

探索停滞判定時に探索を再初期化する Differential Evolution on Scattered Parents

岩井 亮^{1,a)} 加藤 昇平^{1,b)}

概要: 本稿では Differential Evolution on Scattered Parents (DE-SP) に対して, 利便性の向上を目的とした改変を加えた Self-Adaptive Differential Evolution on Scattered Parents with Dynamic Restart (SDE-SP-DR) を提案する. DE-SP から SDE-SP-DR への主な改変は, 始めにパラメータを乱数で設定し, 探索の停滞時を検知した場合にパラメータを乱数で再設定し, 個体群が局所解へ収束した場合は探索の再初期化をする点である. DE-SP は手法利用者によるパラメータの設定が困難な問題があるが, この改良により手法利用者から DE-SP のパラメータが隠蔽される. この改良により従来手法よりも性能が大きく落ちていなければ, SDE-SP-DR は DE-SP よりも利便性が高い手法であることが示唆される. SDE-SP-DR の最適化手法としての有効性を確認するため, 本研究に先行する改良手法との比較実験を行う. この実験により, SDE-SP-RJ は本研究に先行する改良手法より高い性能を持ち, 従って SDE-SP-DR は DE-SP より利便性が高いことが示唆された.

1. 研究経緯概略

Differential Evolution (DE) [1] は, 運用次第では最適化手法の中でも強力なものであることが期待される手法であるという知見が得られつつある [2], [3], [4], [5]. しかし本稿に先行する研究において, DE は大域的単峰性の乏しい探索空間において著しく探索性能が減少することが確認され, 突然変異個体をより多様となる改良を施すことによりこの問題の軽減が試みられた [6], [7]. この改良手法が Differential Evolution on Scattered Parents (DE-SP) である. DE-SP は従来の DE, 典型的な Swarm Intelligence に基づいた手法, Random Search と比較して, 本稿に先行する研究において問題とされた大域的単峰性に乏しい目的関数 (4 節にて詳細を述べる. 外観は図 4(a) に示す.) において最も高い最適化性能を持つことが確認された [8].

この DE-SP において, 最適化性能の改善はなされたが, 手法を実装・適用する際に手法実装者によって設定する必要のあるパラメータの適切な調整が困難であるという問題 [9], [10] は, これらの本稿に先行する研究においては解決されなかった. この解決のため, 本稿に先行する二つの報告 [6], [7] の後, DE-SP のパラメータを削除, あるい

はより値変動に過敏でないパラメータ *¹ への置き換えが二度試みられた [11], [12]. この内, 報告 [11] においては, 改良方法自体は参考に値するものの, 実験環境設計と, その結果から推測や連想が明確に不可能な考察が成されているため本稿では取り上げない. 報告 [12] において提案された Self-Adaptive Differential Evolution on Scattered Parents with Random Jump (SDE-SP-RJ) は, DE-SP のパラメータを乱数を用いて設定し, 内部基準により探索停滞と判断された場合に DE-SP のパラメータを再度乱数を用いて設定する手法である. 手法の詳細は 2.2 節にて述べる. SDE-SP-RJ はこの内部基準に一つのパラメータ設定が必要である. 報告 [12] においてはこの残った一つのパラメータは DE-SP のパラメータよりも寛容である旨が述べられており, 一応は DE-SP のパラメータ調節の困難性に対する改善がなされたと考えられる. ここで寛容とは, 最適化性能が最上である, あるいはそれに準じる性能を発揮可能なパラメータ範囲が広いことを意味する.

当然ながら, パラメータの設定困難性に対して最も理想的な解決は, 手法実装者によって設定が必要なパラメータを全て廃することである. そこで本稿では, その一つの試みとして, Self-Adaptive Differential Evolution on Scattered Parents with Dynamic Restart (SDE-SP-DR) を提案する. SDE-SP-DR では手法実装者により設定しなければな

¹ 名古屋工業大学
Nagoya Institute of Technology, Showa, Nagoya, Aichi, 466-8555, Japan

a) iwai@katolab.nitech.ac.jp

b) shohey@katolab.nitech.ac.jp

*¹ 値変動が極端に過敏でなければ, 実用において調整の容易性につながり得る.

```

1:  $D \dots$  目的関数の次元数
2:  $N \dots$  個体数
3:  $F \in [0, 2] \dots$  スケーリングパラメータ
4:  $C_R \in [0, 1] \dots$  交叉時に用いる閾値
5:  $M \dots$  改悪を受理する個体数
6:  $R_U(\mathbb{R}, [\alpha, \beta]) \dots$  範囲  $[\alpha, \beta]$  の実数一様乱数
7:  $R_U(\mathbb{N}, [\alpha, \beta]) \dots$  範囲  $[\alpha, \beta]$  の自然数一様乱数
8:
9: for  $i = 1$  to  $N$  do
10:   個体  $\chi^i$  を乱数で初期化
11:   目的関数  $E(\chi^i)$  を計算
12: end for
13: while 終了条件が未成立 do
14:   for  $i = 1$  to  $N$  do
15:      $l^i \leftarrow R_U(\mathbb{N}, [1, D])$ 
16:     for  $k = 1$  to  $D$  do
17:       サブ親  $\chi^{a_k}, \chi^{b_k}, \chi^{c_k}$  ( $i \neq a_k \neq b_k \neq c_k$ ) をランダムに
       選択
18:       突然変異個体を計算:  $\chi_k'^i \leftarrow \chi_k^{a_k} + F(\chi_k^{b_k} - \chi_k^{c_k})$ 
19:       if  $R_U(\mathbb{R}, [0, 1]) < C_R$  or  $k = l^i$  then
20:         子個体候補の要素を次のように設定:  $p_k^i \leftarrow \chi_k'^i$ 
21:       else
22:         子個体候補の要素を次のように設定:  $p_k^i \leftarrow \chi_k^i$ 
23:       end if
24:     end for
25:     目的関数  $E(p^i)$  を計算
26:   end for
27:   for  $i = 1$  to  $N$  do
28:     if  $E(p^i) < E(\chi^i)$  or  $E(\chi^i)$  が下位  $M$  個体までの値である
     then
29:       親個体を子個体候補と置換:  $\chi^i \leftarrow p^i$ 
30:     else
31:       親個体  $\chi^i$  がそのまま子個体となり次の摂動に引き継ぎ
32:     end if
33:   end for
34: end while

```

図 1 Differential Evolution on Scattered Parents の疑似コード

らないパラメータは完全に廃されており、その最適化性能も本研究による現在の実験においては SDE-SP-RJ よりも高いことが確認された。

SDE-SP-DR が他の目的関数においても有用であれば、DE の性能向上と利便性向上に関する幾らかの試みは、一先ずは先へと一歩進むことができると言えるであろう。それは本稿を執筆している現時点では今後の課題とする。

2. 先行研究

2.1 Differential Evolution on Scattered Parents

報告 [6] で提案された DE-SP の DE からの改良点を簡潔に述べる。まず、疑似コードを図 1 に示す。DE-SP では DE におけるサブ親を選択する過程を各要素（目的関数の各次元）毎にやり直す（図 1, 17 行目）ことにより突然変異個体の多様性を確保し、結果として生成される子個体の候補の多様性を向上させている。また、個体集団中の下位 M 個体までは、これらの子個体の候補を無条件に受理

する（図 1, 28 行目）ことで局所解に個体が集まってからの離脱を試み易くなるよう改変されている。

2.2 Self-Adaptive Differential Evolution on Scattered Parents with Random Jump

報告 [12] で提案された SDE-SP-RJ の疑似コードを図 2 に示す。SDE-SP-RJ では始めに F , C_R , M が乱数で初期化され、その後、原則として DE-SP と同様の手順で探索が進められる。DE-SP からの差異は、目的関数が個体集団中の下位 M 個体までの個体を各摂動における突然変異計算の開始時に再度乱数を用いて初期化する点（図 2, 24 行目）と、最後に最良解が更新されてからの摂動回数 T_S を記録（図 2, 45 行目から 49 行目）し、 T_S が閾値 $T_{S_{\max}}$ を超えた場合は摂動の開始時に F , C_R , M を再度乱数を用いて振り直す点（図 2, 18 行目から 21 行目）である。この $T_{S_{\max}}$ は手法利用者により設定される必要がある。 T_S は SDE-SP-RJ において探索の停滞を検出する指標であり、 $T_S > T_{S_{\max}}$ となった時に探索が停滞していると判断してパラメータを再設定し、探索の継続が可能なパラメータを模索する設計となっている。また、この改変により M が小さい値から大きい値に移る場合に、既に個体群が局所解に収束していると、突然変異個体の生成され得る広さも狭くなるため、子個体の候補を無条件に受理するだけでは局所解から離脱することが困難となる。摂動開始時に目的関数値が個体集団中の下位 M 個体に属する個体は摂動開始時に再初期化される改変は、この状態に陥ることを避けるためのものである。

3. Self-Adaptive Differential Evolution on Scattered Parents with Dynamic Restart

本稿で提案する SDE-SP-DR の疑似コードを図 3 に示す。SDE-SP-DR に施された改変について詳細を述べる前に、DE-SP で存在していた M が SDE-SP-DR では廃止されていることを予め明記する。この意図は個体群の再初期化に関わる説明部にて述べる。ここから、図 3 の順序に従って、DE-SP、そして SDE-SP-RJ からの改変点を述べていく。まず、 F と C_R が乱数によって設定される点は SDE-SP-RJ と同等の発想による DE-SP からの改変である（図 3, 15 行目）。次に、乱数によって設定された F と C_R は現在の摂動を終える時、「全親個体の子個体候補と置換されなかった場合」に乱数によって再設定される（図 3, 37 行目, 38 行目）。この改変の意図は、探索が停滞した場合にパラメータの再設定が発生することを期待したものである。しかし、この「全親個体の子個体候補と置換されなかった場合」は基準としては不確実なものであり、必ずしも探索が停滞していないにも関わらずパラメータが再設定されることが頻繁に発生することが予測される。このような問題が予想されなが

```

1:  $D \dots$  目的関数の次元数
2:  $N \dots$  個体数
3:  $F \in [0, 2] \dots$  スケーリングパラメータ
4:  $C_R \in [0, 1] \dots$  交叉時に用いる閾値
5:  $M \dots$  改悪を受理する個体数
6:  $T_S$  最良解更新からの振動回数
7:  $T_{S_{\max}}$  最良解更新からパラメータ  $F$ ,  $C_R$ ,  $M$  の振り直しを待つ最大振動回数
8:  $R_U(\mathbb{R}, [\alpha, \beta]) \dots$  範囲  $[\alpha, \beta]$  の実数一様乱数
9:  $R_U(\mathbb{N}, [\alpha, \beta]) \dots$  範囲  $[\alpha, \beta]$  の自然数一様乱数
10:
11:  $F$ ,  $C_R$ ,  $M$  を乱数で初期化:  $F \leftarrow R_U(\mathbb{R}, [0, 2])$ ,  $C_R \leftarrow R_U(\mathbb{R}, [0, 1])$ ,  $M \leftarrow R_U(\mathbb{N}, [0, N])$ 
12:  $T_S$  を初期化:  $T_S \leftarrow 0$ 
13: for  $i = 1$  to  $N$  do
14:   個体  $\chi^i$  を乱数で初期化
15:   目的関数  $E(\chi^i)$  を計算
16: end for
17: while 終了条件が未成立 do
18:   if  $T_S \geq T_{S_{\max}}$  then
19:      $F$ ,  $C_R$ ,  $M$  を乱数を用いて振り直す:  $F \leftarrow R_U(\mathbb{R}, [0, 2])$ ,  $C_R \leftarrow R_U(\mathbb{R}, [0, 1])$ ,  $M \leftarrow R_U(\mathbb{N}, [0, N])$ 
20:      $T_S$  を初期化:  $T_S \leftarrow 0$ 
21:   end if
22:   for  $i = 1$  to  $N$  do
23:     if  $E(\chi^i)$  が下位  $M$  個体までの値である then
24:       個体  $\chi^i$  を乱数で初期化
25:     end if
26:      $l^i \leftarrow R_U(\mathbb{N}, [1, D])$ 
27:     for  $k = 1$  to  $D$  do
28:       サブ親  $\chi^{a_k}, \chi^{b_k}, \chi^{c_k}$  ( $i \neq a_k \neq b_k \neq c_k$ ) をランダムに選択
29:       突然変異個体を計算:  $\chi_k'^i \leftarrow \chi_k^{a_k} + F(\chi_k^{b_k} - \chi_k^{c_k})$ 
30:       if  $R_U(\mathbb{R}, [0, 1]) < C_R$  or  $k = l^i$  then
31:         子個体候補の要素を次のように設定:  $p_k^i \leftarrow \chi_k'^i$ 
32:       else
33:         子個体候補の要素を次のように設定:  $p_k^i \leftarrow \chi_k^i$ 
34:       end if
35:     end for
36:     目的関数  $E(p^i)$  を計算
37:   end for
38:   for  $i = 1$  to  $N$  do
39:     if  $E(p^i) < E(\chi^i)$  or  $E(\chi^i)$  が下位  $M$  個体までの値である then
40:       親個体を子個体候補と置換:  $\chi^i \leftarrow p^i$ 
41:     else
42:       親個体  $\chi^i$  がそのまま子個体となり次の振動に引き継ぎ
43:     end if
44:   end for
45:   if 最良解が更新 then
46:      $T_S$  を初期化:  $T_S \leftarrow 0$ 
47:   else
48:      $T_S$  をインクリメント:  $T_S \leftarrow T_S + 1$ 
49:   end if
50: end while

```

図 2 Self-Adaptive Differential Evolution on Scattered Parents with Random Jump の疑似コード

```

1:  $D \dots$  目的関数の次元数
2:  $N \dots$  個体数
3:  $F \in [0, 2] \dots$  スケーリングパラメータ
4:  $C_R \in [0, 1] \dots$  交叉時に用いる閾値
5:  $M \dots$  改悪を受理する個体数
6:  $T_S$  最良解更新からの振動回数
7:  $T_{S_{\max}}$  最良解更新からパラメータ  $F$ ,  $C_R$ ,  $M$  の振り直しを待つ最大振動回数
8:  $R_U(\mathbb{R}, [\alpha, \beta]) \dots$  範囲  $[\alpha, \beta]$  の実数一様乱数
9:  $R_U(\mathbb{N}, [\alpha, \beta]) \dots$  範囲  $[\alpha, \beta]$  の自然数一様乱数
10:
11: for  $i = 1$  to  $N$  do
12:   個体  $\chi^i$  を乱数で初期化
13:   目的関数  $E(\chi^i)$  を計算
14: end for
15:  $F$ ,  $C_R$  を乱数で初期化:  $F \leftarrow R_U(\mathbb{R}, [0, 2])$ ,  $C_R \leftarrow R_U(\mathbb{R}, [0, 1])$ 
16: while 終了条件が未成立 do
17:   for  $i = 1$  to  $N$  do
18:      $l^i \leftarrow R_U(\mathbb{N}, [1, D])$ 
19:     for  $k = 1$  to  $D$  do
20:       サブ親  $\chi^{a_k}, \chi^{b_k}, \chi^{c_k}$  ( $i \neq a_k \neq b_k \neq c_k$ ) をランダムに選択
21:       突然変異個体を計算:  $\chi_k'^i \leftarrow \chi_k^{a_k} + F(\chi_k^{b_k} - \chi_k^{c_k})$ 
22:       if  $R_U(\mathbb{R}, [0, 1]) < C_R$  or  $k = l^i$  then
23:         子個体候補の要素を次のように設定:  $p_k^i \leftarrow \chi_k'^i$ 
24:       else
25:         子個体候補の要素を次のように設定:  $p_k^i \leftarrow \chi_k^i$ 
26:       end if
27:     end for
28:     目的関数  $E(p^i)$  を計算
29:   end for
30:   for  $i = 1$  to  $N$  do
31:     if  $E(p^i) < E(\chi^i)$  then
32:       親個体を子個体候補と置換:  $\chi^i \leftarrow p^i$ 
33:     else
34:       親個体  $\chi^i$  がそのまま子個体となり次の振動に引き継ぎ
35:     end if
36:   end for
37:   if 全親個体が子個体候補と置換されなかった then
38:      $F$ ,  $C_R$  を乱数を用いて再設定:  $F \leftarrow R_U(\mathbb{R}, [0, 2])$ ,  $C_R \leftarrow R_U(\mathbb{R}, [0, 1])$ 
39:   end if
40:   if  $\min E(\chi) = \max E(\chi)$  then
41:     個体群の再初期化
42:   end if
43: end while

```

図 3 Self-Adaptive Differential Evolution on Scattered Parents with Dynamic Restart の疑似コード

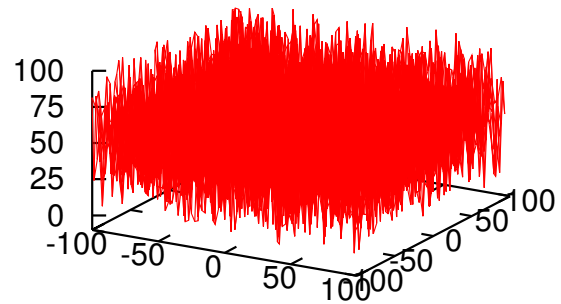
ら本研究においてこの基準を採用した理由は、報告 [4] にて提案されている Self-adaptive Differential Evolution の性質にある。Self-adaptive Differential Evolution では、 F と C_R の二つのパラメータがただの一様乱数と同等な程度に高い頻度で再設定されているにも関わらず、高い性能を示し得る [12] ことが確認された。従って、 F , C_R の高頻度での振り直しは必ずしも探索性能の低下に直結しないと予想し、「探索が停滞していても振り直しが発生しない可能性がある基準」よりは、「探索が停滞していなくとも振り直しは発生するが、停滞していた場合には速やかに振り直しが発生する基準」の方が、改変の試みとして妥当であると、この改変部分において手法設計者は判断したのである。続いて、 F と C_R の再設定判定が終わった後、「最良の目的関数値と最悪の目的関数値が等しい場合」は個体群が再初期化され、探索が再開される (図 3, 40 行目, 41 行目)。この改変の意図は、個体群がある局所解に収束した場合に探索が自動的にやり直されるようにすることである。この基準を探索の停滞 (即ち、 F と C_R の再設定判定) と分けた理由は、個体群が局所解に収束するに当たり、探索が停滞しているように捉えられる状況に何度も陥ることがあるためである。このため、探索が停滞した場合に即座に探索をやり直してしまうと、個体群の収束が不十分なまま探索が初期化され得る。この結果として最適解の発見がいつまで経っても為されないという事態が容易に予想される。従って、SDE-SP-DR においては、探索の停滞の判定よりは厳しい基準を持って個体群の再初期化を行うよう手法設計者は改変した。ただし、こちらの基準は恣意的に設定されたものであることは明記しておく。この基準の設定の妥当性に関する考察は、今後も課題となる。またこの改変により、DE-SP では局所解の離脱を補助するためのパラメータであった M は、SDE-SP-DR において非常に効果の薄いものとなった。このため、SDE-SP-DR では M を廃した。 M を廃することにより、SDE-SP-DR の個体群はその全てが探索に寄与するため、短期的な探索能力の低下を抑える効能も期待できる。

4. Noisy Function 最適化実験

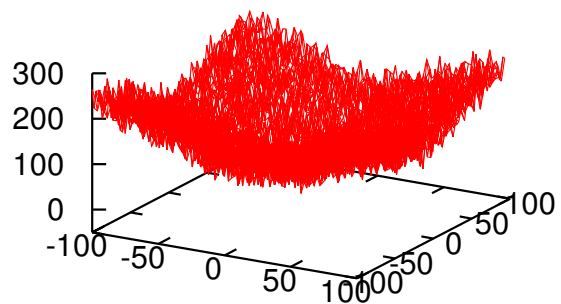
SDE-SP-DR の性能を確認するために、本稿に先行する研究で用いられている Noisy Function 1 (NF1) と Noisy Function 2 (NF2) を用いる。NF1, 2 の外観を図 4 に示す。これらの目的関数の設計を以下に述べる。

- Noisy Function 1 (NF1)

- 整数格子点 (x, y) 上の目的関数の値を、 $R_U(\mathbb{R}, [0, 100])$ を用いて配置する。ただし、点 $(0, 0)$ の値は -1 とし、これを最小値とする。ここで、 $R_U(\mathbb{R}, [0, 100])$ は $[0, 100]$ の範囲で生成される実数の一様乱数である。
- 整数格子点以外の座標 (x, y) の目的関数の値は、 $(\lfloor x \rfloor, \lfloor y \rfloor)$, $(\lfloor x \rfloor + 1, \lfloor y \rfloor)$, $(\lfloor x \rfloor, \lfloor y \rfloor + 1)$, $(\lfloor x \rfloor + 1, \lfloor y \rfloor + 1)$ を用いた 4 点バイリニア法により決定する。



(a) Noisy Function 1



(b) Noisy Function 2

図 4 各 Noisy Function の外観

1) を用いた 4 点バイリニア法により決定する。

- Noisy Function 2 (NF2)

- 整数格子点 (x, y) 上の目的関数の値を、 $|x| + |y| + R_U(\mathbb{R}, [0, 100])$ を用いて配置する。ただし、点 $(0, 0)$ の値は -1 とし、これを最小値とする。
- 整数格子点以外の座標 (x, y) の目的関数の値は、 $(\lfloor x \rfloor, \lfloor y \rfloor)$, $(\lfloor x \rfloor + 1, \lfloor y \rfloor)$, $(\lfloor x \rfloor, \lfloor y \rfloor + 1)$, $(\lfloor x \rfloor + 1, \lfloor y \rfloor + 1)$ を用いた 4 点バイリニア法により決定する。

この設計の都合上、NF1, 2 ののは二次元の目的関数となる。比較手法は次の通りである。

- パラメータを各目的関数に対して調整を施した *2DE
- パラメータを各目的関数に対して調整を施した DE-SP
- $T_{S_{\max}}$ を変化した各々の SDE-SP-RJ
 - それぞれ最大摂動回数の 0.25%, 0.5%, 0.75%, 1%, 2%, 3%, 4%, 5% となるよう設定

各目的関数を最適化する際の実験環境は、最大摂動回数を 300000 回、個体数は 50、試行回数は 1000 回とする。性能の指標は最適解発見回数割合を用いる。最適解発見の成否判断基準は、個体 χ に対して目的関数 $E(\chi)$ を計算した場

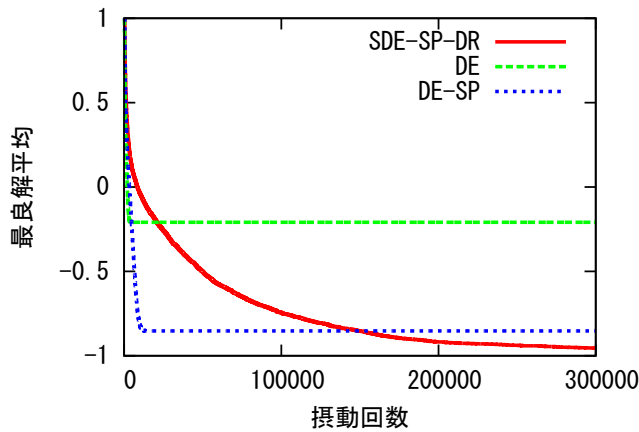
*2 各目的関数に対してパラメータ F , C_R をそれぞれ 0.1 刻みで総当たりを行い、最も性能が高かったものを本来の性能とする。

表 1 SDE-SP-DE, DE-SP, DE の各目的関数における最適化結果 [%]

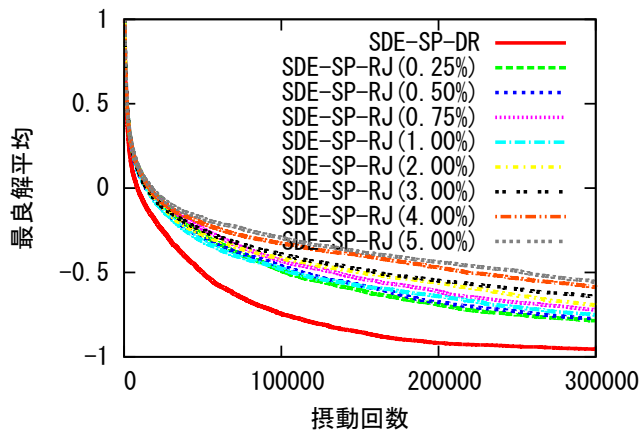
	NF1	NF2
SDE-SP-DR	88.7	99.5
DE-SP	77.9	100
DE	24.2	100

表 2 SDE-SP-RJ の各目的関数における各 $T_{S_{\max}}$ の値ごとの最適解発見回数割合 [%]

$T_{S_{\max}}$	750	1500	2250	3000	6000	9000	12000	15000
NF1	63.1	63.8	62.9	65.6	59.4	55.3	49.6	46.7
NF2	94.5	94.6	94.8	95.1	95.4	95.3	95.7	95.2



(a) SDE-SP-DR, DE-SP, DE の最良解平均推移

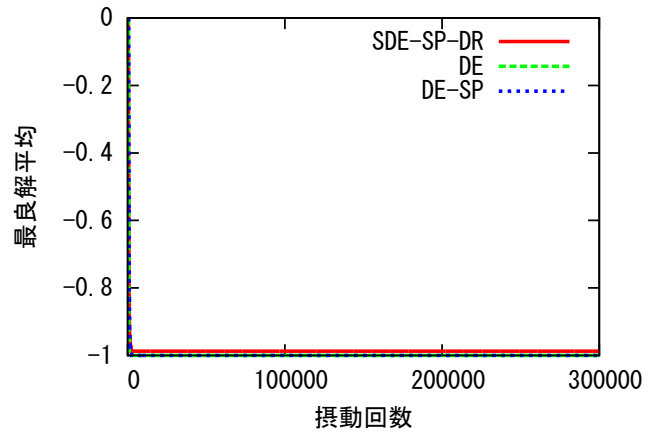


(b) SDE-SP-DR, 各 $T_{S_{\max}}$ を設定した SDE-SP-RJ の最良解平均推移

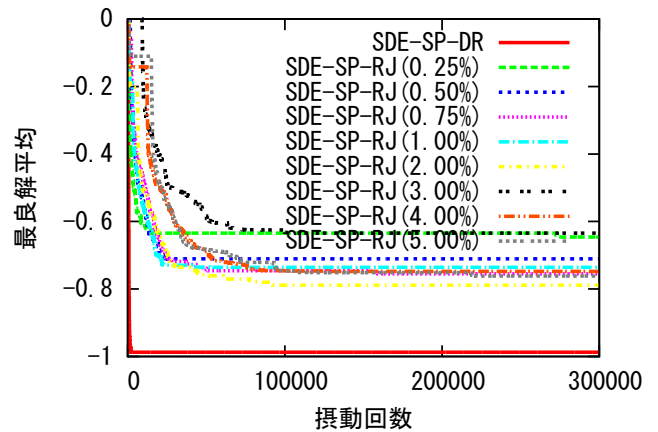
図 5 NF1 を最適化した場合の各手法における最良解平均推移

合に, IEEE 754 の倍精度符号付浮動小数点演算 [13] がアンダーフローを起こし, $E(\chi)$ が目的関数の最小値 $E(\chi^*)$ と等号比較して正になる場合に最適解の発見に成功したとみなすものとする. 即ち, 最適化手法が IEEE 754 の倍精度符号付浮動小数点演算を用いた環境においてこれ以上目的関数を小さくする手がかりを手に入れられなくなるほど個体が最適解付近に近づくことに成功した場合のみ, 最適解の発見に成功したとみなす.

SDE-SP-DR, DE-SP, DE の実験結果を表 1 に, SDE-



(a) SDE-SP-DR, DE-SP, DE の最良解平均推移



(b) SDE-SP-DR, 各 $T_{S_{\max}}$ を設定した SDE-SP-RJ の最良解平均推移

図 6 NF1 を最適化した場合の各手法における最良解平均推移

SP-RJ の実験結果を表 2 に示す. また, NF1 を最適化した場合の最良解平均推移を図 5 に, NF2 を最適化した場合の最良解平均推移を図 6 に示す. ここで, 最良解平均推移とは, 各試行中の各摂動ごとに判明している最良解を保持して置き, 全試行が終わった後に各摂動毎に全試行における保持して置いた最良解の平均を求めたものである. まず, 表に示される結果から, NF1 においては SDE-SP-DR が最も高い性能を示していることが確認される. この結果は, NF1 に対してパラメータ調整が施された DE-SP, DE よりも高いことから, SDE-SP-DR はパラメータ調整の困難性を排除しつつ, 最適化性能を評価基準としても DE-SP から劣っていると明確に判断はされないと考えられるため, SDE-SP-DR はパラメータ調整が困難な目的関数に対して適用を検討する候補に挙げてよい手法であると手法設計者は考えている. NF2 においては, DE-SP, DE よりも若干性能が劣るものの, SDE-SP-RJ と比較すると改善がみられる. このことから, DE-SP や DE で十分な目的関数に対して誤って SDE-SP-DR を用いた場合のリスクは, SDE-SP-RJ を提案していた場合と比較して低減されている. 図 5(a) を確認すると, SDE-SP-DR は

150000 摂動前後で DE-SP の最良解平均を下回っている。また、DE-SP は平均して 1000 摂動前後で探索が停滞している。ここで DE-SP に対して使用すべき計算資源が予め分かっていたという仮定をした場合^{*3}、NF1 において SDE-SP-DR が DE-SP の最良解平均を下回る摂動回数は DE-SP の要する摂動回数の 15 倍程度となる。仮に、本実験と同様に F と C_R を 0.1 刻みで総当たりしてパラメータを確認した場合、パラメータの決定に必要な全摂動回数は $1000 \times 21 \times 11 = 231000$ 摂動である。従って、この条件においては SDE-SP-DR はパラメータを総当たりするよりも (摂動回数を単位として) 速く探索を行うことが可能であると考えられる。ただし、0.2 毎に総当たりを行うならば DE-SP の要する全摂動回数は $1000 \times 11 \times 6 = 66000$ 摂動となり、この場合は DE-SP の方が速くなる。以上のことから、DE-SP でパラメータを決定しようとした場合に 0.1 刻みで総当たりをしないと有効なパラメータを見つけれないと予想されるならば、SDE-SP-DR を用いた方が良いと考えられる。一方、図 5(a)を確認すると、SDE-SP-DR、DE-SP、DE はほぼ同等の推移をすることが確認される。従って、表 1 の考察の差異に述べた、「DE-SP や DE で十分な目的関数に対して誤って SDE-SP-DR を用いた場合のリスクは、SDE-SP-RJ を提案していた場合と比較して低減されている」という結論は、最良解平均推移を確認した場合にも述べることができると考えられる。図 5(b)を確認すると、全ての SDE-SP-RJ のグラフよりも SDE-SP-DR のグラフは下回っている。従って、本稿で提案した SDE-SP-DR は NF1 においては完全に上位互換の挙動をしていると考えられる。図 6(b)でもこの傾向は同様である。

5. まとめと今後の課題

本稿では DE の改良手法である DE-SP のパラメータ設定の困難性を緩和する目的で、更に改良を施された SDE-SP-DR を提案した。次に、性能比較のため同様の目的を持つ関連手法との比較実験について述べ、DE-SP、DE、そして本研究に先行する試みである SDE-SP-RJ と比較した。SDE-SP-DR はこれらと比較して、本実験で設定した基準においては SDE-SP-RJ の完全上位互換であり、NF1 に対しては全比較手法の中で最も高い性能を示すことを確認した。

現在残っている実験を行う時の課題として、NF1、2 の多次元化が挙げられる。本来であれば古典的なベンチマーク関数だけでなく、NF1、2 を多次元化した関数でも実験を行うべきではあるが、本論文の実装のままでは空間計算量が $O(201^D)$ 、時間計算量が $O(2^D)$ となる。このため 4 次元の各 NF から、2014 年現在で既に市販されているような計算機環境では現実的な時間で計算することが困難とな

る問題がある。実験を行うためには少なくとも空間計算量の削減が必要であり、こちらも今後の課題である。

謝辞 本研究は、一部、文部科学省科学研究費補助金 (課題番号 25280100, および, 25540146), ならびに, JST 研究成果最適展開支援事業 (A-STEP) ハイリスク挑戦プログラム (課題番号 AS2524004P) の助成により行われた。

参考文献

- [1] Storn, R. and Price, K.: Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, *Journal of Global Optimization*, Vol. 11, No. 4, pp. 341–359 (1997).
- [2] Bošković, B., Greiner, S., Brest, J. and Žumer, V.: A Differential Evolution for the Tuning of a Chess Evaluation Function, *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*, pp. 1851–1856 (2006).
- [3] 今川哲矢, 伊庭斉志: Differential Evolution を用いた外国為替取引システム, 人工知能学会研究会資料, Vol. 5, pp. SIG-FIN-005-01 (7pages) (2010).
- [4] Omran, M., Salman, A. and Engelbrecht, A.: Self-adaptive Differential Evolution, *Computational Intelligence and Security* (Hao, Y., Liu, J., Wang, Y., Cheung, Y. M., Yin, H., Jiao, L., Ma, J. and Jiao, Y. C., eds.), Lecture Notes in Computer Science, Vol. 3801, Springer Berlin / Heidelberg, pp. 192–199 (2005).
- [5] 串田淳一, 大場和久, 亀井且右: REAL: Differential Evolution における関数評価回数の削減の提案, 進化計算学会論文誌, Vol. 1, No. 1, pp. 79–88 (2010).
- [6] 岩井 亮, 加藤昇平: 大域的単峰性に乏しく多峰な探索空間に有効な Differential Evolution の一改良, 情報学ワークショップ予稿集, Vol. 9, pp. 107–112 (2011).
- [7] 岩井 亮, 加藤昇平: Differential Evolution on Scattered Parents による大域的単峰性に乏しい多峰性探索空間における最適化, 電気学会論文誌 C(電子・情報・システム部門誌) 特集: 省電力時代の電子回路技術, Vol. 133, No. 2, pp. 410–417 (2013).
- [8] Iwai, R. and Kato, S.: Optimization in Multi-Modal Continuous Space with Little Globally Convex using Differential Evolution on Scattered Parents, *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pp. 2002–2007 (2012).
- [9] Gämperle, R., Müller, S. D. and Koumoutsakos, P.: A Parameter Study for Differential Evolution, *WSEAS Int. Conf. on Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation*, Press, pp. 293–298 (2002).
- [10] Liu, J. and Lampinen, J.: On Setting the Control Parameter of the Differential Evolution Method, *Proceedings of the 8th International Conference on Soft Computing (MENDEL 2002)*, pp. 11–18 (2002).
- [11] 岩井 亮, 加藤昇平: 探索停滞時にパラメータを再設定する Differential Evolution on Scattered Parents, 合同エージェントワークショップ&シンポジウム JAWS(Joint Agent Workshop and Symposium), pp. B5–1 (6pages) (2012).
- [12] 岩井 亮, 加藤昇平: 探索の停滞に応じてパラメータを再設定する Differential Evolution on Scattered Parents, 情報処理学会論文誌, Vol. 56, No. 2, pp. 1–11 (2015).
- [13] : IEEE Standard for Floating-Point Arithmetic, *IEEE Std 754-2008*, pp. 1–70 (2008).

^{*3} 即ち、DE-SP を用いて NF1 を最適化する場合には 1000 摂動で十分であることが判明していた場合である。