

2015 年度 卒業論文

卒業論文

Differential Evolution における Archive の性能評価及び改善
Evaluating And Improvement Of Performance Of Archive In Differential Evolution

教養学部学際科学科 総合情報学コース

指導教員: 福永 アレックス

山村 武史

目次

第 1 章	序論	3
1.1	研究の背景	3
1.2	関連研究	3
1.3	本研究の目的	4
1.4	本論文の構成	4
第 2 章	Differential Evolution とアーカイブ	5
2.1	Differential Evolution	5
2.2	Adaptive Differential Evolution (JADE)	8
2.3	Success-History based Adaptive Differential Evolution (SHADE)	10
2.4	DE の探索における収束性と多様性	12
第 3 章	アーカイブの解析	13
3.1	本章の目的	13
3.2	実験設定	13
3.3	実験結果	14
3.4	考察	15
第 4 章	アーカイブ改善の提案	20
4.1	提案手法 1:アーカイブサイズの廃止	20
4.2	提案手法 2:適応的なアーカイブの選択	22
4.3	実験設定	24
4.4	実験結果	24
4.5	考察	25
第 5 章	終わりに	27
5.1	まとめ	27
5.2	謝辞	27
	参考文献	29

第 1 章

序論

1.1 研究の背景

実数値最適化問題とは、ある D 次元の実数値ベクトル $\mathbf{x} = (x_1, x_2, \dots, x_D)$ と、それを評価する関数 $f(\mathbf{x})$ が与えられた時に、その評価関数を最小もしくは最大化するような実数値ベクトル \mathbf{x} を求める問題である。対象問題が、局所解を複数持つ多峰性を有する場合、局所解を避け、大域的な最適解を求めることが実数値最適化問題において重要な課題の一つである。また多くの実問題については、対象問題が単峰性か、多峰性か、及び変数分離可か不可かなどといった情報を探索の事前に知ることは困難である。このような \mathbf{x} の目的関数値 $f(\mathbf{x})$ しか利用出来ない実数値最適化問題を black-box optimization と呼ぶ。そのような実数値最適化問題を対象とする確率的手法の一つとして、差分進化 (Differential Evolution:DE) [1] が用いられる。DE は Evolutionary Algorithm(EA) のひとつであり、EA とは、変化と選択に基づく世代交代により解が進化していく計算の総称である。最初の DE は、1995 年に Storn と Price らによって提案された。DE は単純なアルゴリズムでありながら、他の探索手法と比べても良好な性能を有することが報告されている [1] [2]。このため DE は実問題への多くの適用例が報告されている [2]。

1.2 関連研究

DE の探索性能は用いる制御パラメタに大きく依存し、そのパラメタは集団数、スケール係数 F 、交叉率 CR である。しかしこれらのパラメタの適切な値は使用する関数や問題設定によって異なり、実問題を解く上でこれらのパラメタをユーザーが試行錯誤する必要がある。これらのパラメタを探索中に適応的に変化させていく適応型の DE に関する研究が数多く行われている。それら適応型の DE 手法としては JADE [4]、SHADE [5] などが挙げられる。JADE では適応戦略以外に、探索性能を強化するために過去の劣解を保持するアーカイブを使用する。変異ベクトル作成時にアーカイブに保存された劣解を用いることで、解集団における多様性の維持に役立つ。

しかしアーカイブは問題設定などによっては、探索性能の向上にうまくつながらないこともある [4]。JADE にてその性能比較を行った先行研究では、次元数が高い時にアーカイブは解集団の探索性能を向上させた。一方で、次元数に対して十分な集団数をもつ場合、アーカイブ使用時の方が JADE においてその探索性能は低下した。これは、次元数や集団数によって、解集団が多様性を維持出来ない場合に、アー

カイクがその多様性を向上させるためと述べられている。しかしながらアーカイブが解集団にどの程度多様性を与えているのか、そもそも多様性とは何か、なんらかの尺度による調査はなされていない。本論文では、アーカイブを使用することで解集団における多様性を確かに維持できているのか、次元数、集団数の違いが多様性維持にどのように影響を与えているのか解明するとともに、従来のアーカイブを改善した手法をいくつか提案する。

1.3 本研究の目的

本研究の目的は2つある。

1. アーカイブが多様性の維持にどのように役割をはたしているのか調査する

アーカイブには、生存選択の時に、劣解として、子個体に上書きされた親個体が保存される。変異ベクトル作成時にアーカイブに保存された劣解を用いることで、解集団における多様性の維持に役立つ。本研究では、アーカイブを使用することで、解集団における多様性を確かに維持できているのか、次元数、集団数の違いが多様性維持にどのように影響を与えているのか多様性評価指標 r_s , r_f を基に調査する。

2. アーカイブを改良することで、探索性能を向上できないか新たな手法を提案する。

アーカイブは多くの適応 DE について使われているにも関わらず、そのシステムについての改良は他の制御パラメタであるスケール係数 F や交叉率 CR に比べ試みられていない。アーカイブ性能の分析をふまえた上で、その改善をはかるための新たな手法を提案する。

1.4 本論文の構成

本論文は以下の通りに構成される。2 章で DE の詳細と適応 DE, アーカイブの使用例について説明する。3 章ではアーカイブの性能を多様性評価指標 r_s , r_f の値をもとに調査する。4 章ではアーカイブの改善を試みた二つの手法について説明する。5 章では本研究における知見をまとめる。

第 2 章

Differential Evolution とアーカイブ

2.1 Differential Evolution

まず基本的な DE アルゴリズムについて説明する. DE の集団中の各個体 $i \in \{1, \dots, N\}$ は対象問題の解ベクトル $\mathbf{x}^i = (x_1^i, \dots, x_D^i)$ で表現される. ここで N は集団数, D は次元数である. 探索開始時に各個体は探索領域内にランダムに初期化される. その後, 突然変異戦略による変異個体の生成, 交叉による子個体の生成, 生存選択を, 探索の終了条件を満たすまで繰り返す.

表 2.1 DE における代表的な突然変異戦略

突然変異戦略	定義
rand/1	$\mathbf{v}_i := \mathbf{x}_{r1} + F \cdot (\mathbf{x}_{r2} - \mathbf{x}_{r3})$
rand/2	$\mathbf{v}_i := \mathbf{x}_{r1} + F \cdot (\mathbf{x}_{r2} - \mathbf{x}_{r3}) + F \cdot (\mathbf{x}_{r4} - \mathbf{x}_{r5})$
best/1	$\mathbf{v}_i := \mathbf{x}_{best} + F \cdot (\mathbf{x}_{r1} - \mathbf{x}_{r2})$
best/2	$\mathbf{v}_i := \mathbf{x}_{best} + F \cdot (\mathbf{x}_{r1} - \mathbf{x}_{r2}) + F \cdot (\mathbf{x}_{r3} - \mathbf{x}_{r4})$
current-to-rand/1	$\mathbf{v}_i := \mathbf{x}_i + F \cdot (\mathbf{x}_{r1} - \mathbf{x}_i) + F \cdot (\mathbf{x}_{r2} - \mathbf{x}_{r3})$
current-to-best/1	$\mathbf{v}_i := \mathbf{x}_i + F \cdot (\mathbf{x}_{best} - \mathbf{x}_i) + F \cdot (\mathbf{x}_{r1} - \mathbf{x}_{r2})$
current-to-pbest/1	$\mathbf{v}_i := \mathbf{x}_i + F \cdot (\mathbf{x}_{pbest} - \mathbf{x}_i) + F \cdot (\mathbf{x}_{r1} - \mathbf{x}_{Ar2})$
rand-to-pbest/1	$\mathbf{v}_i := \mathbf{x}_{r1} + F \cdot (\mathbf{x}_{pbest} - \mathbf{x}_{r1}) + F \cdot (\mathbf{x}_{r2} - \mathbf{x}_{Ar3})$

各世代 t において各変異個体 $\mathbf{v}^{i,t}$ となる変異ベクトルを集団中の複数の個体に突然変異戦略を適用することで生成する. 代表的な突然変異戦略を表 (2.1) にて示す. 表 (2.1) において, スケール係数 $F \in (0, 1]$ は突然変異の大きさを調整する制御パラメタである. $\mathbf{x}^{r1}, \mathbf{x}^{r2}, \mathbf{x}^{r3}, \mathbf{x}^{r4}, \mathbf{x}^{r5}$ は \mathbf{x}^i と互いと異なるように集団 $\mathbf{P} = \{\mathbf{x}^1, \dots, \mathbf{x}^N\}$ からランダムに選択した個体である. \mathbf{x}^{best} は各世代における最良個体であり, \mathbf{x}^{pbest} は集団ベクトルを目的関数値の良い順に並び替え, $p \in [0, 1]$ とした時の上位 $\max(N \times p, 2)$ 個体からランダムに選択した個体である. current-to-pbest/1 と rand-to-pbest/1 の $\mathbf{x}^{Ar2}, \mathbf{x}^{Ar3}$ は集団ベクトルと後述のアーカイブベクトルの集合からランダムに選択した個体である. それぞれの突然変異戦略において, best/1, best/2 は変異個体 \mathbf{v}^i を最良個体 \mathbf{x}^{best} の付近に生成する. それに対し current-to-best/1 は対象個体から \mathbf{x}^{best} にむかうように変異個体 \mathbf{v}^i を生成する. そのため rand/1 や rand/2 などに比べ

局所的探索能力が強い戦略である．また加える差ベクトルの数が多いほど多様な変異個体 \mathbf{v}^i を生成しやすい．

次に親個体 \mathbf{x}_i と変異個体 \mathbf{v}_i を交叉させることで子個体 \mathbf{u}_i を生成する．DE の代表的な交叉手法には二項交叉 (binomial crossover) と指数交叉 (exponential crossover) がある．以下では二項交叉について紹介する．二項交叉では交叉率 $CR \in [0, 1]$ とランダムに選択した $j_{rand} \in \{1, \dots, D\}$ に基づき，Algorithm 1 のように子個体 $\mathbf{u}^i = (u_1^i, \dots, u_D^i)$ の各要素 u_j^i を決定する．

Algorithm 1 二項交叉

```

 $j_{rand} = \text{randi}[1, D];$ 
for  $j = 1$  to  $D$  do
  if  $\text{rand}[0, 1] \leq \text{or } j == j_{rand}$  then
     $u_j^i := v_j^i;$ 
  else
     $u_j^i := x_j^i;$ 
  end if
end for

```

全ての個体の子個体を生成した後，次世代に残る個体を決定する．DE では親個体と子個体の目的関数値を評価関数 $f(\mathbf{x})$ を用いて比較し，目的関数値の良いものを次世代へ残す．この選択の際，current-to-pbest/1，及び rand-to-pbest/1 のアーカイブ戦略を用いた場合，子個体より劣っていた親個体 \mathbf{x}^i を，アーカイブに保存する．アーカイブのサイズは集団ベクトルのサイズと等しく，そのサイズを超えた場合ランダムに選択したアーカイブ中の個体を超過分だけ取り除く．current-to-pbest，current-to-prand の突然変異戦略において，アーカイブと集団ベクトルの和集合から親個体 \mathbf{x}^i を選択することで，解集団における多様性の維持に役立つ．最後に本節で述べた DE アルゴリズムの全体を Algorithm 2 に示す．

Algorithm 2 Differential Evolution

集団 $P = \mathbf{x}^1, \dots, \mathbf{x}^N$ の初期化;
while 探索終了条件を満たしていない **do**
 for $i = 1$ to N **do**
 突然変異戦略を用いて変異個体 \mathbf{v}^i を生成;
 \mathbf{x}^i と \mathbf{v}^i に交叉を適用し, 子個体 \mathbf{u}^i を生成;
 end for
 for $i = 1$ to N **do**
 if $f(\mathbf{u}^i) \leq f(\mathbf{x}^i)$ **then**
 $\mathbf{x}^i \rightarrow \mathbf{u}^i$;
 end if
 end for
 もしアーカイブがアーカイブサイズ $|A|$ を超えていれば, 超えた分だけランダムに削除;
end while

2.2 Adaptive Differential Evolution (JADE)

JADE [4] は Zhang らによって 2009 年に提唱された適応型 DE の一つである。JADE では DE の探索性能を大きく左右するスケール係数 F と交叉率 CR を通常の DE のように固定するのではなく、探索中に自動調整する。その自動調整のために適応メタパラメタ μ_F, μ_{CR} を使用する。これらのパラメタは探索開始時に 0.5 に初期化する。通常の DE では F と CR は共通する変数であったが、JADE では、集団の各個体 \mathbf{x}^i ごとに適応パラメタ F_i, CR_i を持つ。そして世代のはじめに F_i と CR_i を、 μ_F, μ_{CR} をもとにそれぞれ式 (2.1)、式 (2.2) のように更新する。

$$F_i = \text{randc}(\mu_F, 0.1) \quad (2.1)$$

$$CR_i = \text{randn}(\mu_{CR}, 0.1) \quad (2.2)$$

ここで $\text{randc}(\mu, \sigma)$ は位置パラメタ μ と尺度パラメタ σ のコーシー分布に従う乱数、 $\text{randn}(\mu, \sigma^2)$ は平均 μ 、標準偏差 σ^2 の正規分布に従う乱数である。 F_i の値が $F_i > 1$ の場合は $F_i = 1$ とし $F_i \leq 0$ の場合は再び式 2.1 を用いて生成を行う。 CR_i の値が区間より外の場合は超えた方の境界値で置き換えられる。コーシー分布は正規分布に比べ、よりひろがりを持つ分布である。そのためコーシー乱数を使った場合の方が、正規乱数に比べ、多様性を維持したパラメタを生成することが出来る。各世代の終了後に、成功した F と CR に基づき μ_F, μ_{CR} を更新する。これらの更新式は式 (2.3)、式 (2.4) となる。

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot \text{mean}_L(S_F) \quad (2.3)$$

$$\mu_{CR} = (1 - c) \cdot \mu_{CR} + c \cdot \text{mean}_A(S_{CR}) \quad (2.4)$$

ここで S_F, S_{CR} とは、各世代において、生存選択を行う際に親個体よりも優れた変異個体を生成することの出来た F_i, CR_i の集合である。次に $c \in [0, 1]$ は学習率であり推奨値は 0.1 である。学習率 c が小さいほど、更新前の μ_F, μ_{CR} に近い値となり、学習率 c が大きいほど、今の世代で成功した F と CR の値による影響が大きくなる。また $\text{mean}_A(\cdot)$ は算術平均、 $\text{mean}_L(\cdot)$ は Lehmer 平均でありそれぞれ式 (2.5)、式 (2.6) となる。

$$\text{mean}_A(S) = \frac{1}{|S|} \sum_{s \in S} s \quad (2.5)$$

$$\text{mean}_L(S) = \frac{\sum_{s \in S} s^2}{\sum_{s \in S} s} \quad (2.6)$$

ここで $S = (s_1, \dots, s_{|S|})$ は S_F, S_{CR} のいずれかである。JADE のパラメタ適応のメカニズムは、探索中に優れた解を生み出すことの出来たスケール係数 F 、交叉率 CR に近づく形で、それらのパラメタを更新していく。そのため、理想的なスケール係数 F 、交叉率 CR が、問題設定や探索途中で変わっていく DE において、適切なパラメタを選択することが JADE において可能である。Algorithm 3 は JADE のアルゴリズムの全体である。

Algorithm 3 JADE

```

  集団  $P = \mathbf{x}^1, \dots, \mathbf{x}^N$  の初期化;
   $\mu_F := 0.5; \mu_{CR} := 0.5f$  で初期化;
  while 探索終了条件を満たしていない do
     $S_F := \emptyset, S_{CR} := \emptyset$ ;
    for  $i = 1$  to  $N$  do
       $F_i := \text{randc}(\mu_F, 0.1)$ 
       $CR_i := \text{randn}(\mu_{CR}, 0.1)$ 
      突然変異戦略を用いて変異個体  $\mathbf{v}^i$  を生成;
       $\mathbf{x}^i$  と  $\mathbf{v}^i$  に交叉を適用し, 子個体  $\mathbf{u}^i$  を生成;
    end for
    for  $i = 1$  to  $N$  do
      if  $f(\mathbf{u}^i) \leq f(\mathbf{x}^i)$  then
         $\mathbf{x}^i \rightarrow \mathbf{u}^i$ ;
         $\mathbf{x}^i := \mathbf{u}^i$ ;
         $S_F := F_i, S_{CR} := CR_i$ ;
      end if
    end for
    if  $S_F, S_{CR} \neq \emptyset$  then
       $\mu_F := (1 - c) \cdot \mu_F + c \cdot \text{mean}_L(S_F)$ 
       $\mu_{CR} := (1 - c) \cdot \mu_{CR} + c \cdot \text{mean}_A(S_{CR})$ 
    end if
    もしアーカイブがアーカイブサイズ  $|A|$  を超えていれば, 超えた分だけランダムに削除;
  end while

```

2.3 Succes-History based Adaptive Differential Evolution (SHADE)

さきほどの JADE につづいて本項では過去の成功したパラメタをもとに適応的に F と CR を変化させる SHADE について説明する. JADE では各世代ごとの成功した有望なパラメタ設定 S_{CR}, S_F に近づくようにパラメタを適応させていった. これは S_{CR}, S_F が対象問題に適したパラメタ値であることを前提としているが, 不適切なパラメタ値が S_{CR}, S_F に含まれる可能性があった. これを防ぐため, SHADE では大きさ H の履歴メモリ M_F, M_{CR} を用いてパラメタ適応を行う. これによって JADE に比べ多様なパラメタ値を保持しながら探索を行えるため, よりロバストな探索が可能となっている. ここで, 離散メモリは, $M_F = (M_{F,1}, \dots, M_{F,H}), M_{CR} = (M_{CR,1}, \dots, M_{CR,H})$ であり, すべての要素は探索開始時に 0.5 に初期化されている. JADE と同様に集団の各個体 \mathbf{x}^i ごとに適応パラメタ F_i, CR_i を持ち, これらのパラメタを各世代のはじめに $[1, H]$ の範囲からランダムに選択したメモリの番号 r の要素 $M_{F,r}, M_{CR,r}$ を用いて式 (2.7), 式 (2.8) のようにして生成する.

$$F_i = \text{randc}(M_{F,r}, 0.1) \quad (2.7)$$

$$CR_i = \text{randn}(M_{CR,r}, 0.1) \quad (2.8)$$

F_i の値が $F_i > 1$ の場合は $F_i = 1$ とし $F_i \leq 1$ の場合は再び式 2.7 を用いて生成を行う. CR_i の値が $[0,1]$ 区間より外の場合は超えた方の境界値で置き換えられる. 各世代の終了時に成功した F_i, CR_i の集合 S_F, S_{CR} の Lehmer 平均を用いて, 式 (2.9), 式 (2.9) のようにメモリ M_F, M_{CR} を更新する.

$$M_{F,k} = \text{mean}_L(S_F) \quad (2.9)$$

$$M_{CR,k} = \text{mean}_L(S_{CR}) \quad (2.10)$$

ここで, $k \in [1, H]$ は更新するメモリの要素を決定するパラメタであり, 探索開始時に 1 に初期化され, 以降更新を行うたびに 1 ずつ増加していく. また, $k > H$ となった場合は $k = 1$ とする. 探索が経過するに連れて M_F, M_{CR} には対象問題に適したかつ, 多様なパラメタ設定が保持される. 一つのパラメタ μ_F, μ_{CR} で管理していた JADE に対し, $M_F = (M_{F,1}, \dots, M_{F,H}), M_{CR} = (M_{CR,1}, \dots, M_{CR,H})$ と SHADE では複数のパラメタを持つため, パラメタにおける多様性が維持しやすいのが特徴である. Algorithm 4 は SHADE のアルゴリズムの全体である.

Algorithm 4 SHADE

```

  集団  $P = \mathbf{x}^1, \dots, \mathbf{x}^N$  の初期化;
   $M_F$  及び  $M_{CR}$  を 0.5 に初期化;
   $k = 1$ ;
  while 探索終了条件を満たしていない do
     $S_F := \emptyset, S_{CR} := \emptyset$ ;
    for  $i = 1$  to  $N$  do
       $r := \text{randi}[1, H]$ 
       $F_i := \text{randc}(M_F, 0.1)$ 
       $CR_i := \text{randn}(M_{CR}, 0.1)$ 
      突然変異戦略を用いて変異個体  $\mathbf{v}^i$  を生成;
       $\mathbf{x}^i$  と  $\mathbf{v}^i$  に交叉を適用し, 子個体  $\mathbf{u}^i$  を生成;
    end for
    for  $i = 1$  to  $N$  do
      if  $f(\mathbf{u}^i) \leq f(\mathbf{x}^i)$  then
         $\mathbf{x}^i \rightarrow \mathbf{u}^i$ ;
         $\mathbf{x}^i = \mathbf{u}^i$ ;
         $S_F := F_i, S_{CR} := CR_i$ ;
      end if
    end for
    if  $S_F, S_{CR} \neq \emptyset$  then
       $M_{F,k} := \text{mean}_L(S_F)$ ;
       $M_{CR,k} := \text{mean}_L(S_{CR})$ ;
       $k = (k + 1) \% H$ ;
    end if
    もしアーカイブがアーカイブサイズ  $|A|$  を超えていれば, 超えた分だけランダムに削除;
  end while

```

2.4 DE の探索における収束性と多様性

通常の DE 及び適応 DE について前節までで説明をした．そして DE の探索性能は一般的に収束性と多様性の間のバランスを左右するパラメタによって決まる．ここで DE における収束性と多様性について再考する．

収束性は、良好な解の近傍に新しい解を発生させることにより向上する．DE においては、ステップ幅を調整する F の値を小さくする．解の更新を速め、子に変位ベクトルに近くなるように CR を大きくするなどが、収束性を向上させる．収束性の向上は最適解により早く近づくことに繋がる一方、局所解に陥りやすくなるという欠点がある．

多様性は、広い範囲で新しい解を発生させることにより向上する．DE においては、ステップ幅を調整する F の値を大きくする．解の更新速度を遅くし、急速な収束を差避けるため CR を小さくする．アーカイブによる突然変異戦略をとるなどが挙げられる．これによって、大域的な最適解を発見出来る可能性は高くなる代わりに、収束性が低下してしまうという問題がある．

このように収束性と多様性は両者ともトレードオフの関係にあたり、収束性と多様性どちらを優先するかはその探索過程で適応的に変化する必要がある．その収束性と多様性をコントロールするパラメタである F と CR を探索過程では適応的に変化させるのが前節でのべた JADE や SHADE であった．アーカイブを使用した先行研究において、アーカイブ使用時に、使用しない時に比べて、常に性能を上げるのではなく、問題設定によっては探索性能が下がったのも、多様性を上げるアーカイブが、収束性を下げることにつながるからである．

次章からは F や CR のように多様性維持に関わるアーカイブについても、適応的に変化させることでより良い結果が出るようになるのではないかと．そもそもアーカイブが多様性の維持に対してどれほど有意な役割をもっているか解析を行う．

第 3 章

アーカイブの解析

3.1 本章の目的

アーカイブを使用した先行研究において JADE の探索性能では少なくとも次元数に依存し、高次元のときに多様性維持に影響を与えるという結果があった [4] . 本章ではそもそも解集団における多様性とは何か、2つの尺度で定義しなおし、アーカイブを使用することで、解集団における多様性を確かに維持できているのか、次元数、集団数の違いが多様性維持に影響を与えているのかどうかを調査する. これにより、アーカイブがどのような条件下で多様性維持に貢献するのかを明確にし、今後アーカイブを用いた優れた DE アルゴリズムを設計するために有益な知見を得る.

3.2 実験設定

解集団における多様性とはそもそも何か. 本実験では次の2つの状態を多様な解を集団ベクトルが維持出来ていると捉えることにした. 一つ目は解集団のベクトル同士が離れた位置にある状態であり、二つ目は解集団のベクトルの示す目的関数値のばらつきが大きい状態である. これら二つの状態を図る指標としてそれぞれ多様性評価指標 r_s , r_f を式 (3.1), 式 (3.2) のように定義した.

$$r_s = \sqrt{\frac{1}{P} \sum_{i=1}^P (d_i - d_{mean})^2} \quad (3.1)$$

$$r_f = \sqrt{\frac{1}{P} \sum_{i=1}^P (f(\mathbf{x}^i) - f_{mean})^2} \quad (3.2)$$

ここで r_s は集団ベクトルにおける重心からの距離の標準偏差, r_f は集団ベクトルにおける目的関数値 $f(\mathbf{x})$ の標準偏差である. また P は集団数, D は次元数である. d_{mean} , f_{mean} はそれぞれ集団ベクトルの重心からの距離の平均, 集団ベクトルの目的関数値 $f(\mathbf{x})$ の平均であり式 (3.3), 式 (3.4) で定義される.

$$d_{mean} = \frac{1}{P} \sum_{i=1}^P d_i \quad (3.3)$$

$$f_{mean} = \frac{1}{P} \sum_{i=1}^P f(\mathbf{x}^i) \quad (3.4)$$

また集団ベクトルの重心からの距離 d_k は式 (3.5) のように定義される．ただし k は $k \in \{1, \dots, P\}$ となる整数値である．ここで \mathbf{c} は解集団 $\mathbf{P} = \{\mathbf{x}^1, \dots, \mathbf{x}^P\}$ における重心ベクトルであり，その要素を $\mathbf{c} = \{\mathbf{c}^1, \dots, \mathbf{c}^P\}$ と表す．

$$d_k = \sqrt{\sum_{i=1}^D (c_i - x_{k,i})^2} \quad (3.5)$$

解集団における重心からの距離の標準偏差と，目的関数値 $f(\mathbf{x})$ の標準偏差をしめすこれらの多様性評価指標 r_s, r_f が大きければ解集団が多様性を維持出来ている事を示す．また実験では目的関数においてシンプルな関数の一つである Sphere 関数を用いて行う．Sphere 関数は式 (3.6) で表される単峰性の関数である．

$$f(\mathbf{x}) = \sum_{i=1}^D x_i^2 \quad (3.6)$$

また実行可能領域は $[-100, 100]^D$ であり，次元数 D は 2, 10, 30, 50, 100 とした．また，集団数 P は 10, 30, 50 とした．最大評価回数は 1 試行あたり $10,000 \times D$ とし，試行回数は 51 回とした．使用した DE は突然変異戦略を式 (3.2) の current-to- p -best とし， $F = 0.5$, $CR = 0.5$ と，アーカイブを使用する時と使用しない場合で比較した．また以下本論文ではアーカイブを使用した際の DE を DE/A，使用しなかった際の DE を DE/NA と表現する．ここでいう DE/A とは式 (3.7) における突然変異戦略 current-to- p -best において \mathbf{x}^{Ar2} を集団 P とアーカイブ A の和集合から \mathbf{x}^i と被らないようにランダムに選択したものである．DE/NA とは \mathbf{DEx}^{Ar2} を \mathbf{x}_i と被らないように集団ベクトルから選択したものである．

$$\mathbf{v}^i = \mathbf{x}^i + F \cdot (\mathbf{x}^{pbest} - \mathbf{x}^i) + F \cdot (\mathbf{x}^{r1} - \mathbf{x}^{Ar2}) \quad (3.7)$$

3.3 実験結果

本節では，多様性評価指標 r_s, r_f の推移を，次元数 D ，集団数 P ，複数の軸をもとに得られた実験結果について述べる．

まず次元数を軸に述べる．次元数 D が 2 である時の多様性評価指標 r_s, r_f を見てみると，探索が 200 世代あたりで終了しているため，DE/A，DE/NA 同士で比べると P が 10, 30, 50 と異なってもそのほとんどに差が見られない．次元数 D が 10 である時を見てみると，多様性評価指標 r_s, r_f は両者とも集団数 P が 10 のときのみ，DE/A の方が，DE/NA に比べて多様性評価指標 r_s, r_f を高く維持したまま，世代数が大きくなるに連れてその差を広げていった．それに対し集団数 P が 30, 50 である時

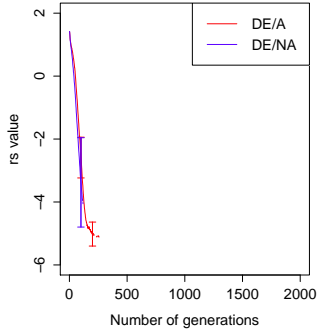
DE/A, DE/NA 同士でほとんど差が見られなかった。次元数 D が 30 の時を見てみると、集団数 P が 30 の時でも、DE/A と DE/NA での差が大きく広がっていることが分かる。次元数 D が 50 の時を見てみると、次元数 D が 30 の時とほとんど変わらない。しかし注目すべき点は集団数 P が 30 である時、次元数 D が 30 であった時に比べ、DE/A と DE/NA の同士の差の広がりが小さくなっている点である。これは多様性評価指標 r_s, r_f の両方をみても同じ傾向が見られる。次元数 D が 100 の時を見ると、集団数 P が 10, 30, 50 と全ての値を取る時で、DE/A と DE/NA の差の広がりが大きく見られた。

次に集団数を軸に述べる。集団数 P が 10 である時の多様性評価指標 r_s, r_f を見てみると、次元数 D が 2 の時は DE/A, DE/NA 同士でほとんど変わらなかったが、次元数 D が 10, 30, 50, 100 の時では大きくその差が開いた。しかしながら、その差の開きを見てみると、ほぼ同程度であった。集団数 P が 30 である時を見てみると、次元数 D が 2, 10, 30 と高くなるに連れて DE/A, DE / NA 同士でその差の開きは大きくなっていったが次元数 D が 30, 50, 100 と高くなっても、差の開きは同程度のものであった。集団数 P が 50 である時を見てみると、次元数 D が 2, 10, 30, 50, 100 と高くなるに連れて、DE/A と DE/NA の差の広がりが大きく見られた。

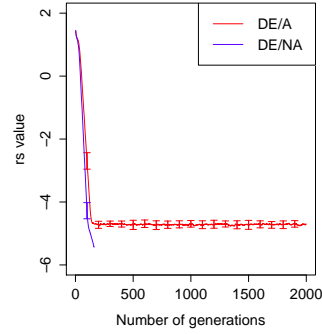
3.4 考察

実験結果から、集団数 P が小さく、次元数 D が高いほど、少なくとも DE の Sphere 関数において、多様性評価指標 r_s, r_f に基づき、アーカイブを使用することで解集団が多様性を維持したまま探索を行えていることがわかった。これらのことからアーカイブが解集団の多様性維持に確かに貢献していること、集団数 P 、次元数 D の設定によってアーカイブの性能が変わってくることが知見として得られた。

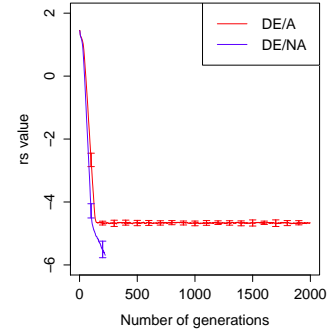
図 3.1 横軸は評価回数の経過を 1 から 2000 世代目まで表示している．縦軸は 51 回試行した多様性評価指標 r_s について、平均値を求めそれに対し、常用対数をとったものである．DE/A と DE/NA の二つのアルゴリズムを用いて、それぞれ次元数 D を 2, 10, 30, 50, 100 とし、集団数 P を 10, 30, 50 とした時の多様性評価指標 r_s が推移する様子を示している．



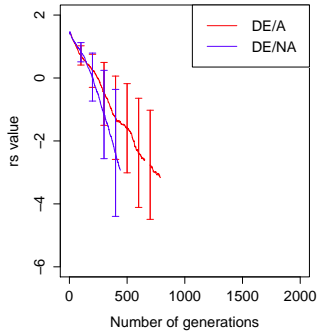
$P = 10, D = 2$



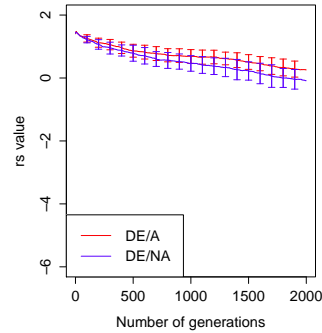
$P = 30, D = 2$



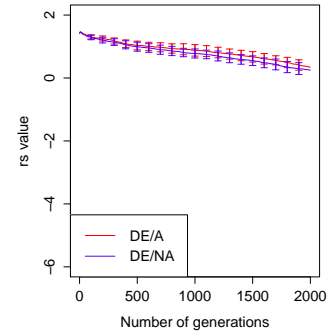
$P = 50, D = 2$



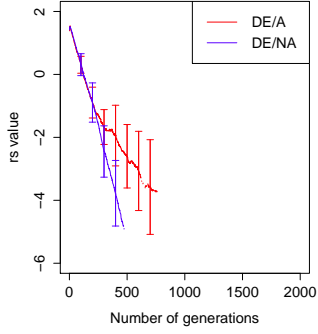
$P = 10, D = 10$



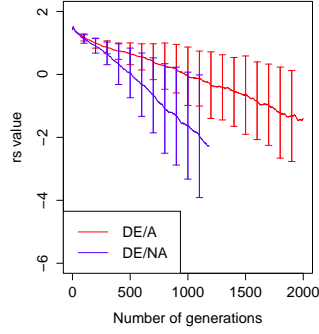
$P = 30, D = 10$



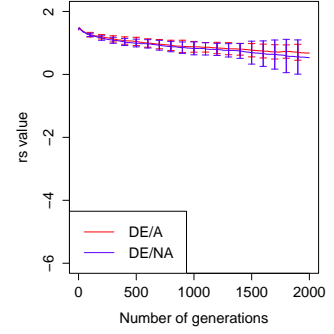
$P = 50, D = 10$



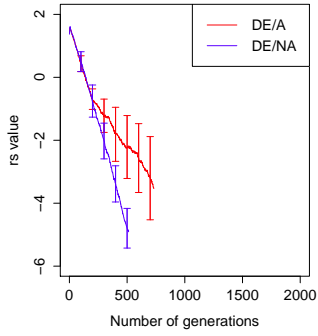
$$P = 10, D = 30$$



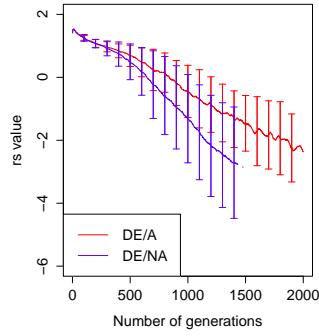
$$P = 30, D = 30$$



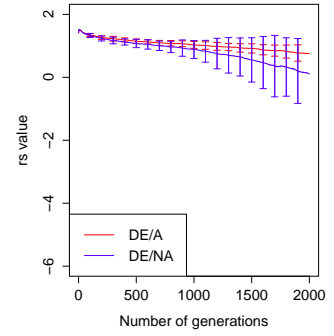
$$P = 50, D = 30$$



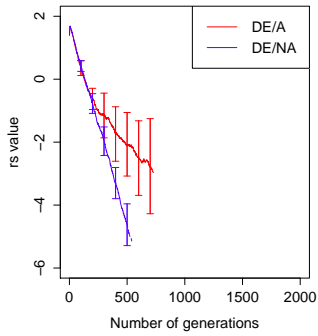
$$P = 10, D = 50$$



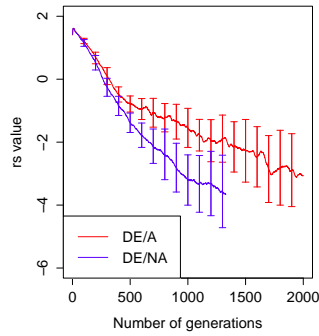
$$P = 30, D = 50$$



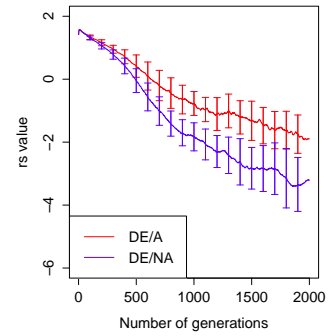
$$P = 50, D = 50$$



$$P = 10, D = 100$$

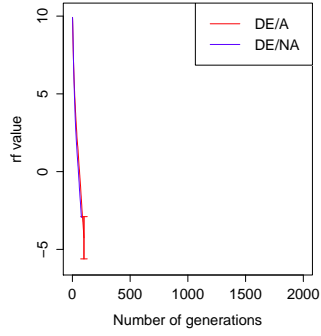


$$P = 30, D = 100$$

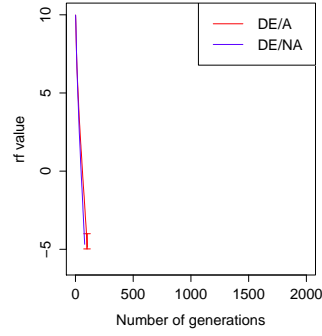


$$P = 50, D = 100$$

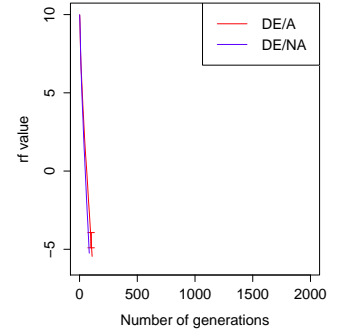
図 3.2 横軸は評価回数の経過を 1 から 2000 世代目まで表示している．縦軸は 51 回試行した多様性評価指標 r_f について、平均値を求めそれに対し、常用対数をとったものである．DE/A と DE/NA の二つのアルゴリズムを用いて、それぞれ次元数 D を 2, 10, 30, 50, 100 とし、集団数 P を 10, 30, 50 とした時の多様性評価指標 r_f が推移する様子を示している．



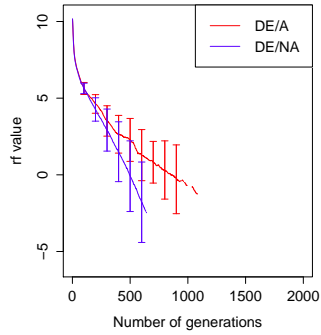
$$P = 10, D = 2$$



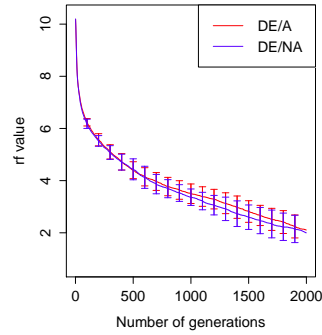
$$P = 30, D = 2$$



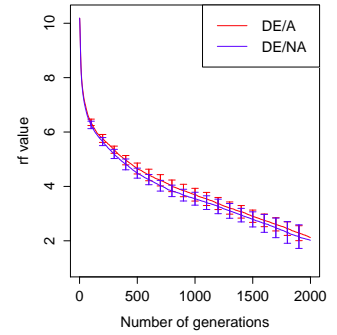
$$P = 50, D = 2$$



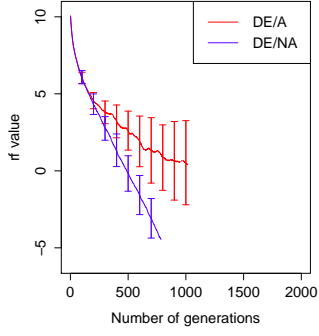
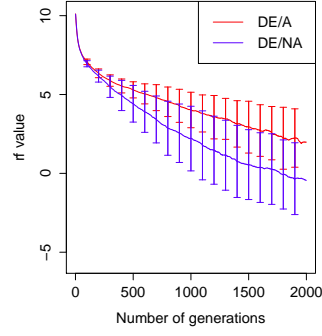
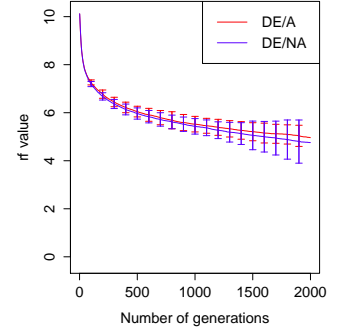
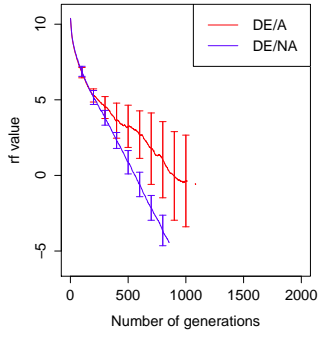
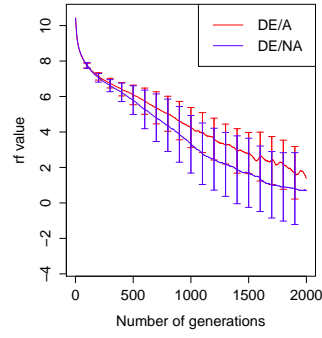
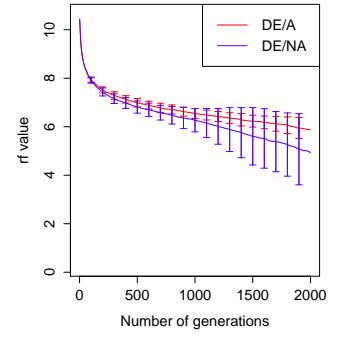
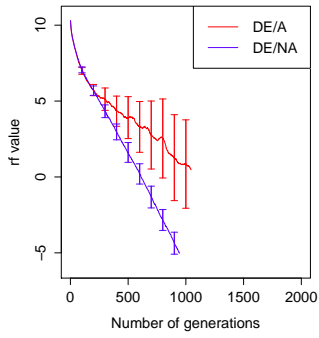
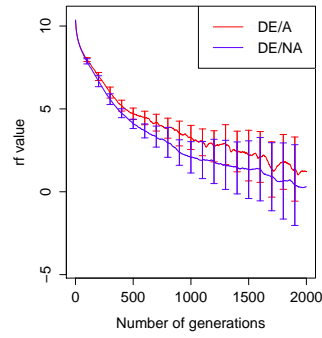
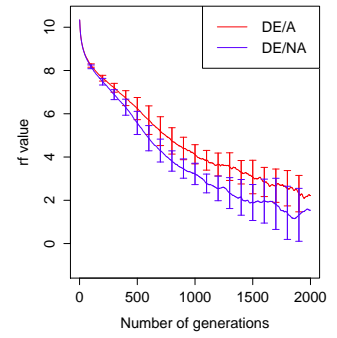
$$P = 10, D = 10$$



$$P = 30, D = 10$$



$$P = 50, D = 10$$


 $P = 10, D = 30$

 $P = 30, D = 30$

 $P = 50, D = 30$

 $P = 10, D = 50$

 $P = 30, D = 50$

 $P = 50, D = 50$

 $P = 10, D = 100$

 $P = 30, D = 100$

 $P = 50, D = 100$

第 4 章

アーカイブ改善の提案

4.1 提案手法 1:アーカイブサイズの廃止

既存手法では、アーカイブのサイズは集団ベクトルと同じサイズだけとり、超過分だけランダムにアーカイブから取りのぞく。このサイズが小さいと、解更新が停滞している場合、すぐに探索の近傍の個体でアーカイブが一杯になり、アーカイブで保持される個体の多様性が失われる。新たに提案する手法では、このアーカイブのサイズを廃止し、探索状況にあった個体であれば全てアーカイブに保存する。

まず解集団 $\mathbf{P} = \{\mathbf{x}^1, \dots, \mathbf{x}^P\}$ における各ベクトルの要素を $\mathbf{x}^k = \{x_1^k, \dots, x_D^k\}$ と表す。このとき、 $k \in \{1, \dots, P\}$ であり、また D は次元数、 P は集団数である。この時、式 (4.1)、式 (4.2) のように現集団中の個体の各変数 $j \in \{1, \dots, D\}$ の下限上限値を $[x_{min,j}, x_{max,j}]^D$ とする。

$$x_{min,j} = \min(x_j^1, x_j^2, \dots, x_j^P) \quad (4.1)$$

$$x_{max,j} = \max(x_j^1, x_j^2, \dots, x_j^P) \quad (4.2)$$

また $\min(\cdot)$, $\max(\cdot)$ はそれぞれ、引数の最小値、最大値を返す関数である。 $\alpha(> 0)$ である時、各世代の終了時に、アーカイブ内の全ての個体について、 $[\alpha * x_{min,j}, \alpha * x_{max,j}]^D$ の範囲に全ての変数値が収まっていれば残し、そうでなければ削除する。こうすることで、アーカイブのサイズがなくても、アーカイブに探索状況にあった多様な個体を保持できるのではと考えられる。Algorithm 5 に全体のアルゴリズムをのせる。

Algorithm 5 提案手法 1:アーカイブサイズの廃止

集団 $P = \mathbf{x}^1, \dots, \mathbf{x}^N$ の初期化;

M_{AR} を 0.5 に初期化;

$k = 1$;

while 探索終了条件を満たしていない **do**

for $i = 1$ to N **do**

 突然変異戦略を用いて変異個体 \mathbf{v}^i を生成;

\mathbf{x}^i と \mathbf{v}^i に交叉を適用し, 子個体 \mathbf{u}^i を生成;

end for

for $i = 1$ to N **do**

if $f(\mathbf{u}^i) \leq f(\mathbf{x}^i)$ **then**

$\mathbf{x}^i \rightarrow \mathbf{u}^i$;

$\mathbf{x}^i = \mathbf{u}^i$;

end if

end for

for $i = 1$ to A **do**

if アーカイブに保持された個体 $\mathbf{x}^{i,A}$ の各要素が現集団中の個体の各変数の下限上限値に収まっている **then**

 アーカイブから $\mathbf{x}^{i,A}$ を削除;

end if

end for

end while

4.2 提案手法 2: 適応的なアーカイブの選択

既存手法では、アーカイブを使用する際は集団ベクトルとアーカイブベクトルの和集合からランダムに個体を選択していた。これをまず、アーカイブを選択する確率 $AR \in [0, 1]$ を導入し、 $[0, 1)$ 区間内の一様乱数より AR が大きければアーカイブから選択、小さければ集団ベクトルから選択するようにする。そしてこの AR を SHADE のスケール係数 F や CR と同じように適応的に探索中に変化させる。まず、大きさ H の履歴メモリ M_{AR} を $M_{AR} = (M_{AR,1}, \dots, M_{AR,H})$ となるようにとる。またすべての要素は探索開始時に 0.5 に初期化されている。集団の各個体 \mathbf{x}^i ごとにアーカイブ選択率 AR_i を持ち、これらのパラメータを各世代のはじめに $[1, H]$ の範囲からランダムに選択したメモリの番号 r の要素 $M_{AR,r}$ を用いて式 (4.3) のようにして生成する。

$$AR_i = \text{randn}(M_{CR,r}, 0.1) \quad (4.3)$$

$\text{randn}(\mu, \sigma^2)$ は平均 μ 、標準偏差 σ^2 の正規分布に従う乱数である。 AR_i の値が $[0, 1]$ 区間より外の場合は超えた方の境界値で置き換えられる。各世代の終了時に成功した AR_i の集合 S_{AR} の Lehmer 平均を用いて、式 (4.4) のように M_{AR} を更新する。

$$M_{AR,k} = \text{mean}_L(S_{AR}) \quad (4.4)$$

ここで、 $k \in [1, H]$ は更新するメモリの要素を決定するパラメタであり、探索開始時に 1 に初期化され、以降更新を行うたびに 1 ずつ増加していく。また、 $k > H$ となった場合は $k = 1$ とする。探索が経過するに連れて M_{AR} には対象問題に適したかつ、多様なパラメタ設定が保持される。こうして適応的にアーカイブを選択する確率を探索中にかえていくことで、より探索性能をあげられると考えられる。Algorithm 6 に全体のアルゴリズムをのせる。

Algorithm 6 提案手法 2:適応的なアーカイブの選択による DE

```

集団  $P = \mathbf{x}^1, \dots, \mathbf{x}^N$  の初期化;
 $M_{AR}$  を 0.5 に初期化;
 $k = 1$ ;
while 探索終了条件を満たしていない do
   $S_F := \emptyset, S_{AR} := \emptyset$ ;
  for  $i = 1$  to  $N$  do
     $r := \text{randi}[1, H]$ 
     $AR_i := \text{randn}(M_{AR}, 0.1)$ 
    突然変異戦略を用いて変異個体  $\mathbf{v}^i$  を生成;
     $\mathbf{x}^i$  と  $\mathbf{v}^i$  に交叉を適用し, 子個体  $\mathbf{u}^i$  を生成;
  end for
  for  $i = 1$  to  $N$  do
    if  $f(\mathbf{u}^i) \leq f(\mathbf{x}^i)$  then
       $\mathbf{x}^i \rightarrow A$ ;
       $\mathbf{x}^i = \mathbf{u}^i$ ;
       $AR_i \rightarrow S_{AR}$ ;
    end if
  end for
  if  $S_{AR} \neq \emptyset$  then
     $M_{AR,k} := \text{mean}_L(S_{AR})$ ;
     $k = (k + 1) \% H$ ;
  end if
  もしアーカイブがアーカイブサイズ  $|A|$  を超えていれば, 超えた分だけランダムに削除;
end while

```

4.3 実験設定

ここでは突然変異戦略を `current-to-pbest/bin/1` 及びパラメタを $F = 0.5$, $CR = 0.5$ とした DE/A と DE/NA と, α を 0.5, 1.0, 1.5, 2.0 とした提案手法 1 によるアーカイブを用いた DE, 提案手法 2 による DE とで比較実験を行う. 評価実験には Black-Box Optimization Benchmarking at CEC'2015 (CEC-BBOB) の 15 個のベンチマーク関数を用いた. $F_1 \sim F_2$ が単峰性関数であり, $F_3 \sim F_9$ が多峰性関数である. $F_{10} \sim F_{15}$ は複数の評価関数を組み合わせた複合関数である. 全ての評価関数において実行可能領域は $[-100, 100]^D$ である. また, 探索中に得られた最良解と最適解との誤差が 10^{-8} 以下になった場合は, 誤差値は 0 となる. ベンチマークの詳細については [7] を参考にいただきたい. 全ての評価関数において次元数は $D = 30$ とし, 1 試行あたりの最大評価回数は $D * 10,000$ とした. また集団数は $P = 30$ である. 試行回数は 51 回とし, この評価回数の平均値がどれほど高い精度の解であるかをもとに手法を評価する. また, 有意水準 0.05 の Wilcoxon の順位和検定を行った.

4.4 実験結果

表 4.1 に CEC2015 ベンチマークセットにおける実験結果をしめす. 表中のデータは各手法中に得られた最良解の目的関数値と最適値との誤差の平均と標準偏差である. また表中の記号 +, -, \approx は DE/A と比べてそれぞれ有意に優れている, 有意に劣っている, 有意差なしを示す. まず提案手法 1 について見てみる. 単峰性の関数 $F_1 \sim F_2$ について見てみると, α が 0.5, 1.0 の時は DE/A に比べて有意に劣っていた. それに対し α が 1.5 の時は関数 F_1 の時に DE/A より有意に劣っていた一方, 関数 F_2 の時に DE/A より有意に優れた結果を示した. α が 2.0 の時は, 関数 F_1 の時に有意差はなかったが, 関数 F_2 の時に DE/A より有意に優れた結果を示した. まとめると, 単峰性関数において, 提案手法 1 の場合, α の値が大きい方が, その性能が向上する傾向が見られた.

多峰性の関数 $F_3 \sim F_9$ について見てみる. 関数 F_3 の時は, DE/NA や提案手法 2 を含める全ての手法で有意な差が出なかった. 同じように全ての手法で有意な差が見られなかった関数として F_{11} があるが, これは関数 F_3 を用いたハイブリッド関数であるため, その特徴が F_{11} にもあらわれたのではと考えられる. 他の多峰性関数 $F_4 \sim F_8$ では α の値が 0.5, 1.0, 1.5 の時は有意な差が DE/A と比べ見られなかったものの, α の値が 2.0 である時のみ, 有意に性能が劣っていた. したがって多峰性関数において提案手法 1 では, α の値が小さい時は, DE/A と変わらず, α の値が大きくなると DE/A より有意に劣る傾向があったといえる.

次は複数の評価関数を組み合わせた複合関数である $F_{10} \sim F_{15}$ について見てみる. 全体的に DE/A より劣る, もしくは有意差がないことが多いが, α が 2.0 のときの関数 F_{14} , F_{15} , α が 1.5 のときの関数 F_{15} で DE/A より有利に優れていた. 複合関数においては α の値が大きいとき DE/A より有意な性能を示すことがあるといえる.

提案手法 2 については, 多峰性の関数 F_4 においてのみ DE/A より有意に優れていたが, 他の関数では有意な差が見られなかった.

4.5 考察

全体としてみると、提案手法 1 は DE/A より劣った性能であったが、パラメタ α の値によっては、DE/A より優れた性能を得ることが出来た。これは問題設定によってはパラメタ α を適切に選ぶことで、優れた可能性を示せることを意味する。したがって提案手法では定数とした α を適応的に選択することでアーカイブ性能の向上に繋がる可能性があると考えられる。

提案手法 2 では DE/A よりわずかに優れた性能を示したものの、殆どの関数において有意な差は得られなかった。これはアーカイブを選択する確率 AR を導入し、それを適応的に変化させたとしても選択するアーカイブに保持された個体自体は従来と全く同じであることが原因ではと考えれる。また DE の他の制御パラメータである F や CR が、変異個体の生成に直接的に関わってくるのに対し、 AR はあくまでアーカイブを選択する確率でしかないので、それを適応的に変化させてもあまり影響はないのかもしれない。

というのも提案手法 1 では DE/A より劣る性能となったが、アーカイブに保持される個体が従来の手法と変わったため、有意な差がより生じたものと考えられる。それに対し、提案手法 2 ではアーカイブに保持される個体は既存手法と変わらずその取り方のみを適応的に変化させた。その結果わずかに DE/A より優れた性能を示したもののほとんど有意な差を示さなかったのではと考えられる。

今後アーカイブの改善を目指すには、アーカイブに保持される個体自体を従来の手法と変えていくことが有用なのではないかということが今回の実験の知見として得られた。

表 4.1 CEC2015 ベンチマークセットにおける, DE/A と提案手法の比較実験の結果. 全てのテスト関数の次元数 D は 30 次元であり, 最大評価回数は, $10,000 \times D$ である. また全てのデータは 51 回の試行の平均である. 各セルの中身は得られた最良解と最適値の誤差の平均と標準偏差である.

F	DE/A	DE(提案手法 1) $alpha=(0.5)$	DE(提案手法 1) $alpha=(1.0)$	DE(提案手法 1) $alpha=(1.5)$	DE(提案手法 1) $alpha=(2.0)$	DE(提案手法 2)	DE/NA
F_1	4.76e+05(3.65e+05)	2.08e+06(1.75e+06) -	2.02e+06(2.23e+06) -	3.23e+05(4.63e+05) +	1.00e+06(1.10e+06) \approx	6.10e+05(5.55e+05) \approx	2.46e+06(1.90e+06) -
F_2	3.27e+05(1.86e+06)	2.09e+08(2.97e+08) -	2.18e+08(2.80e+08) -	8.81e+05(6.18e+06) -	9.21e+02(2.13e+03) +	3.73e+03(3.74e+03) \approx	2.15e+08(2.70e+08) -
F_3	2.08e+01(5.49e-02)	2.08e+01(5.66e-02) \approx	2.08e+01(4.63e-02) \approx	2.08e+01(5.58e-02) \approx	2.08e+01(5.26e-02) \approx	2.08e+01(4.52e-02) \approx	2.09e+01(5.05e-02) \approx
F_4	1.19e+02(1.28e+01)	1.21e+02(1.51e+01) \approx	1.19e+02(1.36e+01) \approx	1.22e+02(1.78e+01) \approx	1.42e+02(2.17e+01) -	1.10e+02(1.74e+01) +	1.20e+02(1.33e+01) \approx
F_5	5.67e+03(3.45e+02)	5.76e+03(4.29e+02) \approx	5.73e+03(3.47e+02) \approx	5.79e+03(4.33e+02) \approx	5.95e+03(3.48e+02) -	5.75e+03(4.18e+02) \approx	5.80e+03(3.61e+02) -
F_6	2.88e+04(1.94e+04)	3.73e+04(2.83e+04) \approx	3.47e+04(2.09e+04) \approx	3.16e+04(1.83e+04) \approx	1.17e+05(1.10e+05) -	3.01e+04(2.22e+04) \approx	3.74e+04(3.06e+04) \approx
F_7	1.06e+01(3.13e+00)	1.05e+01(2.63e+00) \approx	1.10e+01(2.06e+00) \approx	1.07e+01(2.18e+00) \approx	1.15e+01(2.71e+00) -	1.07e+01(2.04e+00) \approx	1.08e+01(2.31e+00) \approx
F_8	8.41e+03(7.45e+03)	9.46e+03(7.99e+03) \approx	1.05e+04(9.02e+03) \approx	8.69e+03(7.71e+03) \approx	1.60e+04(1.33e+04) -	5.60e+03(4.94e+03) \approx	1.08e+04(9.70e+03) \approx
F_9	1.17e+02(5.23e+01)	1.03e+02(6.25e-01) +	1.07e+02(3.09e+01) +	1.07e+02(3.14e+01) \approx	1.06e+02(2.97e+01) \approx	1.06e+02(2.87e+01) \approx	1.03e+02(1.21e+00) +
F_{10}	6.01e+03(6.56e+03)	9.96e+03(1.28e+04) -	1.06e+04(1.07e+04) -	8.84e+03(1.15e+04) -	2.08e+04(1.92e+04) -	5.02e+03(3.22e+03) \approx	2.17e+04(7.46e+04) -
F_{11}	5.18e+02(9.52e+01)	5.21e+02(1.17e+02) \approx	5.23e+02(1.15e+02) \approx	5.08e+02(8.71e+01) \approx	5.06e+02(9.83e+01) \approx	5.30e+02(9.22e+01) \approx	5.15e+02(1.10e+02) \approx
F_{12}	1.05e+02(8.68e-01)	1.06e+02(1.23e+00) -	1.06e+02(1.06e+00) -	1.06e+02(8.23e-01) \approx	1.06e+02(1.03e+00) -	1.05e+02(7.97e-01) \approx	1.06e+02(1.04e+00) \approx
F_{13}	1.12e+02(3.87e+00)	1.13e+02(3.52e+00) \approx	1.13e+02(4.39e+00) \approx	1.17e+02(4.80e+00) -	1.18e+02(5.08e+00) -	1.14e+02(4.19e+00) \approx	1.13e+02(4.08e+00) \approx
F_{14}	3.36e+04(1.70e+03)	3.42e+04(1.81e+03) \approx	3.46e+04(2.13e+03) -	3.35e+04(1.50e+03) \approx	3.31e+04(1.67e+03) +	3.35e+04(1.68e+03) \approx	3.48e+04(1.78e+03) -
F_{15}	1.02e+02(3.89e+00)	1.20e+02(1.19e+01) -	1.17e+02(9.64e+00) -	1.00e+02(0.00e+00) +	1.00e+02(8.61e-03) +	1.02e+02(3.59e+00) \approx	1.22e+02(1.40e+01) -

第 5 章

終わりに

5.1 まとめ

本論文では、解集団における重心からの距離の標準偏差と、目的関数値 $f(\mathbf{x})$ の標準偏差をしめす多様性評価指標 r_s, r_f を用いることで、アーカイブが解集団における多様性の維持に確かに貢献していることを示すことができた。またアーカイブによる多様性維持は、次元数 D が高く、集団数 P が小さい時ほど、大きな影響をあたえることが分かった。

次にアーカイブを改善するための手法としてアーカイブサイズを廃止し、探索にあった個体のみをアーカイブに保持する提案手法 1 と、アーカイブを選択する確率 AR を定義し、 AR を適応的に変化させる提案手法 2 を提案し、その性能を評価する実験を行った。

提案手法 1 は DE/A より劣った性能であったが、パラメタ $alpha$ の取り方によっては、DE/A より優れた性能を得ることもあった。このためパラメタ $alpha$ を定数にするのではなく、適応的に選択することでさらなる提案手法 1 の改善が見込めるかも知れない。

また提案手法 2 では DE/A よりわずかに優れた性能を示したものの、殆どの関数において有意な差は得られなかった。これはアーカイブを選択する確率 AR を適応的に変化させたとしても選択するアーカイブの個体自体は従来手法と同じであることが原因であるのではと考えられる。

これら二つの提案手法の結果から、アーカイブの性能を向上させるには、アーカイブを選択する確率を変化させることよりも、アーカイブに保持される個体自体の取り方を変えた方がよいのではないだろうかということが知見として得られた。

今回得られた知見を通し、さらなるアーカイブ性能の向上を目指していきたい。

5.2 謝辞

研究室の先輩方にはミーティングや中間発表を初め有益なアドバイスをずっと頂きました。指導教員となる福永先生には、お忙しい中必ず週に一度の個別ミーティングの機会を与えてくださり、とても丁寧にご指導していただきました。研究とはどういうものか、どのように進めれば良いのか右も左もわかっていなかった自分にそれらを一から教えてくれたことに深く感謝いたします。また研究室の先輩の中でも、特に田邊さんには、有益なアドバイスを幾つもしていただきました。論文の書き方について、全くわかって

いなかった自分に細かくご指導いただきました。常に 1 を聞くと 10 のことを教えてくださり，その DE 研究における知見の深さに，いつも脱帽していました。改めてここで感謝の意を示します。本当に素晴らしい先輩や指導教員に恵まれた中，卒業論文を執筆できたことに幸福を感じます。修士過程でも一つでも多くのことを学ばせていただきたいと思います。

参考文献

- [1] R.Storn and K. Price. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341-359, 1997
- [2] K.V.Price, R. M.Storn and J.A. Lampinen.Differential Evolution - A Practical Approach to Global Opticization. Springer, Berlin Heidelberg, 2005.
- [3] J. Zhang, V. Avasarala, A. C. Sanderson, and T. Mullen, "Differential evolution for discrete optimization: An experimental study on combinatorial auction problems," in *IEEE CEC*, 2008, pp. 27942800.
- [4] J. Zhang and A. C. Sanderson: JADE: Adaptive DifferentialEvolution With Optional External Archive,IEEE Tran. Evol.Comput.,135, 945/958 (2009)
- [5] R. Tanabe and A. Fukunaga: Success-History Based Param-eter Adaptation for Differential Evolution,Proceedings of the2013 IEEE Congress on Evolutionary Computation, 71/78(2013)
- [6] S. Das and P. N. Suganthan, "Differential evolution: A survey of the e-oe-art," *IEEE Tran. Evol. Comput.*, vol. 15, no. 1, pp. 431, 2011.
- [7] Q. Chen, B. Liu, Q. Zhang, J. J. Liang, P. N. Suganthan, B. Y. Qu, "Problem Definition and Evaluation Criteria for CEC 2015 Special Session and Competition on Bound Constrained Se-Objective Computationally Expensive Numerical Optimization", Technical Report, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Technical Report, Nanyang Technological University, Singapore, Nov 2014.
- [8] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Tran. Evol. Comput.*, vol. 13, no. 2, pp. 398417, 2009.
- [9] J. Brest and M. S. Maucec, "Population size reduction for the differ- ential evolution algorithm," *Applied Intelligence*, vol. 29, no. 3, pp. 228247, 2008.