができた．これは，パレート最適解の探索に MOGA 個体群と SGA 個体群を用いたことが有効に作用したと考えられる．すなわち，SGA 個体群で得られた最適解を MOGA 個体群に移住させることにより，MOGA 個体群で得られるパレート最適解が幅広く分布することが可能になった．
- DCMOGA では，各個体群の評価計算回数を適応的に増減させることで，各個体群が協調した良好な解探索が行われている．

本研究は，文部省学術フロンティア推進事業に基づく同志社大学学術フロンティア研究プロジェクト「知能情報科学とその応用」の一環として行われた．ここに関係各位に謝意を表する．

参考文献

1) 西川, 三宮, 茨木. ”最適化　第 4 章.” 岩波書店, 1982.

2) D. E. Goldberg. ”*Genetic Algorithms in search, optimization and machine learning.*” Addison-Wesly, 1989.

3) C. M. Fonseca and P. J. Fleming. ”Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization.” In *Proceedings of the 5th international coference on genetic algorithms*, pp. 416–423, 1993.

4) J. D. Schaffer. "Multiple objective optimization with vector evaluated genetic algorithms." In *Proceedings of 1st International Conference on Genetic Algorithms and Their Applications*, pp. 93–100, 1985.

5) 廣安知之, 三木光範, 渡邉真也. "領域分割型多目的遺伝的アルゴリズム." 情報処理学会論文誌, Vol. 41, pp. 79–89, 2000.

6) Alex Mayer Mark Erickson and Jeffrey Horn. "The niched pareto genetic algorithm 2 applied to the design of groundwater remediation systems." *First International Conference on Evolutionary Multi-Criterion Optimization*, No. 1993, pp. 681–695, 2001.

7) Kalyanmoy Deb and Tushar Goel. "Controlled elitist non-dominated sorting genetic algorithms for better convergence." *First International Conference on Evolutionary Multi-Criterion Optimization*, No. 1993, pp. 67–81, 2001.

8) T. Murata and H. Ishibuchi. "Moga: Multi-objective genetic algorithms." In *Proceedings of the 2nd IEEE International Conference on Evolutionary Computing*, pp. 289–294, 1995.

9) N. Nafpliotis J. Horn and D. E. Goldberg. "A niched pareto genetic algorithm for multiobjective optimization." In *Proceedings of 1st ICGA*, pp. 82–87, 1994.

10) 坂和 正敏 田中雅博. "遺伝的アルゴリズム." 朝倉書店, 1997.

11) C. M. Fonseca and p. J. Fleming. "An overview of evolutionary algorithms in multiobjective optimization." *Evolutionary Computation*, Vol. 3, No. 6, pp. 1–16, 1995.

12) 北野宏明. "遺伝的アルゴリズム 2." 産業図書, 1995.

13) 比屋根. "並列遺伝的アルゴリズムによる多目的最適化問題のパレート最適解集合の生成法と定量的評価法." 第 9 回自律分散システムシンポジウム, pp. 295–300, 1997.

14) E. Zitzler and L. Thiele. "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach." *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 4, pp. 257–271, 1999.

15) E. Zitzler and L. Thiele. "Multiobjective optimization using evolutionary algorithms - a comparative case study." *Parallel Problem Solving from Nature - PPSN-V*, pp. 292–301, 1998.

16) C. A. Coello. "An updated survey of evolutionary multiobjective optimization techiniques: State of the art and future trends." In *Proceedings of Congress on Evolutionary Computation*, pp. 1–11, 1999.