

進化計算の初期化手法の比較

HARVEY, Matthias
九州大学大学院芸術工学府

高木, 英行
九州大学大学院芸術工学研究院

<https://hdl.handle.net/2324/4482780>

出版情報 : pp.1-, 2021-03-04. The Japanese Society for Evolution Computation
バージョン :
権利関係 :



進化計算の初期化手法の比較

HARVEY, Matthias[†], 高木英行^{††}

九州大学大学院芸術工学府[†], 九州大学大学院芸術工学研究院^{††}

1 はじめに

近年、進化計算の実応用が進んでいる。実社会の問題では、設計変数が多い等の複雑度が高い問題も多く、フィットネス関数や制約条件の確認のための計算コストが高いという大きな問題を抱えている。このような例には、マツダの複数車両設計問題⁷⁾、流体シミュレーション^{3, 5)}、ニューラルネットワークのパラメータ最適化¹²⁾などがある。このための対策研究も取り組まれており³⁾、複雑な問題を多項式やニューラルネットワークで近似することで計算コストを軽減する方法が代表的な対策方式である。

計算資源が限られている場合は、大局的最適解近傍に辿り着くまで最適化探索を続けることは困難、あるいは、不可能である。計算資源が限られている場合は、大局的最適解探索を第一義に考えるよりも、限られた計算資源内でできるだけより良い解を探し出すこと、できるだけ早い探索世代での収束を加速させること、が最適化の戦略になる。この戦略に基づく代表的なアプローチは、特に早い探索世代での探索性能向上を目指す進化計算アルゴリズム改良である。第2は進化計算の改良ではなく、進化計算の初期個体群の生成、つまり、初期化に着目したアプローチであり、本論文はこの第2の取り組みを扱う。

通常、多くの個体群初期化には単純な擬似乱数初期化が用いられるが、色々工夫提案もある。高次元探索空間では、初期個体群の均一性が指数関数的に低下するため、初期個体でカバーできない領域が拡がり、初期個体群の質が劇的に低下することが知られている⁶⁾。初期個体群の均一性は念頭におくべきキーワードの一つである。最近の個体群初期化を扱う大多数の方法は、近似モデルやニューラルネットワークをメタモデ

ルとして個体生成に利用する方法であると言われている³⁾。筆者らの近似モデルによる初期個体生成の研究では、探索空間の次元数に対する十分な個体数がない場合、fitness景観の近似性能が悪いため近似モデルから得る有力領域が実際のfitness景観の良い領域からずれることが多く、この近似モデルに基づく初期個体生成は必ずしも早い世代での探索に役立つわけではないことが明らかになった¹⁰⁾。

本論文の目的は、初期個体群の生成方法、すなわち進化計算の初期化方法がその後の早い世代での探索にどのように影響を与えるかを実験的に明らかにする。本論文の取り組みは進化計算の改良ではなく、代表的な数値手法を用いた四つの初期化方法+我々の提案初期化法の合計5種類初期化法の比較実験を通じて、初期化の違いが進化計算の早い探索世代での収束にどう影響するかを実験的に明らかにする。

2 初期個体群決定方法

2.1 擬似乱数初期化

個体群を初期化する一般的な方法は、擬似乱数発生器で探索境界内に収まる一様乱数を生成することである。よく用いられる方法の一つに、 $2^{19937} - 1$ という長い乱数周期が証明されているMersenne twister¹¹⁾がある。次元の呪い²⁾のために、一様乱数で個体生成をしても探索空間の広さ（次元数）に応じた十分な数の個体を生成しない限り、探索空間では偏った個体群になってしまう。図1にMersenne twisterで生成した乱数例を示す。

2.2 等間隔初期化

等間隔初期化は、探索空間内に等間隔で配置する個体を決定することが目的になる。これは球体パッキング問題とも呼ばれ、ほとんどが未解決の数学的に難しい問題である。これまでのところ、球体パッキング問題は2次元、3次元、8次元、最近になって24次元空間を埋め尽くす超

Comparison of EC Initialization Methods

[†] Matthias Harvey (matthiasharvey@gmail.com)

^{††} Hideyuki Takagi (h.takagi.457@m.kyushu-u.ac.jp)

Graduate School of Design, Kyushu University (†)

Faculty of Design, Kyushu University (††)

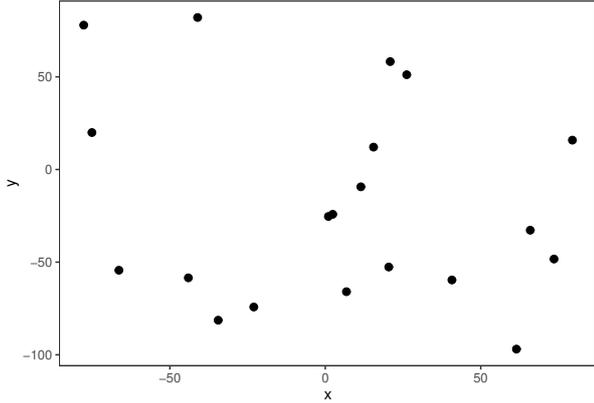


Fig. 1 Twenty points randomly generated on a 2-D space $((x, y) \in [-100, 100])$ using the Mersenne-Twister.

球配置解しかが得られていない⁴⁾。

このように数学的解法ではまだ多くの次元で等間隔配置の初期個体位置を決定できないため、本論文では近似法を用いる。我々の提案近似解法は、予め探索空間に多くの点を生成し、これらの生成点を進化計算探索の個体数分のクラスに分ける。得られた各クラスタの中心座標を初期化個体の位置とする方法である。

クラスタリング対象となる多くの初期点生成には、(a) 擬似乱数点生成と、(b) ある数値のリストのすべての可能な順列（繰り返し）生成する方法の2方法が考えられる。本論文では、以下に示す(b)の方法で多くの点を生成し、 k -means法でクラスタリングをする。こうして得られた2-D探索空間上の20点の例を図2に示す。

1. N をクラスタリング前の初期点の近似数、 d を次元数とする。
2. 初期点に用いる 1 次元の数を $s = \lceil N^{\frac{1}{d}} \rceil$ とする。
3. $\vec{v} = [x_{min}, \dots, x_{max}]$ とする。ここで、 \vec{v} の長さは s 、 x_{min} , x_{max} は探索範囲の下限と上限である。
4. 最後に、ベクトル \vec{v} の繰り返しのある順列をすべて求める。これらのベクトルをクラスタリング対象点とする。

クラスタリング対象点数 N を変えながら初期個体群を生成する予備実験を行い、次節での実験では $N = \text{個体数} \times 10,000$ とすることにした¹⁰⁾。

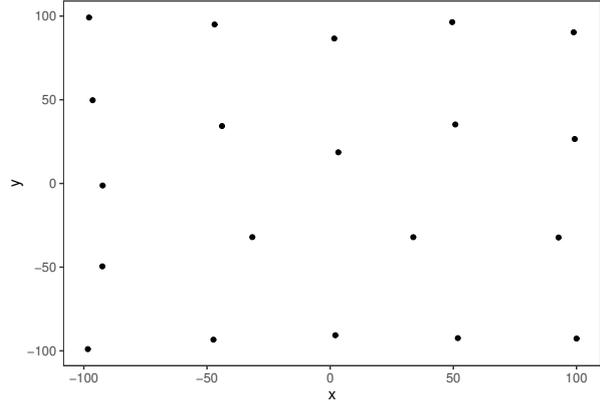


Fig. 2 Twenty equally distributed points using a clustering method.

2.3 Chebyshev 初期化

Chebyshev ノードは、第一種の Chebyshev 多項式の根から得られるある種の実数である。単位半円上の等間隔の点 n の x 座標値に相当し¹³⁾、式(1)で生成できる。

$$x_k = \cos\left(\frac{2k-1}{2n}\pi\right), k = 1, \dots, n \quad (1)$$

Chebyshev ノードは、Runge 現象の影響を最小限に抑えることができるので、多項式補間によく用いられる⁹⁾。 $n = 16$ の場合の単位半円から Chebyshev ノードを求める方法を図3に示す。

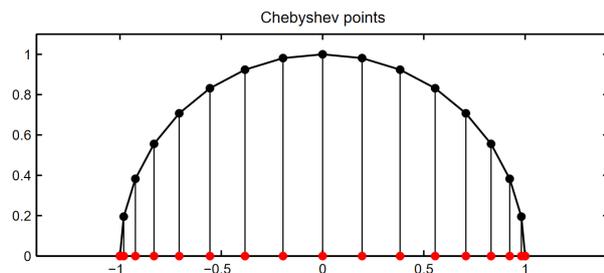


Fig. 3 The Chebyshev nodes can be seen as the x coordinates of n equally spaced points on a unit semicircle ($n = 16$) in this figure.

Chebyshev 初期化は、各次元の Chebyshev ノードを初期化点として用いる個体群初期化法である。

探索範囲が決まっている場合の Chebyshev ノードは常に同じ値になるため、Chebyshev ノード関係を維持しながら異なる初期化群を生成して比較実験するために、 $[0, \text{点間平均距離}]$ の範囲で乱数を生成し、その乱数分 Chebyshev ノードをシフトすることで多様な Chebyshev 初期化群を

評価実験用に用意する．図4に，Chebyshev ノード群をシフトしていない例を，図5に，シフトした例を示す．シフトすることで Chebyshev ノードが探索空間外に出てしまう場合は，その点を境界鏡面反射位置に戻す

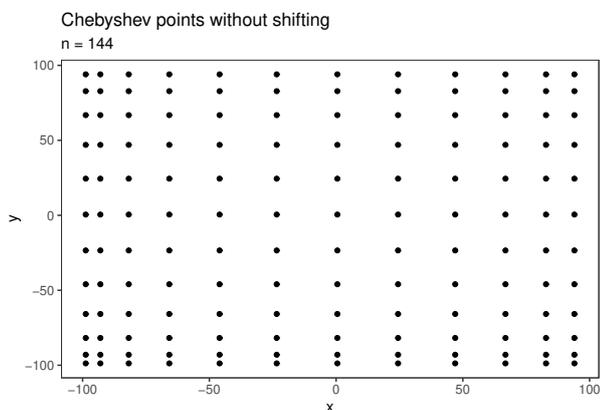


Fig. 4 Chebyshev points without shifting.

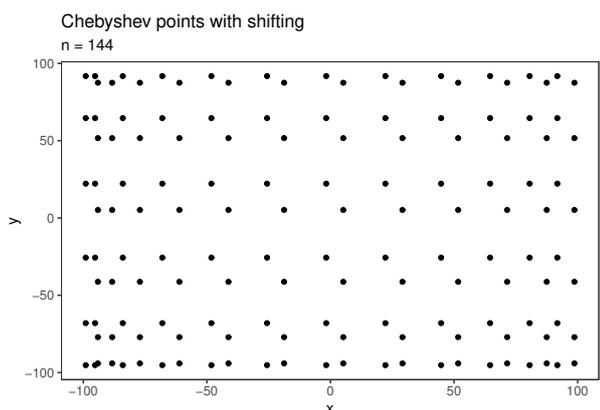


Fig. 5 Chebyshev points with shifting.

Chebyshev ノード数は任意に設定できないため，生成できるノード数に最も近い点数を使用し，残りの不足点数はランダムに生成する．図6は，個体数180点の場合，144点の Chebyshev ノードをランダムに追加した例を示す．

2.4 Lévy Flight 初期化

Lévy flight は random walk の一種で，異なるステップ幅の分布を持つ．Lévy flight の特徴は，他の random walk と異なり大きく跳躍することがある点である．この様子は，ある場所で探索した後，急に遠くの場所に行ってその新たな場所で探索する採集動物の行動に似ている．Lévy flight による random walk 例を図7に示す．

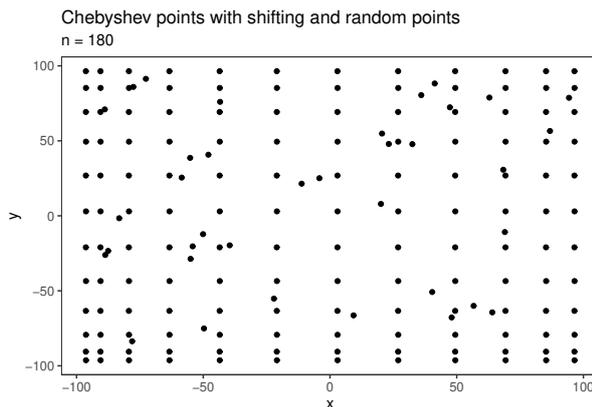


Fig. 6 Chebyshev points with shifting and random points.

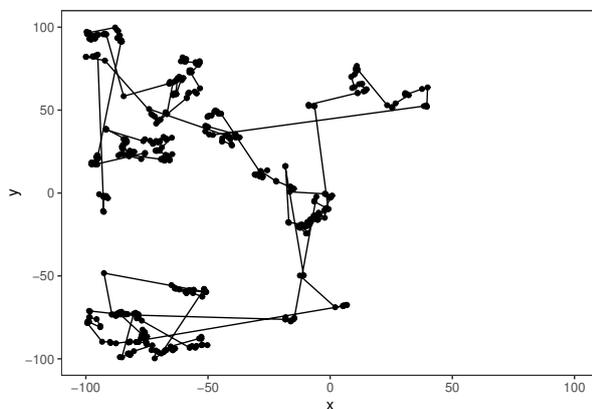


Fig. 7 An example of Lévy flight with 500 points and $\beta = 1$.

Lévy flight を用いた乱数生成過程は，ランダムな方向を選択する過程と，Lévy 分布に従ってステップ長を決める過程の二つから構成される．Lévy flight の実装には，しばしば Mantegna アルゴリズム¹⁴⁾ が用いられ，式(2)でステップ長 s が計算される．

$$s = \frac{u}{|v|^{1/\beta}} \quad (2)$$

ここで， u と v は標準正規乱数 ($u \sim N(0, \sigma_u^2)$, $v \sim N(0, \sigma_v^2)$) で，これらの標準偏差は式(3)で与えられる．

$$\sigma_u = \left[\frac{\Gamma(1 + \beta) \sin(\pi\beta/2)}{\Gamma[(1 + \beta)/2] \beta 2^{(\beta-1)/2}} \right]^{1/\beta}, \quad \sigma_v = 1 \quad (3)$$

ランダムな方向の選択は慎重に行う必要がある．素朴な方法は， $[0, 1]$ の一様ランダムベクトルを生成しそれを正規化することである．しか

し、この方法では正しい一様ランダム方向が得られない。生成された点は球上に一様分布していなければならない。これは任意の次元で n 個の正規乱数変数 x_1, x_2, \dots, x_n を生成し、それらを正規化することで実現できる⁸⁾。

β を色々変化させながら初期個体群を生成する予備実験を行い、次節の実験では、 $\beta = 1.11$ を用いることにした¹⁰⁾。

2.5 加重平均勾配初期化法

これまでの初期個体生成法は、乱数生成を含めた数的手法に基づいて個体数分の初期化座標を決める方法であり、生成された個体のfitness値、すなわち、fitness景観情報を使わない。言ってみれば、目を瞑って個体群を生成し、その後やおらfitness値を計算し始めて、ようやく進化計算アルゴリズムが動き始めるようなもので、生成個体情報を積極的に使っていないもったいない初期化法と言える。

加重平均勾配初期化法は、偵察戦略¹⁵⁾にヒントを得た提案初期化法である。偵察戦略は1点ずつ個体を生成してfitness値を計算し、その直前に生成した個体のfitness値よりも良ければ同じ方向へ進み、悪ければ戻って別方向に進む考え方で、初期化法に応用した場合の特徴は、1点ずつ初期個体を生成してfitness計算をする点と、それまでに生成した各初期個体のfitness情報とそれら個体への距離情報から次の初期個体を生成する点にある。fitness情報があればfitness景観の勾配情報が使えることになり、過去の生成個体への距離に反比例するように勾配情報を重み付き平均して山登り的に次の初期個体を生成する。すると、探索空間を均一に埋めるように初期化する従来法に対し、積極的にfitnessの高い領域に多くの初期個体を生成するようになる。一時的に直登の山登りをしなくても、それまでの勾配情報が増えるにつれて山登りを補正するようにも働く(図8)。

最初の1点から個体数分の全初期個体生成にこの方法を適用すると、最初の1点近傍の局所最適解に向かう可能性が高くなるので、最初に探索空間に偏らないような数点を置き、各々からこの加重平均勾配初期化法を使って1点ずつ逐次に初期個体を増やしていく。この点は、花火アルゴリズムの広域探索を担当する花火個体群と、各花火個体の周辺の局所探索を担当するスパーク個体群の関係に似ている。

以下に本提案初期化法を詳細に説明する。概

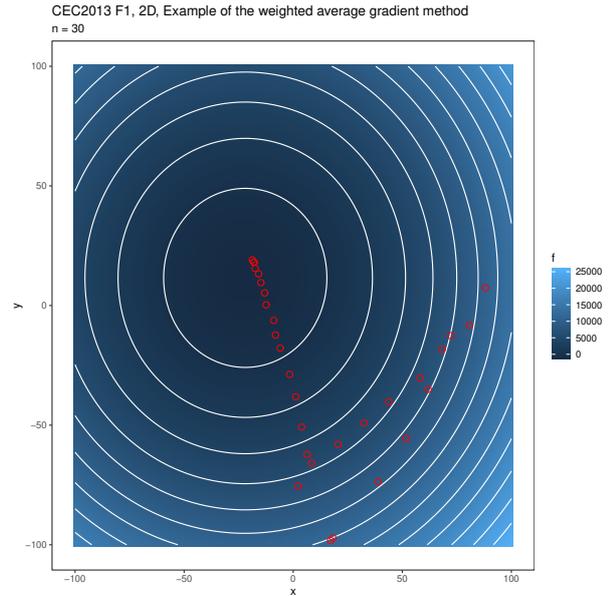


Fig. 8 An example of the weighted average gradient method on the CEC2013 benchmark function F_1 . Optimisation starts around the x-axis and then first moves to the right and up. It then moves back down left and then finally converges to the optimum. According to the # of past points, the direction is corrected to the right direction.

要は、初期化点を \mathbf{x}_1 から現在の点 \mathbf{x}_n までを生成したとし、次の \mathbf{x}_{n+1} をどのように生成するかを考える。直近の生成個体 α 個を重み付け対象にし、古く生成した初期個体になる程 β ずつ重要視しないように重み付けをし、さらに、 \mathbf{x}_n に遠い個体程重要視しないよう距離に反比例するよう重み付けをする。

以下数式で説明する。点 \mathbf{x}_i から現在の点 \mathbf{x}_n への勾配は式(4)で、2点間の単位方向ベクトル \mathbf{u}_i は式(5)で表される。

$$\text{gradient}_i = \frac{f(\mathbf{x}_n) - f(\mathbf{x}_i)}{\|\mathbf{x}_n - \mathbf{x}_i\|} \quad (4)$$

$$\mathbf{u}_i = \frac{\mathbf{x}_n - \mathbf{x}_i}{\|\mathbf{x}_n - \mathbf{x}_i\|} \quad (5)$$

したがって、勾配情報付の単位ベクトル \mathbf{g}'_i は両者の積として式(6)で与えられる。

$$\mathbf{g}'_i = \frac{(\mathbf{x}_n - \mathbf{x}_i)(f(\mathbf{x}_n) - f(\mathbf{x}_i))}{\|\mathbf{x}_n - \mathbf{x}_i\|^2} \quad (6)$$

直近の生成個体 α 点から \mathbf{x}_n に向かう勾配情報付の単位ベクトル群(式(7))から次に生成する

\mathbf{x}_{n+1} を決定する.

$$\mathbf{G} = [\mathbf{g}'_{n-\alpha}, \dots, \mathbf{g}'_{n-1}] \quad (7)$$

新しく生成した初期化点からの勾配情報ほど重要視する (式(8)) の重み付け

$$\mathbf{w} = (1, 2, \dots, \alpha)^T \quad (8)$$

をすると, 時間情報と勾配情報と距離で重み付けされた単位ベクトル群は式(9)で与えられる.

$$\mathbf{G}' = \mathbf{G} \frac{\mathbf{w}}{\sum \mathbf{w}} \quad (9)$$

古い生成個体になる程勾配スケールパラメータ β で重要視を少なくすることをかんがると, \mathbf{x}_{n+1} を決定するほぼ最終的な加重平均勾配情報 $\mathbf{v}_{weighted}$ が決まる.

$$\mathbf{v}_{weighted} = \left[\beta \cdot \sum_{k=1}^{n-j} \mathbf{G}'_k \right] \div (n-j) \quad (10)$$

$\mathbf{v}_{weighted}$ が大きく跳ね上がりすぎないように, またステップサイズを徐々に小さくするために, このステップサイズを \mathbf{v}_{max} で制限する.

$$\mathbf{v}_{max} = \frac{25 \cdot (iteration_{max} - iteration_{curr})}{iteration_{max}} \quad (11)$$

ここで, $iteration_{max}$ は最終的な初期化個体の数, $iteration_{curr}$ は現在までに生成した初期化個体数である. こうして, 最終的な加重平均勾配情報 \mathbf{v} が得られる.

$$\mathbf{v} = \begin{cases} \mathbf{v}_{weighted} \cdot \frac{\mathbf{v}_{max}}{\|\mathbf{v}_{weighted}\|}, & \text{if } \|\mathbf{v}_{weighted}\| > \mathbf{v}_{max} \\ \mathbf{v}_{weighted} & \text{otherwise} \end{cases} \quad (12)$$

次の初期化点 \mathbf{x}_n は $\mathbf{x}_n + \mathbf{v}$ に乱数ベクトルを加えて決定する. $\mathbf{r}_{unscaled}$ を探索範囲 [上界, 下界] の1/10の値の乱数とし, 式(13)のように生成する初期個体数が増えるにつれて小さくなるように乱数ベクトルをスケールリングしていく.

$$\mathbf{r} = \mathbf{r}_{unscaled} \cdot \text{rand} \left[0, \frac{iteration_{max} - iteration_{curr}}{iteration_{max}} \right] \quad (13)$$

こうして, 次の探索点 \mathbf{x}_{n+1} を計算することができる.

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{v} + \mathbf{r} \quad (14)$$

なお, \mathbf{x}_2 までは勾配情報が計算できないため, \mathbf{r} と同様に乱数 \mathbf{v} を使用する.

なお, 探索点 \mathbf{x}_{n+1} が探索範囲を越える場合は, 探索点を探索空間境界で探索空間内に折り返す. 初期個体の fitness 改善できていない場合は対向学習 (opposition-based learning: OBL) 点を生成する. 新たに生成された点 \mathbf{x}_{n+1} の fitness 値が前の点 \mathbf{x}_n よりも良くならなければ, 式(15)で $\mathbf{p} \in \mathbb{R}^n$ の OBL 点 $\tilde{\mathbf{p}}$ を追加するなどの工夫も可能である.

$$\tilde{\mathbf{p}} = \text{探索範囲の下界} + \text{探索範囲の上界} - \mathbf{p} \quad (15)$$

3 比較評価実験と考察

5次元の CEC2013 ベンチマーク関数群 ($f_1 \sim f_9$) を用い, 各初期個体群生成法で初期化した後に差分進化で30世代探索し, 早い世代での収束特性の比較を行う. 進化計算パラメータはアルゴリズムの探索性能に大きな影響を与える¹⁾が, 本論文で用いる差分進化は, R DEoptim ライブラリのデフォルトパラメータを用い, DE/local-to-best/1/bin, 交差率0.5, scale factor 0.8, 個体数50と100, 試行回数30回で, 比較実験を行う.

5-Dと10-Dのbenchmark関数9種類での5種類の初期個体生成法の違いによる収束曲線をそれぞれ図9と10に示す.

初期個体群の fitness 値群, 差分進化で最適化を行った第10, 20, 30世代での5種類の初期個体生成法別のそれぞれの30試行平均 fitness 値間に有意差があるかどうかを Kruskal-Wallis 検定を行う. 有意差があれば, Mann-Whitny の U 検定で得られた p 値を Holm の方法で補正する多重検定を行う. この補正 p 値から, 提案手法の加重平均勾配初期化法が第1位の平均 fitness 値を示した場合は第2の初期化法との間の有意検定をし, 提案手法が第1位でない場合は, 第1位の初期化法との間の有意検定を行う. この結果, 両者間に危険率1%と5%で有意差があれば, 図中の第1世代, および, 第10, 20, 30世代に, それぞれ**, および, * を書き込む.

生成した初期個体群 (第1世代) の検定結果を見ると, 5-D空間に対して50個体を生成する場合は, 提案手法は f_5 のみ平均 fitness は他よりも良く, f_9 では逆に悪くなっているが, 100個体に増やすと, 9関数中6関数で平均 fitness が良くなっている. f_9 では同様に悪くなっている. 一般に, 提案手法のみ fitness を鑑みながら逐次初期個体を生成するため, 確率的には fitness の良い領域に多くの初期個体が生成される. その効果は, 個体数の大きさと,

探索空間の大きさ+局所最適解に向かう問題の複雑さの兼ね合いであり、問題に対して十分な個体数があるのであれば、有意な効果が見込めるはずである。実験結果、特に個体数の違いによる効果の違いは、この理論的裏付けと言える。

次に、早い世代での収束に効果があった場合は、50個体では f_5 、100個体では f_1, f_3, f_5, f_6 であり、他の初期化手法に比べて有意に収束が劣ること場合はない。基本的に、有望な領域に多くの初期値を配置できれば、早い収束が期待できるはずである。fitnessを考慮することで有望な領域に多くの初期値を配置できるかどうかは、上述の考察と同様、個体数と問題の難易度に依存するため、早い世代で収束が加速するかどうかその点に依存する。

理論的考察と実験結果から結論をまとめれば、fitnessを考慮して初期個体を逐次に生成する手法は、fitness計算回数を増やすことなく、他の進化計算初期化法と同等以上に、早い世代での収束高速化が期待できる。

なお蛇足ながら、 f_5 と f_8 の検定結果を見ると、いかに視覚的に平均値の優劣を議論することが危険かを示している。

4 まとめ

本論文では、初期世代の収束性を向上させるために、特にfitness評価に計算コストがかかり、評価回数の上限が限られている最適化問題についての効率を向上させることを目的に研究を行った。個体群初期化方法を工夫することで、早い世代で進化計算（本論文での実験では差分進化）の探索効率を向上させることができることを示した。特に、初期個体群を決定してからfitness計算をして進化計算を始めるよりも、初期個体を配置する際にも決定した初期個体のfitness値を考慮しながら逐次に初期個体を決定する手法の有用性を示した。

今後の研究方向として、加重平均勾配初期化法に動的パラメータの導入や高計算コストの最適化実問題への適用がある。

謝辞

本研究はJSPS科学研究費（課題番号 JP18K11470）の助成を受けたものである。

参考文献

- 1) Beiranvand, V., Hare, W., and Luce, Y., “Best Practices for Comparing Optimization Algorithms,” *Optimization and Engineering*, vol.18, pp. 815-848 (Dec., 2017).
- 2) Bellman, Richard E., *Dynamic Programming*, Princeton University Press, Princeton, NJ, USA (1957).
- 3) Chugh, T., Sindhya, K., Hakanen, J. and Mietinen, k., “A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms,” *Soft Computing*, vol.23, pp.3137-3166 (2019).
- 4) Cohn, H., Kumar, A., Miller, S. D., Radchenko, D., and Viazovska, M., “The sphere packing problem in dimension 24,” *Annals of Mathematics*, vol.185, no.3, pp.1017-1033 (2017).
- 5) Kanazaki, M., Tanakam K., Jeong, S. and Yamamoto, K., “Multi-objective aerodynamic optimization of elements’ setting for high-lift airfoil using kriging model,” 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA, pp.1-11 (Jan., 2006).
- 6) Kazimipour, B., Li, X. and Qin, A. K., “Why Advanced Population Initialization Techniques Perform Poorly in High Dimension?,” *J. of Advanced Computational Intelligence and Intelligent Informatics*, vol.4, no.2, pp.479-490 (2014).
- 7) 小平剛央, 釘持寛正, 大山聖, 立川智章「応答曲面法を用いた複数車種の同時最適化ベンチマーク問題の提案」*進化計算学会論文誌*, v1.8, no.1 pp,11-21 (2017).
- 8) George Marsaglia, “Choosing a Point from the Surface of a Sphere,” *Annals of Mathematics and Statistics*, vol.43, no.2, pp.645-646 (Apr. 1972).
- 9) Mathews, J. H. and Fink, K. D., *Numerical methods using MATLAB*, 4th Edition, Pearson, NJ, USA, (Dec.. 2003).
- 10) Matthias Harvey, 「計算コストの高い最適化問題に関する研究」2020年度九州大学大学院芸術工学府芸術工学専攻修士論文。
- 11) Matsumoto, M. and Nishimura, T., “Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator,” *ACM Trans. Modeling and Computer Simulation*, vol.8, no.1, pp.3-30 (1998).
- 12) Springenberg, J. T., Klein, A., Falkner, S. and Hutter, F., “Bayesian optimization with robust Bayesian neural networks,” 30th Int. Conf. on Neural Information Processing (NIPS2016), pp.4141-4149 (Dec., 2016).
- 13) Trefethen, Lloyd N., “*Approximation Theory and Approximation Practice (Applied Mathematics)*, Society for Industrial and Applied Mathematics (Dec., 2012).
- 14) Yaman, A., Iacca, G., and Caraffini, F., “A comparison of three differential evolution strategies in terms of early convergence with different population sizes,” *AIP Conf.* 2070 (2019).

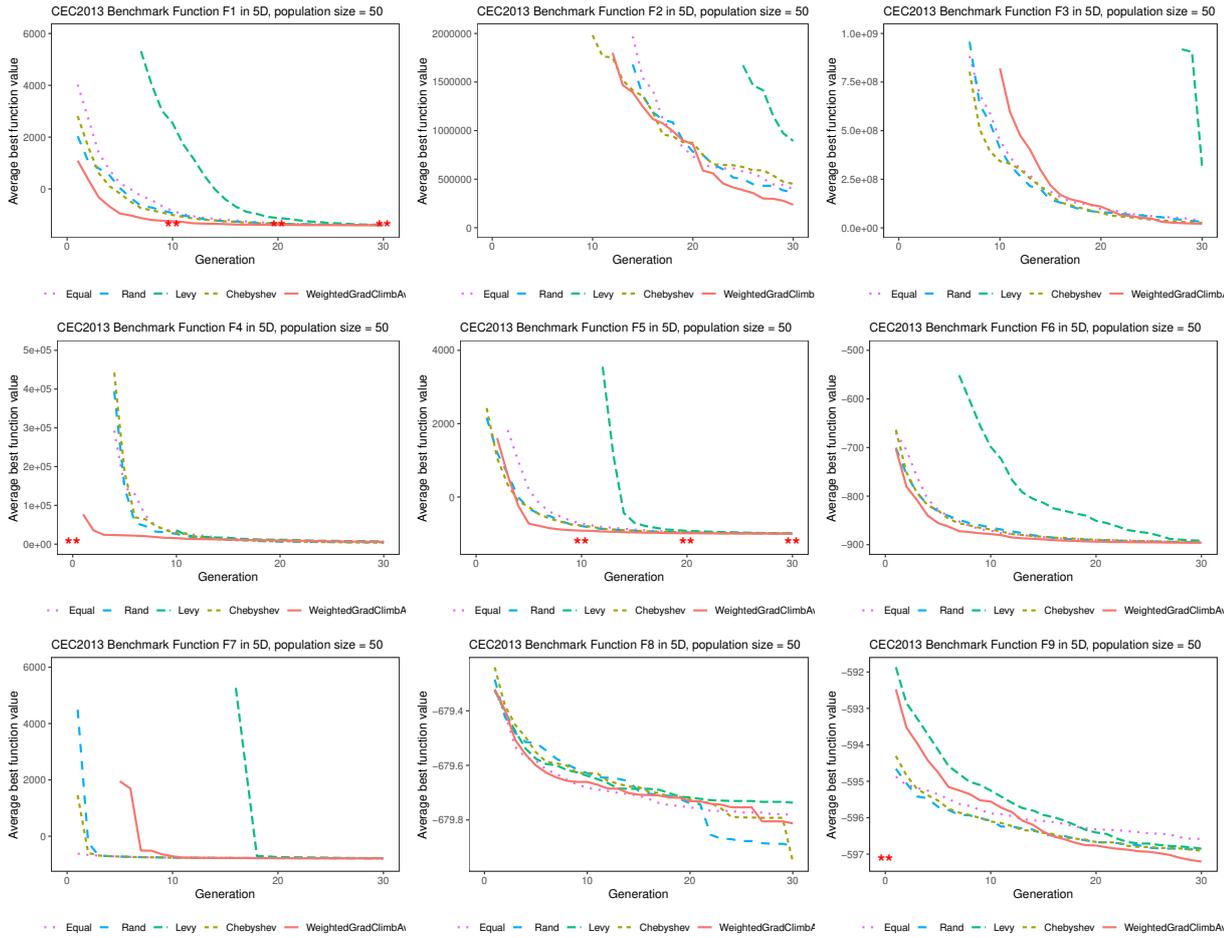


Fig. 9 Average convergence curves of the best fitness in 30 trial runs with 50 population size for CEC2013 benchmark functions $f_1 - f_9$ in 5-D. Graph data exceeding the max. scale value of the Y-axis are not drawn.

- 15) Yu, J., Tan, Y., and Takagi, H., "Scouting Strategy for Biasing Fireworks Algorithm Search to Promising Directions," Genetic and Evolutionary Computation Conf. Companion (GECCO2018), Kyoto, Japan, pp.99-100 (July 15 - 19, 2018).

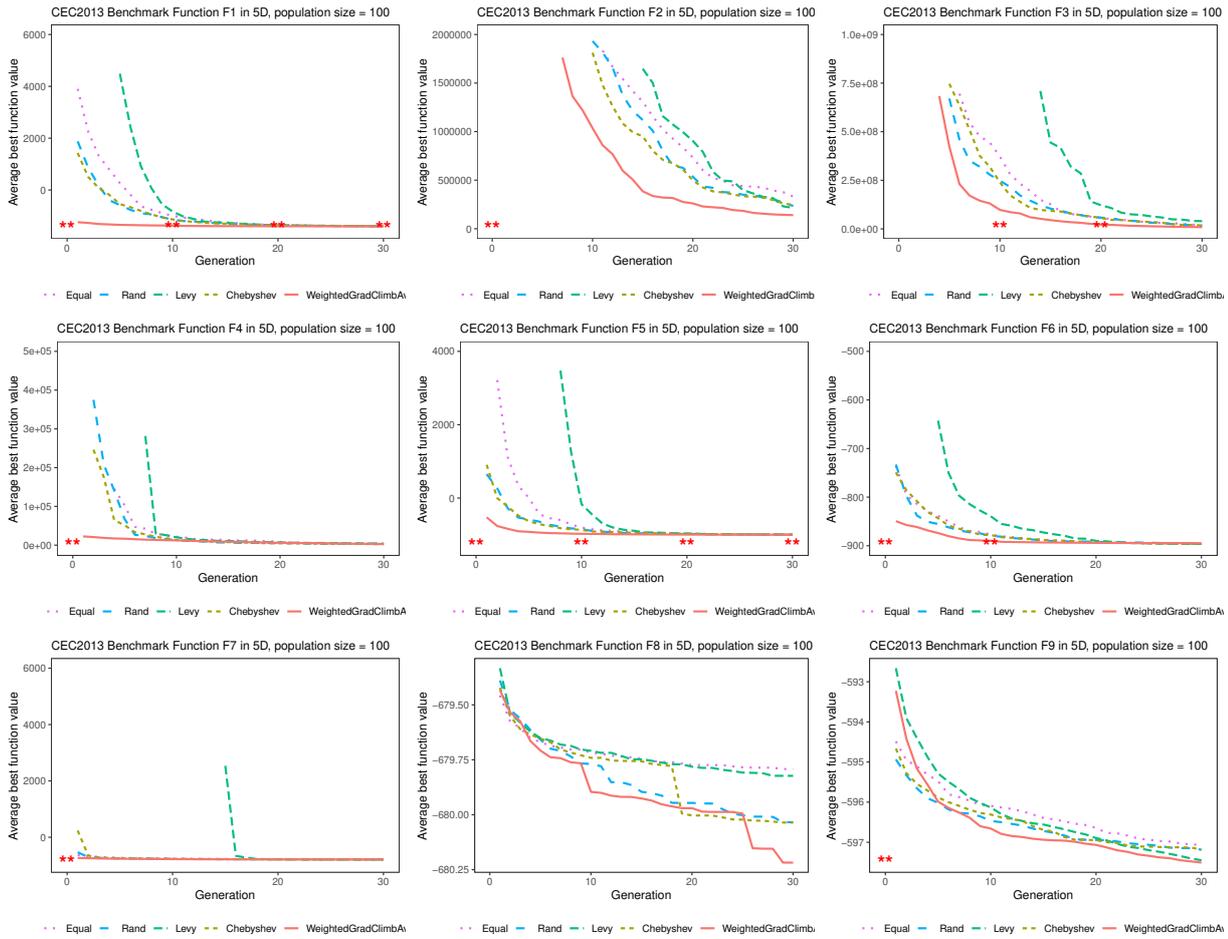


Fig. 10 Average convergence curves of the best fitness in 30 trial runs with 100 population size for CEC2013 benchmark functions $f_1 - f_9$ in 5-D. Graph data exceeding the max. scale value of the Y-axis are not drawn.