

1-103 進化型多目的最適化手法と局所探索とのハイブリッド化

Hybridization of Evolutionary Multiobjective Optimization Algorithms with Local Search

○石渕久生 (大阪府立大学) 吉田匡志 (大阪府立大学)

Hisao Ishibuchi, Dept. of Industrial Engineering, Osaka Prefecture Univ., Sakai, Osaka 599-8531, Japan
Tadashi Yoshida, Osaka Prefecture University

In this paper, we discuss the hybridization of evolutionary multiobjective optimization (EMO) algorithms with local search. First we show that the main positive effect of the hybridization is the improvement in the convergence speed to the Pareto-front through computational experiments using our multiobjective genetic local search algorithm. Next we show that the main negative effect is the deterioration in the diversity of obtained solutions. Then we demonstrate that local search can be easily combined with recently developed EMO algorithms such as SPEA and NSGA-II. Their search ability can be improved by the hybridization with local search. Finally we point out the importance of striking a balance between genetic search and local search.

Key Words: Multiobjective optimization, Evolutionary computation, Hybrid algorithms, Memetic algorithms, Flowshop scheduling

1. はじめに

近年, 進化計算を多目的最適化問題に適用する進化型多目的最適化に関する研究⁽¹⁾⁻⁽⁴⁾が注目を集め, この分野に焦点を絞った国際会議が開催されるまでに至っている⁽⁵⁾⁻⁽⁶⁾. 進化型多目的最適化では, パレート最適解への収束性および獲得された解集合の多様性を同時に実現することが重要である. 進化型多目的最適化に関する初期の研究では, 目的関数空間内の広い範囲に分布した多様なパレート最適解集合を得るために, 多様性の維持に重点が置かれていた⁽⁷⁾⁻⁽¹⁰⁾. 一方, 最近の研究では, 多様性の維持と同様にパレート最適解への収束性にも重点が置かれ, アルゴリズムの実行中に得られた暫定的な非劣解集合を保存するエリート保存戦略の重要性が認識されてきている⁽¹¹⁾⁻⁽¹⁴⁾. パレート最適解への収束性を向上させるための有力な戦略の一つは, 進化型多目的最適化手法と局所探索とのハイブリッド化である. 進化計算手法と局所探索とのハイブリッドアルゴリズムは Memetic Algorithm⁽¹⁵⁾と呼ばれる研究分野を構成し, 単一目的最適化に関しては多くの研究成果が報告されている⁽¹⁶⁾⁻⁽¹⁸⁾. 実際, 現実問題へ応用された進化計算手法の多くが局所探索とのハイブリッドアルゴリズムであると言われている⁽¹⁹⁾. 筆者らは, 多目的最適化問題に対する最初の Memetic Algorithm である多目的遺伝的局所探索 (MOGLS: Multi-Objective Genetic Local Search) の提案を行った⁽²⁰⁾. Jaszkiwicz⁽²¹⁾は親個体の選択方法を修正することで筆者らの MOGLS の改良を行った. また, Knowles & Corne⁽²²⁾は局所探索型の多目的最適化手

法である PAES⁽¹²⁾に交叉操作を導入することで Memetic PAES (M-PAES) を提案した. MOGLS と M-PAES の比較は, Knowles & Corne⁽²³⁾および Jaszkiwicz⁽²⁴⁾で行われている.

進化型多目的最適化手法に局所探索を組み込むことの最大の利点は, パレート最適解への収束性の向上である. これは, 単一目的最適化問題に対するハイブリッド化が最適解への収束性を向上させることに対応している. 一方, ハイブリッド化の問題点は, 世代更新に必要な計算時間の増加である. 一般に, アルゴリズムの実行は限定された計算時間内で行われるため, 世代更新に必要な計算時間の増加は, 世代数の減少という結果になる. このため, 進化型多目的最適化手法の持つ大域的探索能力が十分に発揮されず, 最終的に獲得される解集合の多様性が減少する. これは, 目的関数空間内の広い範囲に分布した多様なパレート最適解の獲得を目的とした進化型多目的最適化において大きな問題となる. 一方, 単一目的最適化問題では, 通常の場合, 最終的に獲得される解集合すなわち最終世代の個体群の多様性は重要ではない.

本研究では, 局所探索とのハイブリッド化が進化型多目的最適化手法の探索能力に及ぼす影響を考察する. まず, 良好な個体にのみ確率的に局所探索を適用することで MOGLS の性能が大きく改良されることを示す⁽²⁵⁾. 次に, 改良された MOGLS を用いた数値実験により, ハイブリッド化がパレート最適解への収束性を向上させることを示す. 同時に, ハイブリッド化が獲得される解

集合の多様性を低下させることも示す。さらに、高い探索能力を持つことが知られている進化型多目的最適化手法である SPEA⁽¹¹⁾および NSGA-II⁽¹⁴⁾に局所探索を組み込む方法を示す。SPEA および NSGA-II のハイブリッドアルゴリズムを用いた数値実験により、これらのハイブリッドアルゴリズムが SPEA や NSGA-II よりも高い探索能力を持つことを示す。最後に、ハイブリッドアルゴリズムにおける遺伝的探索と局所探索とのバランスの重要性を明らかにする。

2. 多目的遺伝的局所探索アルゴリズムの改良

本研究で用いる多目的遺伝的局所探索アルゴリズムの概略図を図1に示す。まず初期個体群が生成され、進化型多目的最適化手法で新しい個体群に更新される。更新された新しい個体群に対して局所探索が適用され、個体群の改良が行われる。さらに、改良された個体群に進化型多目的最適化手法が適用され、新しい個体群が生成される。このように、ハイブリッドアルゴリズムでは、進化型多目的最適化手法による新しい個体群の生成と局所探索による個体群の改良が繰り返される。

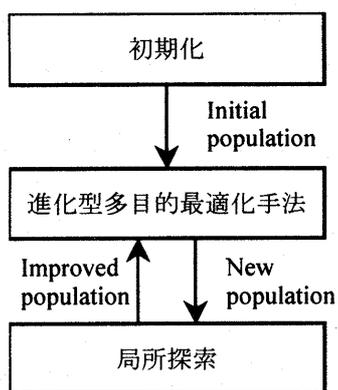


Fig. 1 Generic form of hybrid algorithms.

2.1 従来の MOGLS アルゴリズム

以下のような N 目的最小化問題を用いて、筆者らによる従来の MOGLS⁽²⁰⁾を説明する。

$$\text{Minimize } \mathbf{z} = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_N(\mathbf{x})), \quad (1)$$

$$\text{subject to } \mathbf{x} \in \mathbf{X}. \quad (2)$$

ここで \mathbf{z} は目的ベクトル、 \mathbf{x} は決定ベクトル、 \mathbf{X} は決定変数空間内での実行可能領域である。

進化型多目的最適化アルゴリズムに局所探索を組み込む場合には、局所探索における解の更新規則を設定する必要がある。MOGLS では、親個体の選択および子個体に対する局所探索に、以下のような加重とスカラー関数が用いられる。

$$f(\mathbf{x}) = w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \dots + w_N f_N(\mathbf{x}). \quad (3)$$

ここで、非負の重み w_i ($\sum_i w_i = 1$) は、親個体のペアが選択されるごとにランダムに更新される。すなわち、親個体の各ペアは、異なる重みベクトルを持っている。子個体に対する局所探索は、親個体が選択された重みベクトルを用いて行われる。親個体を持たない個体（初期世代としてランダムに生成された個体）に対して局所探索が適用される場合には、重みベクトルはランダムに設定される。

アルゴリズムの実行中に獲得された暫定的な非劣解集合は、個体群とは別に第2個体群に保持され、各世代で更新される。このため、最終世代の第2個体群が最終的に獲得された解集合となる。従来の MOGLS⁽²⁰⁾は、以下のようなアルゴリズムとして書くことができる。

Step 0) 初期個体群としてランダムに N_{pop} 個の個体を生成する。

Step 1) 現在の個体群に含まれる個体に対して、個々の目的関数の値を計算し、第2個体群を更新する。

Step 2) 以下の操作を $(N_{\text{pop}} - N_{\text{elite}})$ 回繰り返す。

(a) 重みベクトルをランダムに設定する。

(b) 式(3)の加重とスカラー適応度関数を用いて親個体のペアを選択する。なお、現在の個体群 Ψ に含まれる各個体 \mathbf{x} の選択確率 $p_S(\mathbf{x})$ は、以下のようなルーレットにより設定される。

$$p_S(\mathbf{x}) = \frac{f_{\max}(\Psi) - f(\mathbf{x})}{\sum_{y \in \Psi} (f_{\max}(\Psi) - f(y))}, \quad (4)$$

ここで、 $f_{\max}(\Psi)$ は、現在の個体群 Ψ に含まれる最悪の個体の適応度関数値（すなわち、最大の適応度関数値）である。

Step 3) Step 2 で選択された親個体の $(N_{\text{pop}} - N_{\text{elite}})$ 個のペアに対して交叉操作と突然変異操作を適用し、同じ数の子個体を生成する。

Step 4) 第2個体群よりランダムに N_{elite} 個の非劣解を選択し、そのコピーを子個体の集合に加え、サイズ N_{pop} の個体群を構成する。

Step 5) Step 4 で構成された個体群に含まれる個々の個体に対して局所探索を適用する。個々の個体に対する局所探索は、現在の解の近傍からランダムに選択された k 個の近傍解が現在の解を改良しない場合に終了する（局所探索の早期終了）。なお、現在の解よりも良い解が見つければ、直ちにその解へ現在の解を移動し、更新された解の近傍で k 個の近傍解が調べられるまで探索を続ける（First Improvement 方式）。全ての解に対

して局所探索を適用した後、改良された個体群を現在の個体群と置き換える (Lamarckian 方式).

Step 6) Step 1 へ戻る.

2.2 MOGLS アルゴリズムの修正

従来の MOGLS アルゴリズムでは、現在の個体群に含まれる全ての個体に対して局所探索が適用される。そのため、他の解よりも大幅に性能の劣る個体が現在の個体群に含まれている場合には、局所探索の適用はアルゴリズム全体の効率を悪くすると思われる。また、親個体の選択時に用いられた重みベクトルは、子個体に対する局所探索の方向としては必ずしも適切ではない。このような観点から、筆者らは、MOGLS の局所探索に対して、以下のような修正を行った⁽²⁵⁾。

Step 5) 以下の操作を N_{pop} 回繰り返すことにより現在の個体群を更新する。

- (a) 重みベクトルをランダムに更新する。
- (b) 加重和スカラー適応度関数を用いたトーナメント方式により現在の個体群から個体を 1 個選択する。トーナメント選択では全くランダムな復元抽出を用い、トーナメントサイズは 5 とした。
- (c) 上記(b)で選択された個体に対して確率 p_{LS} で局所探索を適応する。局所探索が適応された場合には、局所探索が終了した時点での現在の解を次世代の個体群に加える。一方、局所探索が適応されなかった場合は、上記(b)で選択された個体をそのまま次世代の個体群に加える。

このような修正を加えることにより、良好な個体にだけ局所探索が適用される。また、重みベクトルに適した個体を選択されるので、各個体は適切な局所探索の方向を持つことになる (図 2 参照)。

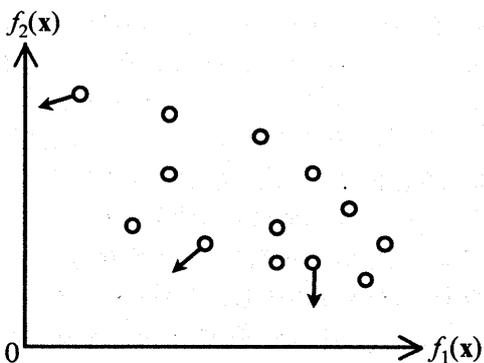


Fig. 2 Local search directions.

2.3 数値実験

Ishibuchi & Murata⁽²⁰⁾と同様の方法で 2 目的 20 機械 40 ジョブフローショップスケジューリング問題を作成した。遺伝的操作としては、図 3 の交叉操作と図 4 の突然変異操作を用いた。なお、図 4 の突然変異操作は、局所探索において近傍解を生成するためにも用いた。数値実験は、以下のようなパラメータ設定を用いて行った。

- 個体群サイズ (N_{pop}): 60,
- 交叉確率: 0.9,
- 個々の個体に対する突然変異確率: 0.6,
- エリート個体の数 (N_{elite}): 10,
- 局所探索の適用確率 (p_{LS}): 0.8,
- 局所探索の停止基準 (k): 2,
- アルゴリズムの終了基準: 10 万個の個体の評価

数値実験の結果を図 5 に示す。この図は、各アルゴリズムを 20 回実行することで得られた 20 個の解集合を示している (合計 40 個の解集合)。図 5 より、局所探索の修正により MOGLS の探索性能が大きく改良していることが分かる。

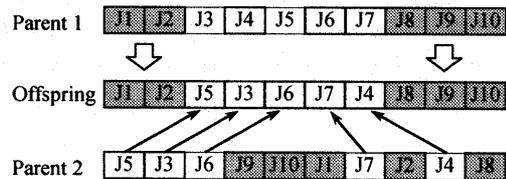


Fig. 3 Two-point crossover.

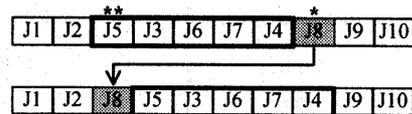


Fig. 4 Insertion mutation.

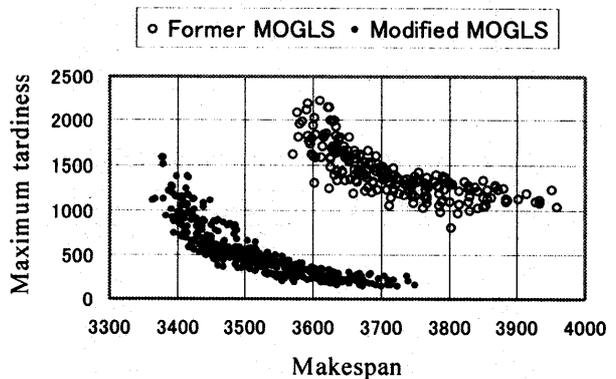


Fig. 5 Former MOGLS and modified MOGLS.

3. 遺伝的探索と局所探索の効果

改良型 MOGLS における遺伝的探索と局所探索の効果をも明らかにするために、Ishibuchi & Murata⁽²⁰⁾と同様の方法で問題サイズの違う4種類の2目的20機械 n ジョブ問題と4種類の3目的20機械 n ジョブ問題を作成し、数値実験を行った ($n=20, 40, 60, 80$)。数値実験では、改良型 MOGLS に加えて、比較のために以下のアルゴリズムを用いた。

MOGA: 局所探索確率を0に設定した改良型 MOGLS。

このアルゴリズムでは、選択、交叉、突然変異という遺伝的操作だけで探索が行われる。

MOLS: 交叉確率と突然変異確率を0に設定した改良型 MOGLS。このアルゴリズムでは、局所探索 (LS) と選択操作だけで探索が行われる。

数値実験の結果を表1～表3に示す。各表では、目的関数の数 N とジョブの数 n により、問題が N/n の形式で示されている。なお、数値実験では個々の問題に対して異なるパラメータ設定を用いた。表1は、十分に長い計算時間を用いて求められた参照解集合(近似パレート最適解集合)からの平均距離を示す。具体的には、参照解集合 S^* と評価対象となる解集合 S が与えられたとき、 S^* に含まれる個々の解とその解に最も近い S の解との距離を正規化した目的関数空間内で測定し、 S^* に含まれる全ての参照解に対する平均値を計算した。この評価基準は Czyzak & Jaskiewicz⁽²⁶⁾で提案されたものであり、Knowles & Corne⁽²⁷⁾では DI_R と呼ばれている。

Table 1 Average distance from reference solutions.

Problem	MOGA	MOGLS	MOLS
2/20	5.84	4.40	8.52
2/40	22.92	17.27	28.52
2/60	23.83	19.35	22.54
2/80	97.75	67.15	72.28
3/20	8.73	7.68	11.63
3/40	21.45	19.61	21.33
3/60	31.49	25.38	53.41
3/80	35.79	31.18	31.49

Table 2 Average number of obtained solutions.

Problem	MOGA	MOGLS	MOLS
2/20	21.05	22.25	13.60
2/40	23.55	22.60	10.80
2/60	19.60	15.75	10.85
2/80	14.35	12.65	8.20
3/20	126.45	107.80	48.40
3/40	94.40	58.00	48.10
3/60	98.05	70.45	20.05
3/80	74.40	51.25	30.80

Table 3 Average ratio of the non-dominated solutions.

Problem	MOGA	MOGLS	MOLS
2/20	0.475	0.737	0.416
2/40	0.283	0.740	0.255
2/60	0.313	0.613	0.502
2/80	0.135	0.659	0.509
3/20	0.483	0.641	0.480
3/40	0.643	0.442	0.632
3/60	0.549	0.778	0.026
3/80	0.427	0.496	0.788

表1から全ての問題に対して MOGLS で最も良い結果が得られていることが分かる。一方、表2は、各アルゴリズムの1回の実行により獲得された解の数の平均値を示したものである。表2から、多数の解を同時に獲得するためには遺伝的操作が重要であることが分かる。また、表3では、3種類のアルゴリズムで獲得された3個の解集合を互いに比較した場合での非劣解の割合の平均値が示されている。例えば、3/80 問題に対して MOLS により獲得された平均30.80個(表2)の解のなかで、平均78.8%の解が他のアルゴリズムで得られた解に優越されない非劣解であった。表3から MOGA に局所探索を組み込むことで非劣解の割合が増加していることが分かる。これは、局所探索がパレート最適解への収束性を向上させているからである。ただし、遺伝的操作を用いない MOLS では、必ずしもパレート最適解への収束性は高くはない(表3の2/40と3/60を参照)。

4. 進化型多目的最適化手法への局所探索の組み込み

2.2節で示した局所探索は、図1に示す枠組みに基づき、他の進化型多目的最適化アルゴリズムへ簡単に組み込むことができる。そこで本研究では、高い探索能力を持つことが知られている進化型多目的最適化手法である SPEA⁽¹¹⁾および NSGA-II⁽¹⁴⁾に局所探索を組み込んだ。しかし、局所探索を適用する個体を選択するときに淘汰圧が加えられるので、高い淘汰圧で親個体の選択を行うような進化型多目的最適化アルゴリズムでは、全体の淘汰圧が高くなりすぎる可能性がある。そこで、局所探索を適用する個体の選択にトーナメント方式を用いる2.2節の局所探索を組み込んだアルゴリズム (Ver.1) に加えて、局所探索を適用する個体をランダムに確率 p_{LS} で選択するアルゴリズム (Ver.2) も実装した。なお、Ver.2 アルゴリズムでは、局所探索における重みベクトルの設定には、Deb⁽³⁾の擬似重みの概念が用いられる。

6種類のアルゴリズムを用いて行った数値実験の結果を表4～表6に示す。3章で示した数値実験と同様に、局所探索によりパレート最適解への収束性が向上するが解の多様性が失われるという結果が得られた。なお、

表中の太字は, Mann-Whitney U test に基づく検定において, ハイブリッド化による性能向上が 95%の信頼度で統計的に有意であったことを示している.

Table 4 Average distance from reference solutions.

Test problem	Pure SPEA	Hybrid SPEA		Pure NSGA-II	Hybrid NSGA-II	
		Ver.1	Ver.2		Ver.1	Ver.2
		2/20	5.1		4.7	5.0
2/40	21.7	15.7	19.6	15.2	14.6	14.4
2/60	19.1	18.4	18.5	17.8	18.0	17.5
2/80	97.3	50.5	71.7	71.3	60.5	64.7
3/20	10.8	9.6	9.3	10.6	9.7	8.8
3/40	16.2	15.9	15.9	20.2	18.2	18.2
3/60	24.9	23.6	23.6	35.9	28.8	28.2
3/80	25.9	26.0	25.4	27.8	27.2	27.1

Table 5 Average number of obtained solutions.

Test problem	Pure SPEA	Hybrid SPEA		Pure NSGA-II	Hybrid NSGA-II	
		Ver.1	Ver.2		Ver.1	Ver.2
		2/20	23.5		22.3	23.2
2/40	21.8	23.7	22.5	23.1	21.5	24.2
2/60	21.5	20.7	23.4	19.9	20.8	22.0
2/80	12.0	9.7	10.1	16.0	16.5	16.0
3/20	30.5	30.6	30.8	48.9	47.0	53.6
3/40	60.3	60.3	60.8	59.2	56.7	58.7
3/60	61.0	30.8	30.8	41.5	31.2	37.4
3/80	60.6	60.6	59.5	53.6	51.1	53.3

Table 6 Average ratio of the non-dominated solutions.

Test problem	Pure SPEA	Hybrid SPEA		Pure NSGA-II	Hybrid NSGA-II	
		Ver.1	Ver.2		Ver.1	Ver.2
		2/20	0.48		0.61	0.50
2/40	0.05	0.34	0.02	0.31	0.39	0.32
2/60	0.24	0.29	0.22	0.25	0.19	0.24
2/80	0.04	0.53	0.11	0.11	0.32	0.26
3/20	0.23	0.58	0.47	0.26	0.36	0.54
3/40	0.29	0.39	0.36	0.24	0.31	0.28
3/60	0.42	0.44	0.47	0.18	0.28	0.15
3/80	0.26	0.32	0.34	0.35	0.23	0.22

さらに, SPEA の Ver.1 ハイブリッドアルゴリズムを用いて局所探索の効果調べた. 具体的には, 局所探索の停止基準 k と局所探索確率 p_{LS} の 132通りの組合せを用いた数値実験を 2/80 問題に対して行った. 数値実験の結果を図6に示す. 図6より, k と p_{LS} が共に大きい場合(図中の右下の隅)では, ハイブリッドアルゴリズムの性能が大きく悪化していることが分かる. この場合

では局所探索に計算時間の大部分が使われ, 世代更新は数世代だけであった. また, 局所探索を全く用いない $p_{LS} = 0$ の場合と比較すると, $k-p_{LS}$ 平面の右上から左下に細長く存在する谷の部分で性能の向上が見られる. 一方, 図7は, 交叉確率 p_C と突然変異確率 p_M の 121通りの組合せに対する結果である. 図7より, 交叉と突然変異を全く行わない場合, すなわち, p_C と p_M が共に0の場合では, 性能が大きく悪化していることが分かる.

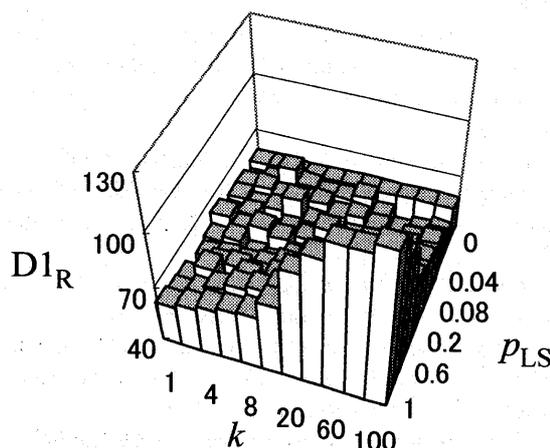


Fig. 6 Effect of local search.

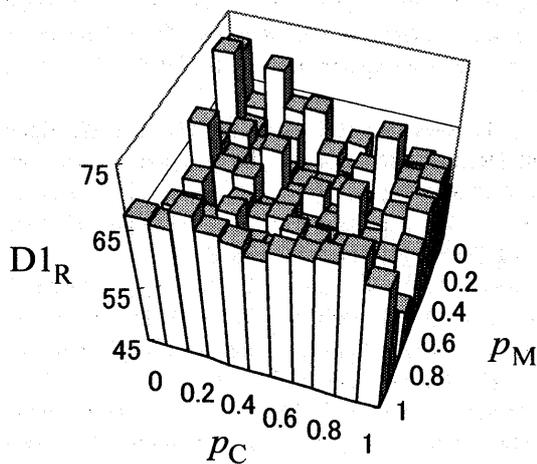


Fig. 7 Effect of genetic search.

5. おわりに

本研究では, 進化的多目的最適化アルゴリズムに局所探索を組み込むハイブリッド化の正の効果と負の効果を示した. 正の効果はパレート最適解への収束性の向上であり, 負の効果は解の多様性の減少である. また, ハ

イブリッドアルゴリズムでは、遺伝的探索と局所探索の両方が必要であることも明らかにした。さらに、SPEAやNSGA-IIのような高い探索性能を持つ進化型多目的最適化アルゴリズムの性能がハイブリッド化により、さらに向上することも明らかにした。最後に、遺伝的探索と局所探索のバランスが悪い場合では、ハイブリッドアルゴリズムの性能が大きく悪化することを明らかにした。なお、本研究の一部は科学研究費補助金（基盤 B 課題番号 14380194）により行われたものである。

参考文献

- (1) C. A. Coello Coello, "A comprehensive survey of evolutionary-based multiobjective optimization techniques," *Knowledge and Information Systems*, vol. 1, no. 3, pp. 269-308, August 1999.
- (2) D. A. Van Veldhuizen and G. B. Lamont, "Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art," *Evolutionary Computation*, vol. 8, no. 2, pp. 125-147, Summer 2000.
- (3) K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Chichester, 2001.
- (4) C. A. Coello Coello, D. A. van Veldhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Publishers, Boston, 2002.
- (5) *First International Conference on Evolutionary Multi-Criterion Optimization*, Zurich, Switzerland, March 7-9, 2001.
- (6) *Second International Conference on Evolutionary Multi-Criterion Optimization*, Faro, Portugal, April 8-11, 2003.
- (7) J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," *Proc. of 1st International Conference on Genetic Algorithms and Their Applications*, pp. 93-100, Carnegie-Mellon University, Pittsburgh, July 24-26, 1985.
- (8) C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization," *Proc. of 5th International Conference on Genetic Algorithms*, pp. 416-423, University of Illinois at Urbana-Champaign, July 17-21, 1993.
- (9) J. Horn, N. Nafpliotis and D. E. Goldberg, "A niched Pareto genetic algorithm for multi-objective optimization," *Proc. of 1st IEEE International Conference on Evolutionary Computation*, pp. 82-87, Orlando, June 27-29, 1994.
- (10) N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221-248, Fall 1994.
- (11) E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. on Evolutionary Computation*, vol. 3, no. 4, pp. 257-271, November 1999.
- (12) J. D. Knowles and D. W. Corne, "Approximating the nondominated front using Pareto archived evolution strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149-172, Summer 2000.
- (13) E. Zitzler, K. Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173-195, Summer 2000.
- (14) K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, April 2002.
- (15) P. Moscato, "Memetic algorithms: A short introduction," in D. Corne, F. Glover, and M. Dorigo (eds.), *New Ideas in Optimization*, McGraw-Hill, pp. 219-234, Maidenhead, 1999.
- (16) *First Workshop on Memetic Algorithms (WOMA I)*, in *Proc. of 2000 Genetic and Evolutionary Computation Conference Workshop Program*, pp. 95-130, Las Vegas, July 8, 2000.
- (17) *Second Workshop on Memetic Algorithms (WOMA II)*, in *Proc. of 2001 Genetic and Evolutionary Computation Conference Workshop Program*, pp. 137-179, San Francisco, July 7, 2001.
- (18) *Third Workshop on Memetic Algorithms (WOMA III)*, Granada, Spain, September 7, 2002 (in press).
- (19) D. E. Goldberg and S. Voessner, "Optimizing global-local search hybrids," *Proc. of 1999 Genetic and Evolutionary Computation Conference*, pp. 220-228, Orlando, July 13-17, 1999.
- (20) H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Trans. on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, vol. 28, no. 3, pp. 392-403, August 1998.
- (21) A. Jaskiewicz, "Genetic local search for multi-objective combinatorial optimization," *European Journal of Operational Research*, vol. 137, no. 1, pp. 50-71, February 2002.
- (22) J. D. Knowles and D. W. Corne, "M-PAES: A memetic algorithm for multiobjective optimization," *Proc. of 2000 Congress on Evolutionary Computation*, pp. 325-332, pp. 98-105, San Diego, July 16-19, 2000.
- (23) J. D. Knowles and D. W. Corne, "A comparison of diverse approaches to memetic multiobjective combinatorial optimization," *Proc. of 2000 Genetic and Evolutionary Computation Conference Workshop Program*, pp. 103-108, Las Vegas, July 8, 2000.
- (24) A. Jaskiewicz, "Comparison of local search-based metaheuristics on the multiple objective knapsack problem," *Foundations of Computing and Decision Sciences*, vol. 26, no. 1, pp. 99-120, 2001.
- (25) H. Ishibuchi, T. Yoshida, and T. Murata, "Balance between genetic search and local search in hybrid evolutionary multi-criterion optimization algorithms," *Proc. of 2002 Genetic and Evolutionary Computation Conference*, pp. 1301-1308, New York, July 9-13, 2002.
- (26) P. Czyzak and A. Jaskiewicz, "Pareto-simulated annealing - A metaheuristic technique for multi-objective combinatorial optimization," *Journal of Multi-Criteria Decision Analysis*, vol. 7, no.1, pp. 34-47, January 1998.
- (27) J. D. Knowles and D. W. Corne, "On metrics for comparing non-dominated sets," *Proc. of 2002 Congress on Evolutionary Computation*, pp. 711-716, Honolulu, May 12-17, 2002.