

# Nearest-better Clustering 法の補完ニッチ手法

李宇豪<sup>†</sup>, 余俊<sup>††</sup>, 高木英行<sup>‡</sup>

九州大学大学院芸術工学府<sup>†</sup>, JSPS 特別研究員 PD (九州大学)<sup>†</sup>, 九州大学大学院芸術工学研究院<sup>††</sup>

## 1 はじめに

代表的な fitness sharing [1] をはじめとして、これまで多峰の複数最適解を探索する niche 手法が提案されてきた [2]. niche 手法の一つに、局所最適解領域の切り出しアルゴリズムをまず走らせ、次に切り出された局所最適解領域のみに進化計算を適用して局所最適解を求める2段階方式の niche 法がある. この場合、前者の局所最適化領域を切り出す (すなわち、探索点を局所最適化領域毎にクラスタリングする) アルゴリズムの性能が重要になる.

Nearest-better clustering (NBC) [4], および、その改良版の NBC2 [5] は局所最適解領域切り出しアルゴリズムの一つで、NBC に進化計算 CMA-ES [4] を組み合わせた手法は CEC2013 niche コンペで優勝する程の性能を示した. NBC と NBC2 はまず探索点を最近傍の fitness 値が良い探索点に繋ぐことで接続木 (spanning tree) を形成する. 次に、接続木構造の中の長い辺を切断して複数の部分接続木に分割する. 部分接続木のノード群が局所最適解領域毎にグループ化された探索点群になる (つまり、局所最適解領域を切り出す).

このように強力な NBC, NBC2 であるが、高次元での niche 問題では全探索空間の局所最適解領域を分割することが困難になるという問題があった [3].

本論文の第1の目的は、高次元問題でも niche 領域を切り出すことができる手法を提案することである. 第2の目的は、評価実験を通して提案手法の適用性能を解析することである.

## 2 提案法

### 2.1 基本的な考え方

はじめに、提案手法の説明に使う用語と記号を Table 1 に定義する.

提案する局所最適解領域切り出し法である「重み付勾配-距離ベースのクラスタリング法

Table 1: 本論文でも用いる用語と記号の定義

記号	定義
局所最良探索点	niche 領域内で一番 fitness 値が良い探索点. 局所最適解ではない.
$CV_{i,j}$	$i$ 番目探索点と $j$ 番目探索点を結ぶ辺の山登り値
$d_{i,j}$ と $g_{i,j}$	$i$ 番目探索点と $j$ 番目探索点の間の距離と勾配
$\hat{d}_{i,j}$ と $\hat{g}_{i,j}$	正規化した $d_{i,j}$ と $g_{i,j}$ .
$w_g$ と $w_d$	勾配情報と距離情報への重み
$\alpha$	重み決定法1 (WGrAD <sub>1</sub> ) で用いるステップサイズ

(WGrAD)」は、まず  $N$  探索点を初期化し、次に以下の2ルールに基づいて長い辺にならないように各探索点を接続しながら接続木 (クラスタ) を生成する.

**ルール1:** 一つの探索点から fitness 値が良い探索点に出会うまで局所最適解領域内を山登りし (fitness の良い方を探し), その探索点と繋ぐ.

**ルール2:** 局所最良探索点から近傍探索点が見つかるまで下り, 両探索点を接続する.

WGrAD 法は第  $i$  番目探索点から残りの  $N - 1$  個の探索点の間で最大山登り関数値を持つ探索点を繋ぐ手法である. この提案法では、式(1)に示す勾配情報と距離情報の重み付き線形和で山登り関数値  $CV_{i,j}$  を表わす.

$$CV_{i,j} = w_g \times \hat{g}_{i,j} + w_d \times \hat{d}_{i,j} \quad (1)$$

$$\hat{g}_{i,j} = \frac{g_{i,j} - \min(g_{i,j})}{\max(g_{i,j}) - \min(g_{i,j})} \quad (2)$$

$$\hat{d}_{i,j} = \frac{\max(d_{i,j}) - d_{i,j}}{\max(d_{i,j}) - \min(d_{i,j})} \quad (3)$$

$$g_{i,j} = (f(\vec{x}_j) - f(\vec{x}_i)) / d_{i,j}, \quad d_{i,j} = \|\vec{x}_i - \vec{x}_j\| \quad (4)$$

ここで  $\max(d_{i,j})$ ,  $\min(d_{i,j})$ ,  $\max(g_{i,j})$ , and  $\min(g_{i,j})$  ( $1 \leq i, j \leq N, i \neq j$ ) は、各々、 $N - 1$  個の探索点

の中での最大距離, 最少距離, 最大勾配, 最少勾配を示す. その他の記号はTable 1を参照されたい. 勾配と距離は式(4)で求められる.

探索点群のクラスタリング結果は $w_g$ と $w_d$ に影響される.  $w_g$ が $w_d$ より大きい場合, 各探索点は「より良い」探索点と接続し少数の接続木が生成されるようになるが, その逆に $w_g$ が $w_d$ より小さい場合, 各探索点は「よい近い」点と接続するようになり, 狭い領域をカバーする多くの接続木が生成されるようになる.

本論文では, WGrADに組み込む二つの重み決定方式(WGrAD<sub>1</sub>とWGrAD<sub>2</sub>)を以下の節で提案する.

## 2.2 重み決定方式 1: WGrAD<sub>1</sub>

式(1)の山登り関数は, パラメータ $w_d$ のみの式(5)に書き直すことができる.  $w_d$ が大きくなるにつれて距離要素が重視され, 多くの接続木が生成される.

$$CV_{i,j} = (1-w_d) \times \hat{g}_{i,j} + w_d \times \hat{d}_{i,j}, \quad w_d \in [0, 1] \quad (5)$$

fitness景観に対する経験的な情報を使うことなく最適 $w_d$ 値を決定するために,  $w_d$ を徐々に増加させる動的決定法を提案する. この方法は, 同じniche領域の小さな接続木を統合してniche検出の精度を上げるために, 辺を切断しHill-Valleyテスト [8]をこれらの辺に適用する. 具体的には,  $w_d$ を0から1までステップサイズ $\alpha$ で変化させる.  $w_d$ 値を増やすということは, 相対的に勾配要素よりも距離要素をより重視することになるので, 探索点によっては, 遠くの良い探索点よりも近くのfitness値の悪い探索点と繋ぐことになる. この結果, 接続木のいくつかの辺は切断され, 一つの接続木が複数の部分接続木に分割されることになる.

重み $w_d$ と $w_g$ の違いによってどのように探索点の繋ぎ状況が変わるかをFig. 1に示す.  $w_d$ が小さく $w_g (= 1 - w_d)$ が大きい時には, 次のステップ $w_d (= w_d + \alpha)$ 時に探索点Bが「より良い」探索点Aと繋がれ, 一つの大きな接続木が作られる. しかし,  $w_d$ が大きく $w_g$ が小さい時には, 次のステップ時に探索点Bが「より近い」探索点Cと繋がれ二つの接続木ができる. 探索点Aと探索点Bは同じクラスタではなくなるので, 切断すべき辺A-Bが決まる.

切断辺が決まると, Hill-Valleyテストを各切断辺に適用して, 辺を枝刈りするか, 必要に応じ

て新しい辺を追加する. 両端点が同じniche領域にある場合は, 二つの両端点を持つ二つの部分接続木を一つに統合し, そうでない場合はテストした探索点を無視して新しいnicheの可能性ある領域を探す.

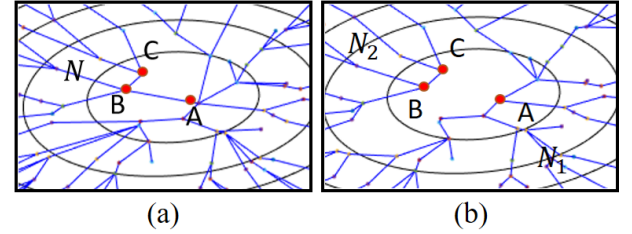


Fig. 1: 探索点接続への重み $w_d$ と $w_g$ の影響. (a)  $w_d$ が小さく $w_g (= 1 - w_d)$ が大きい場合, 次のステップ $w_d (= w_d + \alpha)$ 時に各探索点は「より良い」探索に繋がれ, 一つの大きな接続木が形成される. (b)  $w_d$ が大きく $w_g$ が小さい場合, 各探索点は「より近い」探索点に繋がれ, 二つの接続木が形成される.

## 2.3 重み決定方式 2: WGrAD<sub>2</sub>

距離に応じた異なる重み付けを探索点毎に行う第2の重み決定方式を提案する. この異なる重み付けのために, 式(1)を式(6)に書き直す.

$$CV_{i,j} = \hat{d}_{i,j} \times \hat{g}_{i,j} + (1 - \hat{d}_{i,j}) \times \hat{d}_{i,j} \quad (6)$$

局所最適解領域のより良い探索点を繋ぐために, 各々, 近傍探索点には, 距離要素には小さく勾配要素には大きい重み付けをする. 反対に, fitnessは良いが遠い探索点とは繋がないように, 遠くの探索点には, 距離要素に大きく勾配要素に小さい重み付けをする.  $\hat{d}_{i,j}$ の決定は式(3)を参照されたい.

## 3 評価実験

CEC2015 nicheベンチマーク関数群から, 8関数 $\times$ 6種類の次元 ( $D = 2, 4, 6, 8, 10, 20$ )  $\times$  30試行で二つの提案手法と従来のNBCとを比較評価する. これらのベンチマーク関数は局所解が離れているカテゴリC1の関数群と, 局所解が近いところにあるカテゴリC2関数群に分類できる.

第2.1節で述べたパラメータ $\alpha$ は0.1とする. 従来のNBC法も提案手法も局所最適領域の切り出しを行うアルゴリズムであるので, 第1段階でNBC, WGrAD<sub>1</sub>, WGrAD<sub>2</sub>が各々初期 $N$ 探索点をクラスタリングして局所最適領域を切り出し,

第2段階では共通進化計算が各アルゴリズムで切り出した局所最適領域の局所最適解を探索する。本実験では、この進化計算に差分進化を用い、探索できた局所最適解で比較評価する。差分進化のパラメータには、差分進化の原論文 [7] で用いられている記号を使うと、個体数  $NP = 80$ 、スケールファクタ  $F = 0.5$ 、交差率  $Cr = 0.5$ 、クラスタリングのための探索点数  $N = 200D$ 、差分進化演算  $DE/best/1/bin$  である。

各局所最適領域での局所最適解探索のための差分進化終了条件は (1) fitness 計算回数が上限に達した時、(2) 個体間距離が  $1.0e^{-8}$  以下になった時、(3) 最終  $T (= 50)$  世代での fitness 改善値が  $1.0e^{-8}$  以下になった時、である。探索全体の差分進化終了条件は、条件 (1) を満たすか、主要探索過程が終了した時である。

評価は、検出した大局的最適解数の30試行平均、および、切り出した局所最適領域数の30試行平均に対して、3手法間の有意差検定を行う。本提案手法は局所最適領域の切り出し法であり、切り出された領域数が大局的最適化数+局所最適化数と見なす。検定方法は、Wilcoxonの符号検定で得られた  $p$  値をHolmの多重検定で補正し3手法間のどこに有意差があるかを検定する。Table 2にこの検定結果を示す。

#### 4 考察

まず、局所最適領域切り出しアルゴリズムの特性について考察する。NBCは距離情報とfitness情報を使い、低次元関数（ほとんどの2次元関数）で効果的に局所最適領域を切り出している。この手法では距離情報を使って長い辺を切り離すことで複数の接続木を作る。しかし、高次元では複数クラスタ（局所最適領域）を作ることが困難である。探索次元に対して探索点数が十分でない場合、つまり、高次元探索空間での限られた探索点数は疎になり過ぎてしまい、探索点間距離が有意に変わらないことが原因であると言われている [3]。高次元での期待する距離以上に辺が長くなってしまう場合、NBCでは辺の切り離しが困難になってしまう。

NBCの辺切り離し法と異なり、WGrADでは、局所最良探索点同士を結ぶ長い辺を生成せず局所最良探索点と最近傍探索点を繋ぐために、重み付き距離情報と重み付き勾配情報を中核に用いる。これが、カテゴリC1関数の  $F_1, F_6, F_7$  で

WGrADがNBCよりも有意にうまく働いた理由である。しかし、局所最良探索点同士が近くにある場合は、これらの局所最良探索点同士が結びつきやすく、本来二つの局所最適領域として切り出すべきところを一つの局所最適領域と見做してしまう。これが、低次元のカテゴリC2関数でNBCにいつも同等以上にならない理由である。

Fig. 2は提案WGrADのこれらの特性を説明している。局所最良探索点Aが遠くに離れた局所最良探索点Cではなく近傍のfitnessの低い探索点Bと繋がり、辺A-Bを生成し、局所接続木を構成する。一方、接続辺C-E-F-Gは近傍に位置するので、二つの局所最良探索点同士が繋がり、本来多峰として分離すべき領域を一つの局所最適領域と見なしてしまっている。

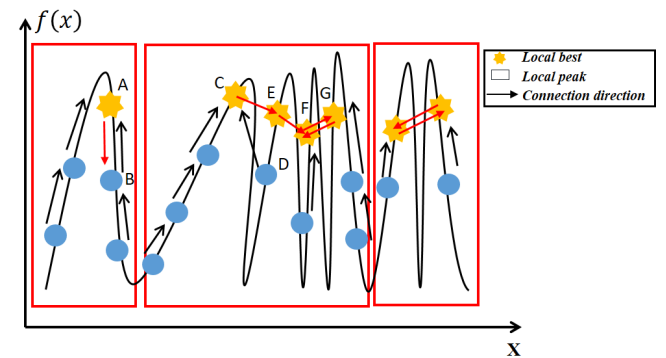


Fig. 2: 提案手法でniche領域切り出しに失敗する場合の例。局所最良探索点が近傍の局所最良探索点と繋がるとniche領域を見失うことがある。

第2に、ベンチマーク関数特性について考察する。低次元カテゴリC1関数では、局所最適解同士が互いに離れているので、局所最良探索点が見つかることができる。NBCとWGrADともniche領域の検出が容易な関数である。しかし低次元カテゴリC2関数の局所最適解同士は近いので、局所最良探索点も互いに近くなる。この場合NBCでは、長い辺を切り離しWGrADよりも多くの接続木を生成することができるため、NBCの方が多くの局所最適解領域を切り出しやすくなる。

検定結果から以下の4点に実験結果をまとめることができる。

1. 高次元ベンチマーク関数(4-D, 6-D, 8-D, 10-D, 20-D)では、WGrADが多くの局所最適領域を切り出し、 $F_1, F_2, F_6 - F_8$ でNBC2よりも良い性能を示す (Table 2(a))。

Table 2: 終了条件時までには検出された(a) 大局的最適解数の30試行平均値の検定, および, (b) 切り出された局所最適領域数の30試行平均値の検定. 検定は $p$ 値をHolmの多重検定で補正して有意差検定を行う.  $A \ll B$  および  $A < B$  は, 各々危険率1%と5%で有意にAよりBが良いことを意味する.  $A \approx B$  は両者間に有意な差がないこと,  $\sim$ は3手法間に有意差が見られないことを示す. 表中の0, 1, 2は, 各々(NBC2 + DE), (WGrAD<sub>1</sub>+ DE), (WGrAD<sub>2</sub>+ DE) 意味する.  $D$ は次元.

(a) 3手法で検出した大局的最適解数の差の検定

	カテゴリ	2-D	4-D	6-D	8-D	10-D	20-D
$F_1$	C1	$\sim$	$\sim$	$\sim$	$0 < 2 \approx 1$	$\sim$	$\sim$
$F_6$	C1	$\sim$	$0 \ll 1 \approx 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 < 2$	$0 \ll 1 < 2$
$F_7$	C1	$\sim$	$0 \ll 2 \approx 1$	$0 \ll 1 \approx 2$	$0 \ll 1 \approx 2$	$0 \ll 1 < 2$	$0 \ll 2 \approx 1$
$F_2$	C2	$\sim$	$0 \ll 1 \approx 2$	$0 \ll 1 < 2$	$0 \ll 2 \approx 1$	$0 \ll 1 \approx 2$	$0 \ll 1 \approx 2$
$F_3$	C2	$2 \ll 1 < 0$	$\sim$	$\sim$	$\sim$	$\sim$	$\sim$
$F_4$	C2	$\sim$	$0 \approx 2 < 1$	$\sim$	$\sim$	$\sim$	$\sim$
$F_5$	C2	$2 \ll 1 \approx 0$	$0 \approx 2 \ll 1$	$\sim$	$\sim$	$\sim$	$\sim$
$F_8$	C2	$\sim$	$0 \ll 1 \approx 2$	$0 \ll 1 \approx 2$	$0 \ll 1 \approx 2$	$0 \ll 1 \approx 2$	$0 \ll 1 \approx 2$

(b) 3手法で検出した切り出した局所最適領域数の差の検定

	カテゴリ	2-D	4-D	6-D	8-D	10-D	20-D
$F_1$	C1	$2 \ll 1 \ll 0$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$
$F_6$	C1	$1 \ll 2 \approx 0$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$
$F_7$	C1	$0 \approx 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$	$0 \ll 1 \ll 2$
$F_2$	C1	$2 \ll 1 \ll 0$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$
$F_3$	C1	$2 \ll 1 \ll 0$	$0 < 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$
$F_4$	C1	$2 \ll 1 \ll 0$	$0 < 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$
$F_5$	C1	$2 \ll 1 \ll 0$	$0 \approx 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$
$F_8$	C1	$2 \ll 1 \ll 0$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$	$0 \ll 2 \ll 1$

2. 低次元カテゴリC2関数では, NBC2がWGrADよりも良い性能を示す (Table 2(b)).

3. 低次元カテゴリC1関数 ( $F_1, F_6, F_7$ ) では3手法とも大局的最適解を探し出している.

これらより, 局所最適領域切り出しアルゴリズムの長所短所をTable 3にまとめる. 両手法は共に同じ距離情報とfitness値を用い, 性能的に補完関係にあるので, 両者を組み合わせることで両者の短所を補いあうことができる.

## 5 結論

重み付きの勾配情報と距離情報を用いたniche領域検出法を提案した. 実験結果から, 提案法と従来のNBC法とはお互いの短所を補う補完関係にあることを示した. 局所最適領域検出のための探索点クラスタリング方法の改良と, 特性が分からない最適化問題での両手法の切り替え方法の開発が, 今後の課題である.

Table 3: 次元に対するNBCとWGrADの一般的な性能

	低次元タスク	高次元タスク
C1	NBCとWGrADは共に良い性能を示す	WGrADが良い性能を示す
C2	NBCが良い性能を示す	

## 謝辞

本研究はJSPS科学研究費(課題番号18K11470, 19J11792)の助成を受けたものである.

## 参考文献

- 1) D. G. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," 2nd Int. Conf. on Genetic algorithms and

their application, Hillsdale, NJ, USA, pp. 41–49, July 1987.

- 2) X. Li, M. G. Epitropakis, K. Deb, and A. Engelbrecht, “Seeking multiple search points: An updated survey on niching methods and their applications,” *IEEE Trans. on Evolutionary Computation*, vol. 21, no. 4, pp. 518–538, Aug. 2017.
- 3) M. W. Pereira, G. S. Neto, and M. Roisenberg. “A Topological Niching Covariance Matrix Adaptation for Multimodal Optimization”. *IEEE Congress on Evolutionary Computation (CEC2014)*, Beijing, China, pp. 2562–2569, July 2014.
- 4) M. Preuss, “Niching the CMA-ES via nearest-better clustering,” *12th Annual Conf. Companion on Genetic and Evolutionary Computation (GECCO2010)*, Portland, OR, USA, pp. 1711–1718, 2010.
- 5) M. Preuss, “Improved Topological Niching for Real-Valued Global Optimization,” *Applications of Evolutionary Computation*, Berlin, Heidelberg: Springer, pp. 386–395, 2012.
- 6) B. Y. Qu, J. J. Liang, P. N. Suganthan, and Q. Chen, “Problem definitions and evaluation criteria for the CEC 2015 competition on single objective multi-niche optimization,” *Technical Report 201411B*, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and *Technical Report*, Nanyang Technological University, Singapore, Nov., 2014.
- 7) R. Storn and K. Price, “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.
- 8) R.K. Ursem, “Multinational evolutionary algorithms,” *IEEE Congress on Evolutionary Computation (CEC1999)*, Washington, DC, USA, USA, vol. 3, pp. 1633–1640, 1999.