差分進化型改良ABCアルゴリズム

○香川　卓哉*1　土方　拓也*1
宇谷　明秀*2

キーワード：高次元最適化問題　多峰性関数　大域的最適解　差分進化　ABC アルゴリズム

1．はじめに

実システムの大規模化・複雑化に伴い，多くの工学設計問題が多数の局所解を有する高次元連続型多峰性関数の最適化問題として定式化されるようになってきた．特に文献[1]，[2]のような自律分散型ネットワークの設計問題は設計変数間に依存関係はないか，今後の大規模領域でのネットワーク運用を考慮に入れると設計変数(最適化問題の次元数)は数百のオーダになることが想定される．筆者らは既往の研究において高次元最適化問題に対する解探索性能に優れたArtificial Bee Colony(ABC)アルゴリズム[3]の改良法を提案し，代表的なベンチマーク関数を用いた既往の手法との性能比較実験を通してその有効性を検証している[4]．しかし，数百次元の工学設計問題(高次元最適化問題)への運用前提とした場合，更なる解探索性能の改善(強化)が必要である．本論文では，文献[4]の手法に差分進化アルゴリズム[5]の変異ベクトル生成式を導入した差分進化型改良ABCアルゴリズムを提案する．提案手法の有効性は大域的最適解に匹敵する準最適解が解探索領域内に分布する高次元連続型多峰性関数のベンチマーク問題に対する数値実験を通して検証される．

2．ABCアルゴリズムの改良法と提案手法

以下，ABCアルゴリズムの改良法[4]について概説した後，提案手法に加えた改善策(差分進化アルゴリズムの変異ベクトル生成式の導入)，及びその具体的な解探索手順について述べる．

2．1　ABCアルゴリズムの改良法

文献[4]の手法のベースとなるABCアルゴリズムは多次元解探索空間に配置された探索点と３種類の探索群(employed bees，onlookers，scouts)を基本要素として構成されている．ここで，各employed beesはある一つの探索点と関係づけられており，解探索過程で関連づけられた探索点の更新を試みる．onlookersによる解探索では相対的に価値の高い探索点の更新が繰返し試みられる．scoutsは遺伝的アルゴリズムにおける突然変異に相当する役割を担っている．しかし，ABCアルゴリズムには幾つかの問題点が内在している．文献[4]で指摘した問題点を以下に示す．

・適合度算出方法
・onlookersによる探索における探索点の選択方法
・参照点(更新を求める際の基準点)の選択方法
・scoutsによる探索における探索実行条件

文献[4]の手法では上記四つの問題点に対する改善策が導入され，高次元最適化問題に対する解探索性能の向上を実現している．しかし，解探索の過程において，各探索点の更新候補点は，各探索点の位置を基準に，その位置からの更新量を求めることで算出される．よって，大規模な工学設計問題(数百次元の最適化問題)へ適用することを想定した場合，初期値依存性の問題により，解探索性能が低下する可能性がある．数百次元の最適化問題への適用にあたっては更なる解探索性能の向上が期待される．

2．2　提案手法

連続型多峰性関数の最適化問題(実数値最適化問題)を対象にした差分進化型アルゴリズム[5]は進化論的計算手法に分類される最適化手法であり，交差率などのパラメータ設定や変異ベクトル生成式の選択などによって解探索性能が大きく変化する．しかし，これらを適切に設定(チューニング)することができれば有力な最適化手法になり得るポテンシャルを有している．本研究では，大域的探索性向を有し，また多峰性関数の最適化問題において最も効果のある以下の変異ベクトル生成式[5]をemployed beesの探索，すなわちそれらの更新候補点の算出のために利用する．

$$\boldsymbol{v}_i = \boldsymbol{x}_{r_1} + F(\boldsymbol{x}_{r_2} - \boldsymbol{x}_{r_3}) \qquad (i = 1, \ldots, SN) \tag{1}$$

ここで，$\boldsymbol{v}_i\,(i = 1,\ldots, SN)$は各個体$\boldsymbol{x}_i\,(i = 1,\ldots, SN)$の変異ベクトルを表す．$r_1$, r_2, r_3は個体番号(i)以外の中からランダムに選択された互いに異なる個体番号であり，SNは個体総数，Fは増幅率と呼ばれる更新量制御パラメータである．この変異ベクトル生成式の導入により，各探索点に関係づけられた各employed beesの解探索において，探索点の初期配置の影響が緩和され，解探索空間内をより大域的に探索できるようになる．emplpyed beesの大域的探索能力の向上とonlookersによる相対的に価値の高い探索点の集中的な局所探索によって，提案手法の高次元最適化問題に対する解探索性能は文献[4]の手法よりも向上すると考えられる．

D変数の最小化問題

$\min. f(\boldsymbol{x})$,　subj. to $\boldsymbol{x} \in R^D$

を考える．目的関数[$f(\boldsymbol{x})$]が与えられたとき，$f(\boldsymbol{x})$の値を最小にする$\boldsymbol{x}$の値を求める最適化問題に対する提案手法の

具体的な解探索手順を以下に示す．

［**Step 0**（準備）］

・コロニーサイズ（N）と探索点の総数（SN）を設定する．

・適合度上位の探索点の数（α）を設定する．

・総繰返し回数（T_{max}）を設定する．

・解への収束状況判定パラメータ（dr）を設定する．

・許容限界値（f_{bound}）を設定する．

・f_{bound}に対する精度値（$f_{accuracy}$）を設定する．

・総繰返し回数（T_{max}）を設定する．

・探索段階１の最低繰返し回数（$T1_{min}$）を設定する．

・更新量制御パラメータ（F）を設定する．

［**Step 1**（初期化）］

・繰返し回数のカウンタを$k=1$にする．

・探索段階２への切替判定値を初期化する（$f_{judge}=0$）．

・各探索点の初期位置ベクトル（$\boldsymbol{x}_i^1$）を乱数によって生成する．ここで，下付$i\in\{1, 2, \dots, SN\}$は探索点の番号を表し，上付（$k=$）1は繰返し回数を表す．

・初期状態における全体最良解（$\boldsymbol{best}^1$）を決定する．

$$i_b = arg\min_i f(\boldsymbol{x}_i^1), \quad i=1,\dots,SN$$

$$\boldsymbol{best}^1 = \boldsymbol{x}_{i_b}^1$$

・初期状態における平均評価値（f_{init}）を計算する[4]．

［**Step 2**（employed beesによる解探索）］

1）全ての探索点[$\boldsymbol{x}_i^k$(i=1,…,SN)]に対して，それらの更新候補点[$\boldsymbol{v}_i^k$($i=1,\dots,SN$)]を生成する．

$$v_{ih}^k = x_{r_1h}^k + F(x_{r_2h}^k - x_{r_3h}^k), \quad i=1,\dots,SN$$

$$v_{ij}^k = x_{ij}^k, \quad i=1,\dots,SN$$

ここで，r_1, r_2, r_3は探索点番号（i）以外の中からランダムに選択された互いに異なる探索点番号を表す．また，$h\in\{1, 2, \dots, D$（次元数）$\}$は探索点ごとにランダムに選択された一つの変数番号を表し，$j\in\{1, 2, \dots, D\}$は選択された番号（h）以外の残りの設計変数番号を表す．

2）各探索点（$\boldsymbol{x}_i^k$）を更新する．

$$I_1 = \left\{ i \mid f(\boldsymbol{v}_i^k) < f(\boldsymbol{x}_i^k), \quad i=1,\dots,SN \right\}$$

とし，次のように更新する．

$$\boldsymbol{x}_i^k = \begin{cases} \boldsymbol{v}_i^k, & i \in I_1 \\ \boldsymbol{x}_i^k, & i \notin I_1 \end{cases}$$

［**Step 3**（onlookersによる解探索）］

1）onlookersの探索カウンタを$l=1$にする．

2）各探索点（$\boldsymbol{x}_i^k$）の適合度（fit_i^k）を計算する．

$$fit_i^k = \begin{cases} \dfrac{1}{f(\boldsymbol{x}_i^k) - f_{bound}}, & f(\boldsymbol{x}_i^k) - f_{bound} \geq f_{accuracy} \\ \dfrac{1}{f_{accuracy}}, & f(\boldsymbol{x}_i^k) - f_{bound} < f_{accuracy} \end{cases}$$

$$(i=1,\dots,SN)$$

3）$k < T1_{min}$であれば下記5）へ行く．

そうでない場合（$k \geq T1_{min}$）において，$f_{judge} \geq dr$であれば下記4）へ行き，そうでない場合は次式によってf_{judge}の値を更新する．

$$f_{judge} = \frac{f_{init} - f(\boldsymbol{best}^k)}{f_{init} - f_{bound}}$$

そして，この更新値が$f_{judge} \geq dr$になれば下記4）へ行き，そうでない場合は下記5）へ行く．

4）各探索点（$\boldsymbol{x}_i^k$）の相対価値確率（P_i^k）を算出する．

$$P_i^k = fit_i^k \Bigg/ \sum_{n=1}^{SN} fit_n^k$$

5）（$k \geq T1_{min}$）&（$f_{judge} \geq dr$）の場合は相対価値確率（P_i^k）に基づくルーレット選択から一つの探索点（$\boldsymbol{x}_c^k$）を選択し，そうでない場合は適合度の高い上位αの探索点の中からランダムに一つの探索点（$\boldsymbol{x}_c^k$）を選択する．ここで，$c\in\{1, 2, \dots, SN\}$は選択された探索点番号を表す．そして，この選択された探索点（$\boldsymbol{x}_c^k$）についてのみ，その更新候補点（$\boldsymbol{v}_c^k$）を生成する．

$$v_{ch}^k = x_{ch}^k + \phi_{ch}^k(x_{ch}^k - x_{mh}^k)$$

$$v_{cj}^k = x_{cj}^k$$

ここで，$h\in\{1, 2, \dots, D\}$ランダムに選択された一つの設計変数番号を表し，$j\in\{1, 2, \dots, D\}$は選択された番号（h）以外の残りの設計変数番号を表す．またϕ_{ij}^kは[-1,1]の一様乱数であり，$m\in\{1, 2, \dots, SN\}$については，（$k \geq T1_{min}$）&（$f_{judge} \geq dr$）の場合は探索点番号（c）以外で相対価値確率（P_i^k）に基づくルーレット選択によって選択された探索点番号を表し，そうでない場合は探索点番号（c）以外で適合度の高い上位αの探索点の中からランダムに選択された探索点番号を表す．

6）探索点（$\boldsymbol{x}_c^k$）を更新する．

$$\boldsymbol{x}_c^k = \begin{cases} \boldsymbol{v}_c^k, & f(\boldsymbol{v}_c^k) < f(\boldsymbol{x}_c^k) \\ \boldsymbol{x}_c^k, & f(\boldsymbol{v}_c^k) \geq f(\boldsymbol{x}_c^k) \end{cases}$$

7）$l=N\text{-}SN$であれば[Step4]へ行く．

そうでなければ$l=l+1$として上記5）へ戻る．

［**Step4**（全体最良解の更新）］

全体最良解を更新する．

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{x}_i^k, \quad i=1,\dots,SN$$

$$i_b = arg\min_i f(\boldsymbol{x}_i^{k+1})$$

$$\boldsymbol{best}^{k+1} = \boldsymbol{x}_{i_b}^{k+1}$$

［**Step5**（終了判定）］

$k=T_{max}$であれば解探索を終了する．

そうでなければ$k = k + 1$として**Step2**へ戻る．

３．数値実験

文献[1]，[2]の自律分散型ネットワークの設計問題は高次元ではあるが，設計変数間に依存関係のない最適化問題である．本研究ではこのような設計問題への提案手法の適用を考慮し，変数間に依存関係のない次の2つの代表的なベンチマーク関数を用いて，提案手法の高次元最適化問題に対する有効性を調査した．これらの2関数は大域的最適解に匹敵する準最適解が解探索領域内に分布する連続型多峰性関数である．

・Rastrigin関数

$$\min.\ f_2(\boldsymbol{x}) = \sum_{j=1}^{D}\left\{x_j^2 - 10\,cos(2\pi x_j) + 10\right\}$$

$$\text{subj. to}\ \ -5.12 \le x_j \le 5.12, \quad j = 1,\ldots,D$$

$$\boldsymbol{x}^* = (0,\ldots,0), \quad f_2(\boldsymbol{x}^*) = 0$$

・Schwefel関数

$$\min.\ \ f_3(\boldsymbol{x}) = 418.98288727\,D + \sum_{j=1}^{D} -x_j\,sin(\sqrt{|x_j|})$$

$$\text{subj. to}\ \ -512 \le x_j \le 512, \quad j = 1,\ldots,D$$

$$\boldsymbol{x}^* = (420.968750,\ldots,420.968750), \quad f_3(\boldsymbol{x}^*) = 0$$

ここで，各問題の解析的に求められた大域的最適解は$\boldsymbol{x}^*$，それに対応する目的関数の値は$f_1(\boldsymbol{x}^*)$, $f_2(\boldsymbol{x}^*)$としてまとめられている．これら多峰性関数の１変数の場合の形状を図1に示す．図1から，Rastrigin関数[$f_1(\boldsymbol{x})$]，Schwefel関数[$f_2(\boldsymbol{x})$]は局所解の谷が深いことを確認することができる．なお，本実験では既往の手法との性能比較を行った．比較対象手法を以下に列挙する．

1）粒子群最適化(PSO)[6]アルゴリズムに対して，大域探索と局所探索をバランスするためのパラメータ調整機能が導入された改良法[4]
2）差分進化アルゴリズム[5]
3）ABCアルゴリズム[3]
4）高次元問題に対する解探索性能に優れたABCアルゴリズムの改良法(Advanced ABC)[4]

文献[4]に倣い，集団サイズを60とした場合の実験結果を以下に示す．以下に示す実験結果において，上記1)～4)の手法はそれぞれPSO，DE，ABC，AABCと表記されている．ここで, 各手法のパラメータ等の実験設定値は文献[4]で用いた値(各手法において良好な結果が得られる値)を採用している．また，提案手法における更新量制御パラメータはRastrigin関数ではF = 0.2，Schwefel関数ではF = 0.1に設定している[7]．

二つのベンチマーク関数のそれぞれに対する高次元の場合(D = 50～150)の実験結果(繰返し回数:1,000)を表1と表2示す．ここで，表中のBestは50試行の実験における最良解の目的関数値，Ave. は50試行の実験における目的関数の平均収束値，Worstは50試行の実験の結果最悪解の目的関数値を表す．結果を確認すると提案手法を用いた場合，

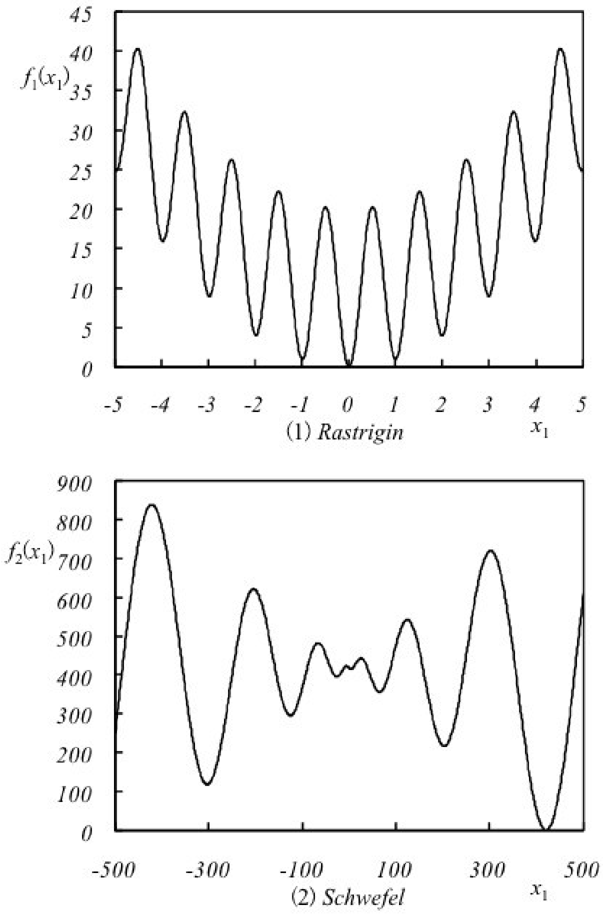

図１　各関数の形状(D = 1)

表1　Rastrigin関数に関する実験結果の一覧

Method	Dim.	Best	Ave.	Worst
PSO	50	1.05×10^{2}	1.81×10^{3}	3.14×10^{3}
	100	2.72×10^{2}	3.71×10^{3}	6.10×10^{3}
	150	3.54×10^{2}	6.60×10^{3}	9.31×10^{3}
DE	50	7.62×10^{-2}	4.55×10^{0}	4.23×10^{1}
	100	1.54×10^{2}	2.21×10^{2}	3.22×10^{2}
	150	2.98×10^{2}	5.59×10^{2}	7.38×10^{2}
ABC	50	4.51×10^{-4}	3.53×10^{0}	1.25×10^{1}
	100	6.22×10^{1}	7.16×10^{1}	1.02×10^{2}
	150	1.28×10^{2}	2.77×10^{2}	3.22×10^{2}
AABC	50	2.08×10^{-10}	8.10×10^{-1}	2.99×10^{0}
	100	1.21×10^{1}	1.39×10^{1}	2.92×10^{1}
	150	4.67×10^{1}	5.66×10^{1}	9.53×10^{1}
Proposal	50	3.23×10^{-11}	4.01×10^{-1}	1.99×10^{0}
	100	9.96×10^{-1}	2.88×10^{0}	9.50×10^{0}
	150	3.66×10^{1}	5.31×10^{1}	6.30×10^{1}

表2　Schwefel関数に関する実験結果の一覧

Method	Dim.	Best	Ave.	Worst
PSO	50	5.39×10^{3}	9.78×10^{3}	1.25×10^{4}
	100	1.92×10^{4}	6.92×10^{4}	9.88×10^{4}
	150	3.35×10^{4}	4.07×10^{5}	5.65×10^{5}
DE	50	1.14×10^{3}	2.24×10^{3}	4.62×10^{3}
	100	1.24×10^{4}	3.45×10^{4}	4.10×10^{4}
	150	5.72×10^{4}	6.70×10^{4}	7.54×10^{4}
ABC	50	8.98×10^{2}	1.30×10^{3}	1.97×10^{3}
	100	6.02×10^{3}	9.31×10^{3}	1.00×10^{4}
	150	9.60×10^{3}	1.62×10^{4}	3.20×10^{4}
AABC	50	1.18×10^{2}	2.03×10^{2}	1.18×10^{3}
	100	2.54×10^{3}	3.03×10^{3}	5.26×10^{3}
	150	6.78×10^{3}	9.02×10^{3}	9.91×10^{3}
Proposal	50	0.00	6.31×10^{1}	2.30×10^{2}
	100	2.03×10^{-2}	1.48×10^{2}	4.27×10^{2}
	150	1.85×10^{3}	2.53×10^{3}	3.49×10^{3}

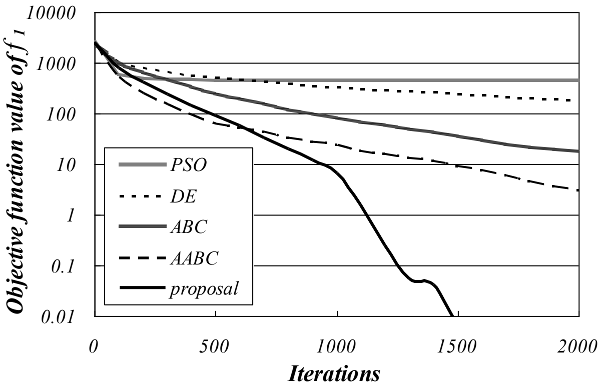

図1　収束曲線［Rastrigin関数（$D=100$）］

Rastrigin関数及びSchwefel関数の全次元の全ての実験結果（最良値/平均値/最悪値）で既往の手法よりも良好な結果が得られている．図1は各手法の収束過程を比較した実験結果（収束曲線）の一例であり，Rastrigin関数（$D=100$）の繰返し回数2,000までの平均値の収束過程が示されている．提案手法ではemployed beesの探索に差分進化アルゴリズムの変異ベクトル生成式を導入し，employed beesの大域的探索性向を強化したために，その解探索性能は探索の前段階ではAABCより若干劣っている．しかし，その後は解探索が順調に進み，最終的には約1,500回の繰返し計算で大域的最適解が求まっている．提案手法の高次元最適化問題に対する有効性及び既往の手法に対する優位性を確認することができる．

4．まとめ

本論文では，高次元連続型多峰性関数の最適解として定式化される工学設計問題に対する解探索手法として，差分進化型改良ABCアルゴリズムを提案した．また大域的最適解に匹敵する準最適解が解探索領域内に分布する二つのベンチマーク関数（二つの高次元連続型多峰性関数）に対する数値実験を通して，提案手法の既往の手法に対する優位性を確認した．提案手法はセンサネットワーク[1]，[2]，[8]をはじめ，サプライチェーンネットワークやスマートグリッド[9]，[10]などの分野の高次元設計問題に対して効果的に用いることができる．これらの分野への提案手法の導入（適用法）に関しては今後の研究課題としたい．

［参考文献］

1）宇谷明秀，山本尚生，”複数の許容解を探索するParticle Swarm Optimizationとその複数シンク無線センサネットワークにおけるシンクノード配置問題への適用，”信学論（D），vol.J93-D，no.5，pp.555-567，May 2010.
2）長島淳也，宇谷明秀，山本尚生，”複数許容解探索型粒子群最適化手法の無線センサネットワークへの適用―フラッディング効率化のための各センサノードの送信電力調整，”日本知能情報ファジィ学会誌，vol,23，no.1，pp.65-77, 2011.
3）D. Karaboga and B. Basturk, “A powerful and efficient algorithm for numerical function optimization: artificial bee colony（ABC）algorithm,” J. Global Optimization, vol.39, pp.459-471, 2007.
4）宇谷明秀，長島淳也，午膓隆太，山本尚生，“Artificial Bee Colony（ABC）アルゴリズムの高次元問題に対する解探索性能の強化，信学論（D），vol.J94-D, no.2, pp.425-438, Feb. 2011.
5）R. Storn and K. Price,”Differential evolution-A simple and effic ient　heuristic for global optimization over continuous space,” J, Global Optimization, vol,11，pp.341-359，1997.
6）M. Clerc and J. Kennedy, “The particle swarm-explosion, stabili -ty, and convergence in a multidimensional complex space,” IEEE Trans. Evol. Comput., vol.6, no.1, pp.58-73, 2002.
7）M. Clerc and J. Kennedy, “The particle swarm-explosion, stability, and convergence in a multidimensional complex space,” IEEE Tra- ns. Evol. Comput., vol.6, no.1, pp.58-73, 2002.
8）石塚美加，会田雅樹，“センサネットワークにおける耐故障性の高い確率的配置の実現，”　信学論（B），vol. J88-B, no. 11，pp. 2181-2191，Nov．2005.
9）西　竜志，”サプライチェーンにおける分散協調型最適化技術，”　人工知能学会誌, vol, 19, no. 5, pp, 571-578, 2004.
10）谷口賀則，平山勝敏，”複数供給源からの分散協調型エネルギー供給量決定プロトコル，”　第25回人工知能学会全国大会論文集（CD-ROM），no.1F1-2，2011.

＊1　東京都市大学工学研究科　博士前期課程
＊2　東京都市大学　准教授　博士（工学）