

多目的最適化における DIRECT を用いた全域探索手法の検討

王 路易

1 はじめに

多目的最適化とは、複数の評価基準を有する最適化である。しかし、これらの複数の評価基準は互いに競合することが多いため、ただ 1 つの最適解は存在しない。そのため、多目的最適化では、互いに劣らない「パレート最適解」という概念を用いて探索を行う。一般にパレート最適解は複数存在することが多く、それぞれの評価基準同士にどの程度トレードオフの関係があるのかを知る上でも、パレート最適解を数多く求めることが重要となる。

多目的最適化では、多点探索によってパレート解集合を一度の探索で得ることができる、遺伝的アルゴリズム (Genetic Algorithm: GA) を用いることが多い。GA を多目的最適化に適用したものが、多目的遺伝的アルゴリズム (多目的 GA) であり、これまでに数多くの多目的 GA に関する研究が行われてきている。

しかし、これらの手法では、探索解集合が良好な解周辺のみに集まる傾向にあるため、未探索領域が広くなる。そのため、得られた解集合が本当にパレート最適解集合なのかわからない。これを「解の信頼性が低い」と呼ぶ。解の信頼性を高めるためには、未探索領域が狭くなるような全域探索を行う必要がある。そこで、本報告では、得られた解集合の信頼性を向上させるため、全域探索可能な最適化手法である DIRECT を多目的最適化問題に適用した NSDIRECT を提案する。

2 多目的 GA の問題点

多目的 GA 手法では、ある世代において「偶然」他の個体より適応度が圧倒的に高い個体が生まれた際、その個体の遺伝子が母集団中に急速に増えて探索がかなり早く収束してしまうといった現象がある。つまり、多目的 GA の探索では、母集団中の非劣個体に依存され、設計変数空間に対して全域探索をする保証はない。したがって、このような探索では設計変数空間において未探索領域を生じ、得られた解集合は解最適解集合である信頼性が欠けている。そこで、設計変数空間に対してなるべく全域探索を行い、得られた解集合に対する信頼性を向上させるメカニズムについて考慮するべきであると考えられる。

3 DIRECT

單一目的最適化問題における、全域探索手法として DIRECT (DIviding RECTangle)²⁾ が知られている。DIRECT では、設計変数空間において、超立方体群の分割を繰り返すことにより探索を進める。超立方体群からどれを分割するかは、それぞれの超立方体の中心の評価値と超立方体の領域を基準にして決定する。中心の評価値が良くて、更に領域が広い超立方体は最適解が含まれ

ている可能性の高い有望な空間だからである。そして、分割する度に全ての超立方体の有望さを確認しているため、DIRECT は全域探索であると言える。DIRECT の処理の概要は次の通りである。

1. 初期化 (設計空間の変換)
2. 超立方体の分割
3. 分割すべき超立方体/超直方体の特定
4. 超立方体/超直方体の分割
5. 終了条件が満たされたまで 3, 4 を繰り返す

以降、各操作について述べる。

3.1 初期化と超立方体の分割

DIRECT の探索は設計空間を次のような超立方体空間に変換することから始まる。

$$\bar{\Omega} = \{x \in R^N : 0 \leq x_i \leq 1\} \quad (1)$$

この空間の中央の点を c_1 とし、 c_1 を評価して $f(c_1)$ を得る。次にこの立方体を分割する。分割は $c_1 \pm \delta \vec{e}_i (i = 1, \dots, N)$ の点を各次元において評価することで行う。 \vec{e}_i は基底ベクトルである。 δ は超立方体の一辺の $1/3$ の長さであり、超立方体を各次元において 3 等分することになる。分割すべき次元の順番は式 (2) の w_i で定義されており、 w_i の値が小さい次元から分割が行われる。

$$w_i = \min(f(c_1 + \delta e_i), f(c_1 - \delta e_i)) \quad (2)$$

$$i = 1, \dots, N$$

この方法で分割を行った 2 次元空間での例を Fig. 1 に示す。Fig. 1 の左の図中の数字は各点における目的関数値を示している。この場合、 $w_1 = 60.0$, $w_2 = 150.1$ となるため、まず x_1 方向で分割が行われる。この分割により、新たに 2 つの長方形が作られ、設計空間全体から構成されていた中央に c_1 を持つ正方形は、 x_1 方向の辺が $1/3$ の長方形となる。この後、 c_1 を持つ長方形に対して x_2 方向で分割を行い、初期分割が終了する。

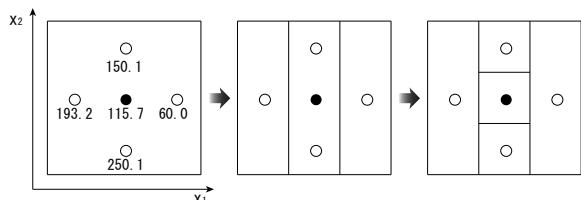


Fig.1 Design variable space after the first division

3.2 分割すべき超立方体/超直方体の特定

ここでは、分割すべき超直方体を特定する方法について述べる。DIRECT は次の定義を満たすすべての po-

tentially optimal な超直方体あるいは超立方体を分割する。

ϵ を $\epsilon > 0$ なる定数とし, f_{min} を現在得られている目的関数値の最小値とする。このとき, 次の式(3)を満たす $K > 0$ が存在すれば, ある超直方体/超立方体 j は potentially optimal である。

$$\begin{aligned} f(c_j) - Kd_j &\leq f(c_i) - Kd_i, \forall i, \text{and} \\ f(c_j) - Kd_j &\leq f_{min} - \epsilon|f_{min}| \end{aligned} \quad (3)$$

ここで, c_j は超直方体 j の中心の点である。 d_j は超直方体の大きさを定義するものであり, Jones らは中心 c_j から超直方体の頂点までの距離を用いている。また, ϵ は経験的に 1.0×10^{-4} が良いとされている。この定義を図で表したものを見 Fig. 2 に示す。

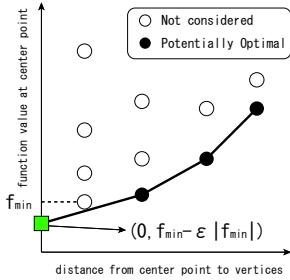


Fig.2 Selection of hyper-rectangles to be divided

3.3 超立方体/超直方体の分割

超直方体の分割では, 最も長い辺を持つ次元についてのみ分割が行われる。分割の順序については式(4)の w_j が小さい順に行われる。

$$w_j = \min(f(c_i + \delta_i e_j), f(c_i - \delta_i e_j)) \quad (4)$$

$$j \in I$$

ここで I はある超直方体 i における, 最長の辺をもつ次元の集合である。また, δ_i は超直方体 i の最長の辺の $1/3$ の長さである。DIRECT は I に含まれるすべての次元に対して分割を行う。以上の操作を繰り返すことにより, DIRECT の探索は行われる。一連の操作による探索のイメージを Fig. 3 に示す。

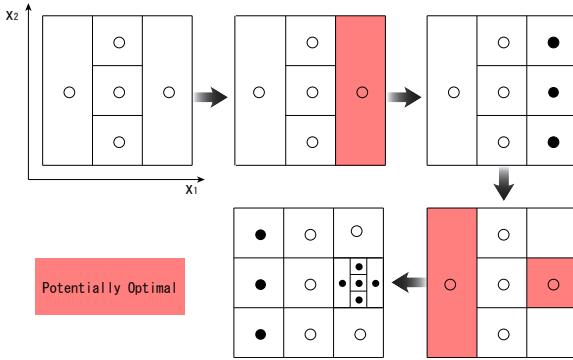


Fig.3 Several iterations of DIRECT search

4 DIRECT の多目的最適化への適用

多目的最適化においても全域探索を行うため, DIRECT を多目的最適化問題に適用する。本提案手法

を NSDIRECT (Non-Dominated Sorting DIRECT) と呼ぶ。

DIRECT では超立方体の分割基準の 1 つに評価値を用いた。しかし多目的最適化では評価基準が複数あるため, NSDIRECT では, 非優越ソート (Non-Dominated Sort) によるランキング法と混雑距離 (Crowding Distance)¹⁾ を導入した。改良点としては, 超立方体の分割基準として評価値 $f(c_1 \pm \delta e_i^j)(i = 1, \dots, N)$ ではなく, 以下の 2 つを用いる。

1. 超立方体の中心点における適合度値 (Rank)
2. 超立方体の中心点における混雑距離 (CD)

5 数値実験

5.1 実験概要

本実験では, 提案手法である NSDIRECT が一般的な多目的 GA 手法より信頼性の高い解集合を得られるかを確認するために, 一般的な多目的 GA と探索の様子を比較する。また, 各手法が設計変数空間において全域探索を行ったかを確認するために, テスト関数のランドスケープを用いて比較する。なお, 多目的 GA の手法には NSGA-II¹⁾ を用いる。NSGA-II および NSDIRECT の評価回数は共に 5000 回とした。

5.2 対象問題

本実験で用いる対象問題は, 2 目的の連続最適化問題の单峰性関数 ZDT2 と, 多峰性関数 ZDT4 である。なお, 設計変数の数は 2 とする。

ZDT2 のパレート最適フロントは形状は非凸型である。式(5)に ZDT2 の定式を示す。

$$\begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = g(x_2, \dots, x_n) \cdot h(f_1, g) \\ g(x_2, \dots, x_n) = 1 + 9(\sum_{i=2}^n x_i)/(n-1) \\ h(f_1, g) = 1 - (f_1/g)^2 \\ \text{subject to } x_i \in [0, 1], \quad i = 1, \dots, n, \quad n = 2 \end{cases} \quad (5)$$

ZDT4 のパレート最適フロントは形状は凸型である。式(6)に ZDT4 の定式を示す。 d の値は 0.5 とした。

$$\begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = g(x_2, \dots, x_n) \cdot h(f_1, g) \\ g(x_2, \dots, x_n) = 1 + 10(n-1) \\ \quad + (\sum_{i=2}^n ((x_i - d)^2 - 10\cos(4\pi(x_i - d))) \\ h(f_1, g) = 1 - \sqrt{f_1/g} \\ \text{subject to } x_1 \in [0, 1] \\ \quad x_i \in [-5, 5], \quad i = 2, \dots, n, \quad n = 2 \end{cases} \quad (6)$$

5.3 評価方法

本報告では Generational Distance (GD) を用いて, 得られたパレート解集合を評価する。GD は探索によって得られたパレートフロントが, パレート最適フロントからどれだけ離れているかを示す評価関数である。GD の値が小さい個体ほど, パレート最適フロントに近いを意味する。GD は式(7)によって求められる。また GD の概念図を Fig. 7 に示す。

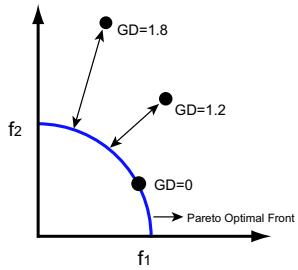


Fig.4 Generational Distance

$$GD = \left(\frac{1}{n_{PF}} \sum_{i=1}^{n_{PF}} d_i^2 \right)^{\frac{1}{2}} \quad (7)$$

このとき, n_{PF} はパレートフロントに含まれる個体数である. また, d_i はパレートフロントに含まれる個体 i とパレート最適フロントの中で 1 番近い個体との目的関数空間におけるユークリッド距離である.

5.4 Parameters

本実験で用いた NSGA-II のパラメータを Table 1 に示す.

Table1 NSGA-II Parameter Settings.

Maximum Generation	50
Population size	100
Crossover Method	2 point crossover
Crossover Rate	1.0
Gene length	10*Dimension
Mutation Rate	1.0/Gene length
Crowding tournament size	2

5.5 実験結果

各手法により ZDT2 と ZDT4 の探索を 30 試行ずつ行った. 得られた GD の中央値と 30 試行の各試行の非劣解の全プロットをそれぞれ Fig. 5, Fig. 6 に示す.

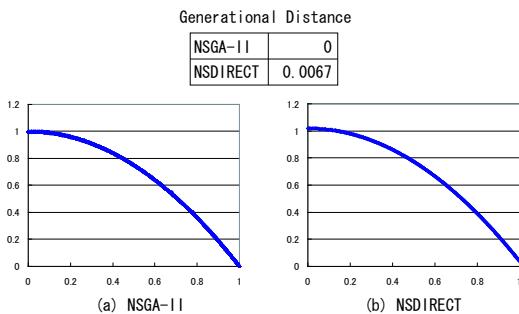


Fig.5 Search Results and GD of ZDT2 in 30 trials

Fig. 5, Fig. 6 に示した探索結果より, ZDT2 では NSGA-II と NSDIRECT はほぼ同じような解集合を得られていることが確認できた. しかし, ZDT4においては, NSDIRECT で得られた解集合は NSGA-II に比べ多様性が欠けていることがわかった. また GD 値を比較すると, すべてのテスト問題において, NSGA-II で得られたパレートフロントは完全にパレート最適フロントに到達していることが確認できた. その一方で, NSDIRECT

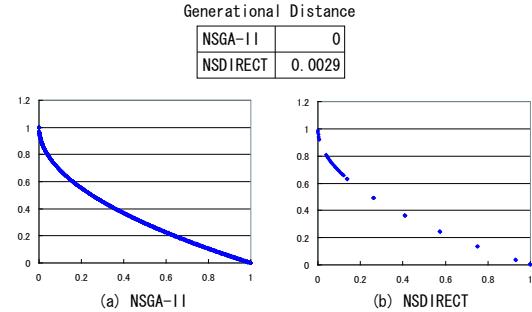


Fig.6 Search Results and GD of ZDT4 in 30 trials

では, 得られたパレートフロントはパレート最適フロントに到達していないが, パレート最適フロントにはほぼ等しいことが確認できた. その理由として, NSDIRECT で得られる解は超直方体の中心点のみであるため, 超直方体の辺上に位置する解は求められないことがあげられる. ZDT2 と ZDT4 のどちらのテスト問題においても, パレート最適フロントが超直方体の辺上に存在するため, NSDIRECT では到達できない. したがって, 解集合の精度については NSDIRECT と NSGA-II はほぼ同じであることが確認できた. これらのことから, NSDIRECT は単峰性の関数では有効だが, 多峰性の関数には適さないことがわかった.

次に各手法による ZDT2 と ZDT4 の設計変数空間における探索履歴を Fig. 7, Fig. 8 に示す.

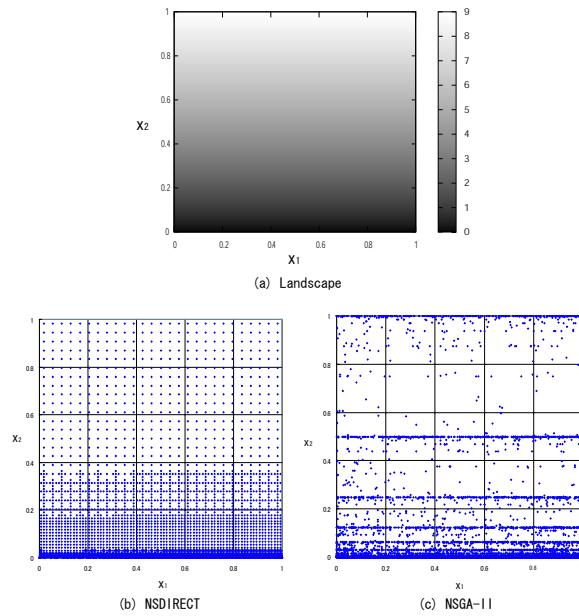


Fig.7 All plots in design variable space of ZDT2

Fig. 7, Fig. 8 に示した探索履歴から, ZDT2 と ZDT4 のどちらの問題においても, NSDIRECT は探索空間を広範囲にわたって探索し, 大きな未探索領域を残していないことから, 全域探索を実現していることが確認できた. 一方, NSGA-II では, パレート最適フロントが求められたが, 設計変数空間では大きな未探索領域が存在していることが確認できた. したがって, 同等な精度のパレートフロントが得られた場合, NSGA-II よりも未探索

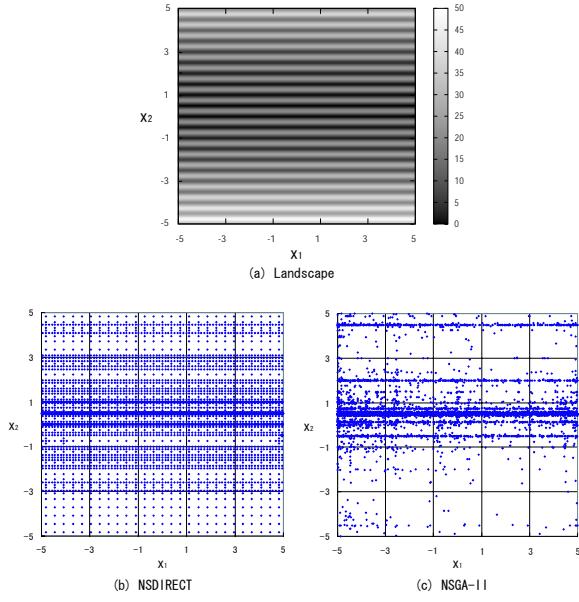


Fig.8 All plots in design variable space of ZDT4

領域の少ない提案手法 NSDIRECT のほうが得られたパレートフロントに対する信頼性が高いと考えられる。

また、NSDIRECT の探索履歴より、パレート最適フロントだけでなく、局所パレートフロントの位置も大まかに把握できたことがわかった。Fig. 7 (a), Fig. 8 (a) に示した各テスト問題のランドスケープと比べると、NSDIRECT の探索履歴が設計変数空間のランドスケープを示していることがわかった。この情報は得られたパレートフロントに対する信頼性を検証するための材料になると考えられる。

6 まとめ

本報告では、DIRECT に改良を加えて、多目的最適化問題に適用した NSDIRECT を提案し、検証を行った。NSDIRECT では、全域探索を行うという DIRECT の特徴を生かし、未探索領域を減少させ、得られた解がどの程度探索して得られた解であるかを把握することで最終的なパレート解集合に対する信頼性を向上できた。

提案した NSDIRECT の性能を検証するために、数値実験を行い、NSGA-II と比較した。結果、同等な精度が得られた。その一方で、解の多様性については、提案手法は多様性の閾値では NSGA-II より劣る結果となった。これより、NSDIRECT は単峰性関数では有効なもの多峰性関数に関しては課題が残るといえる。また、未探索領域については、提案手法の方が NSGA-II より少ないという結果が得られた。さらに、テスト問題のランドスケープと比べると、NSDIRECT の探索履歴が設計変数空間のランドスケープ情報を示していることがわかった。

以上のことから、NSDIRECT では未探索領域を減少させることができ、設計変数空間の情報を大まかに把握することができると考えられる。これより、一般的な多目的 GA の手法に比べて得られた解への信頼性を向上させることができる。今後の課題として、多峰性関数に適用させることが上げられる。

参考文献

- 1) K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II", KanGAL report 200001, Indian Institute of Technology, Kanpur, India, (2000).
- 2) C.D. Perttunen Lones, D.R. and B.R. Stuckman. Lipschitzian optimization without the Lipschitz constant. Journal of Optimization Theory and Applications, Vol.79, No.1, pp.157-181, (1993).