

花火アルゴリズムへの偵察戦略導入

余俊[†], 譚營^{††}, 高木英行[‡]

九州大学[†], 北京大学/九州大学^{††}, 九州大学[‡]

1 はじめに

これまで多くの進化計算アルゴリズムが提案され, それらの性能を更に向上させる改善手法も多く提案されてきた^{2, 3, 4)}. 改善の手法は, 共分散行列適応進化戦略 (CMA-ES)⁶⁾ や差分進化¹³⁾ の差分ベクトルのように個体群の分布情報を次世代個体生成に利用する方法, アルゴリズムパラメータを適応的に変える方法^{2, 10)}, 広域探索から局所探索への移行を制御する方法¹⁷⁾, 大局的最適化領域を推定し探索方向にバイアスを掛ける方法^{8, 9, 16)}, 山登り法, 焼きなまし法, タブーサーチ, 貪欲法等の各種メタヒューリスティック手法と組み合わせる方法^{1, 5, 11, 12)}, など多岐にわたる.

本論文は, 進化計算の性能向上手法の提案であり, 特に, 個体の局所探索能力向上方法の提案とその評価を目的としている. 局所領域の個体群が次にどちらの方向に向かうべきかを, まずは偵察する探索を行い, その後に個体群が有望方向に移動するよう個体生成を行う偵察戦略が第1の提案である. さらに本論文では, 第2の提案として花火アルゴリズム用にフィルタリング戦略も提案する. この戦略は花火個体 (親個体に相当) よりも fitness 値が悪いスパーク個体を取り除いて次世代個体を決定することで, 無駄な領域を探索しないようにする.

2 花火アルゴリズム

大空で炸裂しその周辺に多くのスパークを生成する花火の炸裂は, ある点の周りの局所探索モデルと類似させることができる. 花火アルゴリズムは炸裂過程繰り返し模擬して, 大局的最

適解を探そうとするアルゴリズムである. Fig. 1 は, 花火アルゴリズムの炸裂モデルを示している. fitness の良い花火個体周辺には狭い範囲に多くのスパーク個体を生成して局所探索能力を高め, そうでない花火個体周辺には広い範囲に少ないスパーク個体を生成して広域探索の役割を増やす. Algorithm 1 に炸裂, 突然変異, 選択の主要 3 演算からなる花火アルゴリズムを示す.

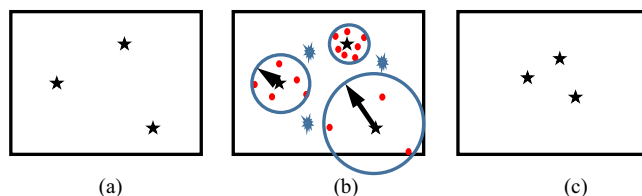


Fig. 1 花火アルゴリズムの探索過程. (a) 花火個体をランダムに生成, (b) 花火個体周辺にスパーク個体 (赤丸) と突然変異スパーク個体 (青丸) を生成, (c) 次世代の新しい花火個体になる個体を (b) から選択. 終了条件まで (b) と (c) を繰り返す.

Algorithm 1 花火アルゴリズム概略.

- 1: 花火個体 n 個のランダム初期化.
- 2: 各花火個体の fitness 評価.
- 3: while 終了条件が満たされるまで do
- 4: 各花火個体周辺にスパーク個体を生成.
- 5: Gauss 突然変異による突然変異スパーク個体生成.
- 6: if 探索範囲外にスパーク個体が生成されたら then
- 7: マッピング規則で探索範囲内に戻す.
- 8: end if
- 9: 各生成スパーク個体の fitness 評価.
- 10: 次世代用の n 個の新花火個体を選択.
- 11: end while
- 12: end of program.

花火アルゴリズムの性能向上を目指して色々な改良が行われてきており¹⁵⁾, 本論文の実験では, そのうちの一つである Enhanced 花火アルゴリズム (EFWA)¹⁸⁾ をベースアルゴリズムとして採用し, これに我々提案手法を組み込むことで提案手法の効果を評価する.

Scouting Strategy Applied to Fireworks Algorithm

[†] Jun YU (yujun@kyudai.jp)

^{††} Ying TAN (ytan@pku.edu.cn)

[‡] Hideyuki TAKAGI

(<http://www.design.kyushu-u.ac.jp/~takagi/>)

Graduate School of Design, Kyushu University (†)

Peking University/Kyushu University (††)

Faculty of Design, Kyushu University (‡)

3 提案戦略

3.1 偵察戦略

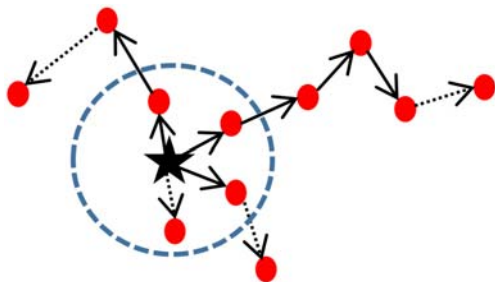


Fig. 2 fitness 景観上の偵察戦略の軌跡．黒の星印は偵察開始点，赤丸は逐次偵察点．実線は現在の探索方向が良さそうなので更に偵察を続け，破線はfitnessが悪化し偵察を打ち切ることを示す．これを花火アルゴリズムに組み込むと，星印が花火個体になり，赤丸がスパーク個体になる．

偵察戦略とは，局所領域での有望な探索方向を予め探った後に，その有望方向にバイアスがかかるように子個体を生成する方法である．本論文では花火アルゴリズムを例に偵察戦略の組み込みを説明・評価を行うが，この局所領域の偵察の考えは一般の進化計算にも組み込み可能である．

偵察戦略のアルゴリズムは，探索開始点の周辺に一つの子個体を生成することから始まる．その子個体が親個体よりも良ければその優良子個体の周辺に次の子個体を生成する．優良子個体が生成される限りこの処理を繰り返す．もし子個体とその直前の個体よりも悪ければその偵察探索は打ち止めにし，最初の探索開始点から同様の処理を繰り返す．すなわち，fitnessが向上する有望な探索方向であれば新規生成個体の周辺探索を継続し，そうでなければ局所探索方向が芳しくないとして打ち切る．その結果，Fig. 2のように，何回連続して探索を継続したか（深さ数）が有望な探索方向の指標になる．

偵察戦略を具体的に花火アルゴリズムに組み込む場合について説明する．花火個体が探索開始点である．この周辺に一つのスパーク個体をランダムに生成する．このスパーク個体が花火個体よりも良ければ，この優良スパーク個体周辺にランダムに次のスパーク個体を生成する．優良スパーク個体が生成され続ける限り，この逐次スパーク個体生成を繰り返し，悪ければ探索開始個体である花火個体まで戻って同様の処理を繰り返す（Fig. 2）．この探索を，花火アル

ゴリズムが決定する各花火個体周辺のスパーク個体数に達するまで繰り返す．Algorithm 2にスパーク個体生成の詳細を示す．

提案する偵察戦略を花火アルゴリズムに組み込むことでどのように変わるかを説明する．ベースになる花火アルゴリズムが同じであれば，適応的に決まる各花火個体の生成スパーク個体数と花火個体周辺の探索範囲の大きさは同じである．異なるのは炸裂してスパーク個体を生成する方法のみである．これまで花火アルゴリズムのバリエーションが色々提案されて来た¹⁵⁾が，これらは，探索範囲の大きさを適応的に決めるものの，その固定探索範囲内（Fig. 2の黒丸破線内）に全スパーク個体を一度にランダム生成するものであった．偵察戦略では固定探索範囲内でスパーク個体を生成するという制約がなく，上述の有望な探索方向であるかどうかの基準で逐次生成する優良スパーク個体の周辺に次のスパーク個体を生成する．したがって花火個体周辺の固定探索範囲を超えて，有望探索方向にどんどん偵察が進む（スパーク個体を生成しつづける）（Fig. 2）．この点が大きな違いである．

3.2 フィルタリング戦略

第2の改良は，個体群内の競争圧力を高める戦略である．従来のEFWAでは，生成されたスパーク個体が親個体（花火個体）よりも良いかどうかに関わらず，次世代に生き残る可能性は大なり小なりすべての個体にあった．そのため好ましくない領域への探索を繰り返す可能性もあり，探索資源の無駄使いになり得る．特に花火個体周辺のスパーク個体数が少ない場合は，このような無駄な領域への探索は避けたい．

提案するフィルタリング戦略は，花火個体のfitnessよりも悪いスパーク個体を除き，残された良いスパーク個体，突然変異個体，親である花火個体の中から次世代個体を選択するものである．この戦略は期待が持てる領域での探索を増やし，可能性の高い方向への進化を加速させる．Fig. 3にこのフィルタリング戦略例を示す．

4 評価実験

CEC2013ベンチマーク関数⁷⁾の28関数を評価関数として用い，公平な評価のために世代数ではなくfitness計算回数で探索性能を比較する．各関数とも，3種類の異なる次元数で各々30回試行をし，Wilcoxonの符号検定で，最大fitness計算回数

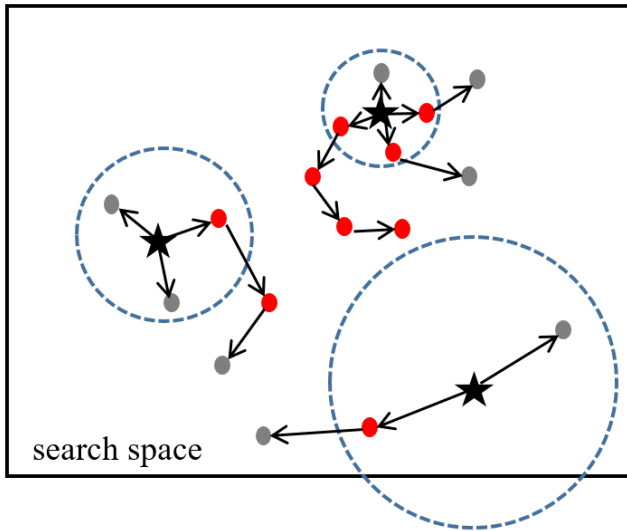


Fig. 3 fitnessの悪い個体のフィルタリングの様子．黒の星印は花火個体，丸個体はスパーク個体で，赤丸は生き残り，灰色丸は次世代選択候補から取り除かれる．矢印は探索軌跡方向を示す．

MAX_{NFC} 時点での従来のEFWAと（EFWA + 提案戦略）の平均fitness値の間の有意差検定を行う．

Table 2 に検定結果を示す．ほとんどの場合，提案戦略を組み込むことで有意に性能が向上していることが分かる．

Table 1 EFWAアルゴリズムのパラメータ設定．

2D, 10D, 30D 探索での花火個体数	5
全スパーク個体数 m	60
Gaussian 突然変異個体数	5
花火個体周辺の最大探索範囲 A_{max}	40
固定パラメータ値	$a = 0.04$ $b = 0.8$
2D, 10D, 30D 探索での最大fitness計算回数 MAX_{NFC}	1,000, 10,000, 40,000
ベンチマーク関数の次元数 D	2, 10, 30
試行回数	30

5 考察

提案戦略の優位性について述べる．従来の花火アルゴリズムでは，少数花火個体の周辺に多くのスパーク個体を生成するが，選択演算後にはこれら個体は捨てられる．今回の実験条件の場合，スパーク個体数は花火個体数の12倍であり，局所空間情報は花火個体からだけのもので，広がったスパーク個体からの多くの局所空間情報は十分に使われずじまいになってしまう．ス

Algorithm 2 偵察戦略組み込んだ新しい花火アルゴリズム．

- 1: 生成スパーク個体数 S_i の決定．
- 2: 探索範囲 A_i の決定．
- 3: i を 0 初期化．
- 4: **while** $i < S_i$ の間 **do**
- 5: 探索範囲 A_i 内にランダムな1個のスパーク個体を生成し fitness を評価．
- 6: $i = i + 1$.
- 7: **if** スパーク個体が花火個体よりも悪い **then**
- 8: step 4 に戻り，繰り返す．
- 9: **end if**
- 10: **while** 次に生成するスパーク個体が直前に生成したスパーク個体あるいは親の花火個体よりもよい間 **do**
- 11: 新しいスパーク個体を直前に生成したスパーク個体を中心とする探索範囲 A_i 内にランダムに生成する．
- 12: $i = i + 1$.
- 13: **end while**
- 14: **end while**
- 15: **end of program.**

パーク個体から得られる局所空間情報を有効に使うために，花火個体周辺の複数方向を逐次に偵察する戦略を提案した．

固定探索範囲内でランダムに探索するのではなく，より良い探索方向であればどんどん深く偵察を進める方式であるので，より良い方向への偵察が進む程，偵察回数が増えて局所探索性能が増す．さらに，提案手法は各花火個体周辺探索に必要な計算資源（スパーク個体数）は従来法と同じであり，計算コスト増はない．本論文では花火アルゴリズムに提案戦略を組み込むことで評価したが，偵察戦略は局所領域の有望な探索方向を知ることが目的であり，他の進化計算アルゴリズムに組み込むことも可能である．

最初の試みとして，本論文では単一点の偵察軌跡から局所空間情報を得ようとしたが，これに限定されるものではない．今後の拡張として，一層性能向上が期待できる色々な方法があろう．例えば，1スパーク個体から多数の偵察スパーク個体を生成する方法であれば，1点を生成する方法よりも局所最適解に捕まるリスクが少なくなるかもしれない．また，fitness 景観上の偵察探索の範囲を適応的に調整するインテリジェントルールを設計できるかもしれない．

2番目のフィルタリング戦略の主目的は，期待ができる方向への探索の可能性を高め，非効率的な探索をなくすことである．従来の花火アルゴ

Table 2 最大fitness評価回数時での30回試行平均fitness値の差に対するWilcoxon符号検定結果．
 \gg と $>$ は各々危険率1%と5%で有意差あり， \approx は有意差が見られなかったことを示す．提案手法は，偵察戦略とフィルタリング戦略をEFWAに組み込んだものである．

Func.	2-D	10-D	30-D
F_1	提案手法 \gg EFWA	EFWA $>$ 提案手法	EFWA \gg 提案手法
F_2	提案手法 \approx EFWA	提案手法 $>$ EFWA	提案手法 \gg EFWA
F_3	提案手法 \approx EFWA	提案手法 $>$ EFWA	提案手法 \gg EFWA
F_4	提案手法 $>$ EFWA	提案手法 $>$ EFWA	提案手法 $>$ EFWA
F_5	提案手法 \gg EFWA	提案手法 \gg EFWA	提案手法 \approx EFWA
F_6	提案手法 \approx EFWA	提案手法 $>$ EFWA	提案手法 \gg EFWA
F_7	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 \gg EFWA
F_8	EFWA \approx 提案手法	提案手法 \approx EFWA	提案手法 \approx EFWA
F_9	提案手法 $>$ EFWA	提案手法 \gg EFWA	提案手法 \approx EFWA
F_{10}	提案手法 \gg EFWA	提案手法 \gg EFWA	提案手法 \approx EFWA
F_{11}	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{12}	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{13}	提案手法 \approx EFWA	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{14}	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{15}	提案手法 \approx EFWA	EFWA \approx 提案手法	EFWA \approx 提案手法
F_{16}	提案手法 \approx EFWA	EFWA \gg 提案手法	EFWA \gg 提案手法
F_{17}	提案手法 $>$ EFWA	提案手法 $>$ EFWA	提案手法 $>$ EFWA
F_{18}	提案手法 $>$ EFWA	提案手法 $>$ EFWA	提案手法 $>$ EFWA
F_{19}	提案手法 \gg EFWA	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{20}	提案手法 \approx EFWA	提案手法 \gg EFWA	EFWA \gg 提案手法
F_{21}	EFWA \approx 提案手法	EFWA \approx 提案手法	提案手法 \approx EFWA
F_{22}	提案手法 \approx EFWA	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{23}	提案手法 \approx EFWA	提案手法 \approx EFWA	EFWA \approx 提案手法
F_{24}	提案手法 \approx EFWA	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{25}	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{26}	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 \gg EFWA
F_{27}	EFWA \approx 提案手法	提案手法 \gg EFWA	提案手法 $>$ EFWA
F_{28}	提案手法 \approx EFWA	提案手法 \gg EFWA	提案手法 \gg EFWA

リズムでは，生成スパーク個体が親の花火個体よりも悪くても，次世代に生き残る可能性がある．このようなスパーク個体が選択されると期待の持てない領域で無駄な探索を行うことになり，収束の低下を招く．2番目の戦略は不良個体を取り除くことで，全個体が最適解方向に効率的に進化するようにする．もちろん，親の花火個体よりも悪いスパーク個体をすべて取り除くのではなく，多様性維持のためにいくつかは残しておく．収束と多様性維持とのバランスをどのように取るかは今後の我々の検討課題である．

次に，提案戦略の適用性について述べる．本論文で用いたEFWAに限らず，dynFWA¹⁹⁾のような色々な花火アルゴリズムにも適用できる．さらに，提案の二つの戦略は各種進化計算に単独に，あるいは，二つ一緒に組み込むことが可能である．

性能解析のために，探索終了時での従来EFWAと(EFWA+提案戦略)の平均fitness値の差をWilcoxon符号検定で調べた．Table 2の検定結果

は，提案戦略を組み込むことで多くの場合有意に性能を向上が図ることができ収束速度も向上することを示した．さらにこの改善効果はタスクの次元数があがり複雑度が上がるほど顕著であることを示した．これはおそらく，複雑な高次元問題に対してスパーク個体数が不十分であり，固定範囲内の探索に代わって偵察探索が局所空間の情報を早く利用できることことができるからであろう．しかし，Sphere関数(F_1)とKatsuura関数(F_{16})では単独のEFWAに比べて性能が劣化した．Sphere関数は単純な単峰性関数なので，あちこちを偵察探索することなくすぐ収束する個体に焦点を当てればよいからであると容易に理解できる．しかしKatsuura関数に関しては，なぜこのような結果になったかを知り解決策を考えるには一層の解析が必要である．

6 結論

本論文では，局所空間での探索能力を向上させる二つの戦略(より良い方向に深く探索を進める偵察戦略と不良個体の取り除くフィルタリング戦略)を提案した．EFWAに組み込む評価実験を通して提案戦略は局所空間をより効率的に探索できることが明らかになった．

今後は，一層局所空間情報を探索に反映するための手法を開発していく．更に，fitness景観の特性を理解し，個体の進化をより効率的になるようガイドしていく方法を考えていきたい．

謝辞

本研究はJSPS科学研究費(課題番号 JP15K00340)と中国国家自然科学基金(課題番号 61673025)の助成を受けたものである．

参考文献

- 1) Adler, Dan, "Genetic algorithms and simulated annealing: A marriage proposal," IEEE Int. Conf. on Neural Networks (ICNN1992), San Francisco, CA, USA, pp.1104-1109 (March/April, 1993).
- 2) Das, S. and Suganthan, P.N., "Differential evolution: A survey of the state-of-the-art," IEEE Trans. on Evolutionary Computation, vol.15, no.1, pp.4-31 (2010).
- 3) Das, S., Mullick, S. S. and Suganthan, P.N., "Recent advances in differential evolution? An updated survey," Swarm and Evolutionary Computation, vol. 27, pp.1-30 (April 2016).
- 4) Eiben, Á.E., Hinterding, R., and Michalewicz, Z., "Parameter control in evolutionary algorithms,"

- IEEE Trans. on Evolutionary Computation, vol.3, no.2, pp.124–141 (1999).
- 5) Fox, Bennett L., “Integrating and accelerating tabu search, simulated annealing, and genetic algorithms,” *Annals of Operations Research*, vol.41, no.2, pp.47–67 (June 1993).
 - 6) Hansen, N. and Ostermeier, A., “Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation,” *IEEE Int. Conf. on Evolutionary Computation (ICEC1996)*, Nagoya, Japan, pp.312–317 (May, 1996).
 - 7) Liang, J., Qu, B., Suganthan P.N. and H., A., G., “Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization,” *Technical Report (2013)*. http://al-roomi.org/multimedia/CEC_Database/CEC2013/RealParameterOptimization/CEC2013_RealParameterOptimization_TechnicalReport.pdf
 - 8) 村田昇, 西井龍映, 高木英行, 裴岩「世代間移動ベクトル群の収束点推定法」2014進化計算シンポジウム, 廿日市市 (2014年12月20-21日).
 - 9) Murata, N., Nishii, R., Takagi, H. and Pei, Y., “Analytical estimation of the convergence point of populations,” *2015 IEEE Congress on Evolutionary Computation (CEC2015)*, Sendai, Japan, pp.2619–2624 (May, 2015).
 - 10) Qin, A.K. and Suganthan P.N., “Self-adaptive Differential Evolution Algorithm for Numerical Optimization,” *IEEE Congress on Evolutionary Computation (CEC2005)*, Edinburgh, UK, pp.1785–1791 (Sept., 2005).
 - 11) Renders, J.-M. and Bersini, H., “Hybridizing genetic algorithms with Hill-climbing methods for global optimization: Two possible ways,” *IEEE Congress on Evolutionary Computation (CEC1994)*, Orlando, FL, USA, pp. 312–317 (June, 1994).
 - 12) Sen, M.K., Gupta, A.D., Stoffa, P.L., Lake, L.W. and Pope, G.A., “Stochastic reservoir modeling using simulated annealing and genetic algorithm,” *SPE Annual Technical Conf. and Exhibition*, Washington, DC, USA, pp. 939–950 (Oct., 1992).
 - 13) Storn, R. and Price, K., “Minimizing the real functions of the ICEC '96 contest by differential evolution,” *IEEE Int. Conf. on Evolutionary Computation*, Nagoya, Japan, pp.842–844 (May, 1996).
 - 14) Tan, Y., and Zhu, Y., “Fireworks algorithm for optimization,” *Advances in Swarm Intelligence*, vol. 6145 of the series *Lecture Notes in Computer Science*, pp.355–364 (2010).
 - 15) Tan, Ying, *Fireworks algorithm: A novel swarm intelligence optimization method*, Springer Berlin Heidelberg (Oct., 2015).
 - 16) 高木英行, 印具毅雄, 大西圭「単峰性関数当てはめによるGA収束高速化」*知能と情報 (日本知能情報フアジイ学会誌)*, vol.15, no.2, pp.219–229 (2003).
 - 17) Yu, J. and Takagi, H., “Acceleration for fireworks algorithm based on amplitude reduction strategy and local optima-based selection strategy,” *Int. Conf. on Swarm Intelligence (ICSI2017)*, Fukuoka, Japan, pp.477–484 (July, 2017).
 - 18) Zheng, S., Janecek A., and Tan, Y., “Enhanced fireworks algorithm,” *IEEE Congress on Evolutionary Computation (CEC2013)*, Cancun, Mexico, pp.2069–2077 (June, 2013).
 - 19) Zheng, S., Janecek A., Li, J. and Tan, Y., “Dynamic search in fireworks algorithm,” *IEEE Congress on Evolutionary Computation (CEC2014)*, Beijing, China, pp.3222–3229 (July, 2014).