法政大学学術機関リポジトリ
HOSEI UNIVERSITY REPOSITORY

PDF issue: 2025-07-15

複数解探索のための成長型粒子群最適化法について

竹村, 卓也 / TAKEMURA, Takuya

(出版者 / Publisher)
法政大学大学院理工学・工学研究科
(雑誌名 / Journal or Publication Title)
法政大学大学院紀要. 理工学・工学研究科編 / 法政大学大学院紀要. 理工学・工学研究科編
(巻 / Volume)
57
(開始ページ / Start Page)
1
(終了ページ / End Page)
3
(発行年 / Year)
2016-03-24
(URL)
https://doi.org/10.15002/00013044

複数解探索のための成長型粒子群最適化法について

A GWOING PARTICLE SWARM OPTIMIZER FOR IDENTIFICATION OF MULTI SOLUTIONS

竹村　卓也
Takuya TAKEMURA

指導教員 斎藤利通

法政大学大学院工学研究科電気工学専攻修士課程

This paper presents the growing insensitive particle swarm optimizer (GIPSO) for multi-solution problems. Especially, we consider the case where the number of solutions is unknown. The GIPSO uses ring-topology and has an insensitive parameter. The number of particles can increases and the swarm can grow. If parameter velues are selected suitably, the GIPSO can identify all the solutions and can clarify the number of solutions. Performing fundamental numerical experiments, we investigate the algorithm capability.
Key Words : *particle swarm optimizers, multi-solution problems*

1.　まえがき

粒子群最適化法 (Particle Swarm Optimizer,PSO)[1] は生物集団の振る舞いを元に考案された最適化手法である．生物は位置と速度を持つ粒子群にモデル化される．これらは解候補である．位置の評価は目的関数によって行われ，各粒子はトポロジーによって定められた粒子と良い位置の情報をお互いに交換し，速度を調整していくことで最適解を探索する．最適解とは目的関数が取り得る最小値や最大値のことを指す．PSO は概念的には単純で実装が容易なため，多くのシステムに適応されている．例として信号処理 [3]，フィルター設計 [4]，ニューラルネット学習 [5]，パワーエレクトロニクス回路設計 [6] に応用されている．

PSO は目的関数の勾配を使用しないアルゴリズムであり，微分不可能な問題に適応可能である．また簡単な目的関数であれば，広大な探索空間でも少量の粒子で最適解を探索することが可能であることから，大域探索に適しているといえる．現在，複数の最適解を持つ複数解問題 (Multi − solution − problems，MSP) において幾つか提案がなされてきた [7]-[10][11][12] [13]．しかしながら，すべての解を見つけることができず，十分に最適化されない問題がある．本論文では，大域探索の為, 粒子の成長の一端として粒子の増加を考慮した PSO を提案し，その動作の検討を行う．また, MSP を対象とした解の個数を自動的に判別するアルゴリズムを提案する．本 PSO は乱数要素を持たない決定論的な PSO である [11][12]．これは, 確定的な動作解析や再現性のある性能評価に有利である．次に, 系は離散的であり, 各粒子の動作は格子点に限定される．本アルゴリズムでは探索範囲において解の位置と個数を自動で判別するものである．見つけられた解に対して, 領域を設ける．その領域と位置の情報を参照し新たな解であるか, すでに見つけられた解かどうかを判断するものである．このアルゴリズムは様々な問題への応用が期待されるが, 本論文ではその第一歩として問題に対して適応する．基本的な数値実験によって, 解探索と解の位置と個数の探索の過程を調べる．本論文の結果は柔軟な複数解問題アルゴリズムを開発する為の基本情報となり得るかもしれない．

2.　本論

探索の対象とする目的関数は次式で与えられる．

$$\begin{aligned} &F_A : S_A \to R_+, \\ &S_A = \{(x_1, x_2) | X_L \le x_i \le X_R, i = 1, 2\} \end{aligned} \tag{1}$$

ただし, $[X_L, X_R]$ は各成分の探索範囲であり, $X_R > X_L$ とする．R_+ は非負の実数である．目的関数 F は次式で定義される複数解 $\vec{x}_s^i$ をもつものとする．

$$\begin{aligned} &F_A(x_s^i) = 0, \\ &x_s^i \equiv (x_{s1}^t, x_{s2}^t) \in S_A, i = 1 \sim N_A \end{aligned} \tag{2}$$

ただし, x_s^i は i 番目の解で, N_A は解の総数である．複数解問題では, N_A が未知の場合に, 探索空間 S_A 内の全ての解を見つけ出すことを目的とする．

便宜のため, 探査空間を分割数 Lp で格子点に分割し, 離散化する．S_D を $L_p \times L_p$ 格子点からなる次元 $D = 2$ の離散的探索空間とする．

解探索のための PSO の粒子数を N とする．離散時間 t における i 番目の粒子は P_i^t で表される．粒子 P_i^t のもつ情報は位置ベクトル $\vec{x}_i$, 速度ベクトル $\vec{v}_i$, によって特徴付けられる．ただし, $i = 1 \sim N$ である．今まで見つけた最良の適応度を $F(\vec{x}_{Pbest_i})$, $Pbest_i$ をパーソナルベストとする．群の中で最良の適応度を $F(\vec{x}_{Gbest_i})$, $Gbest_i$ をグローバルベストとする．通常, Gbest はその時点での近似解となる．

粒子の結合形態は様々あるが, 本論文では図 1 のリング結合を用いる．リング結合を用いたため両側の粒子が近傍粒子である．本論文では, $F(\vec{x}_{Gbest_i})$ とグローバルベストの代わりに, リング結合の場合に用いられる自身と最近傍の粒子の中で最良の適応度 $F(\vec{x}_{Lbest_i})$ とローカルベスト $Lbest_i$ に基づいて探索を行う．

また，本アルゴリズムは粒子の増加を含んでいる．これは，粒子全体で各粒子に多様性を持たせること，粒子の探索

範囲を拡大させること，などを狙って導入したものである．そのアルゴリズムを以下に示す．

STEP1: 粒子 P^i を初期化する．探索空間 S_A に $\vec{x}_i^t$ を $[X_L, X_R]$ の範囲でランダムに設定し、$\vec{v}_i^t$ を $[X_L/2, X_R/2]$ の範囲でランダムに設定する．

STEP2: 各粒子の位置 $\vec{x}_i^t$ を目的関数 $F_A(\vec{x_i^t})$ によって評価する．粒子 $\vec{P}_i^t$ の位置が式 (3) の条件を満たした時, それは近似解として記録される．

$$0 \leq F(\vec{x}_i^t) < C_A \tag{3}$$

STEP3: 各粒子のパーソナルベストとローカルベストを更新する．

$$\begin{array}{ll} \vec{x}_{pbest_i}^t = \vec{x}_i^t & \text{if } F(\vec{x}_i^t) < F(\vec{x}_{pbest_i}^t) \\ \vec{x}_{lbest_i}^t = \vec{x}_{pbest_i}^t & \text{if } F(\vec{x}_{pbest_i}^t) < (\vec{x}_{lbest_i}^t) \end{array}$$

STEP4: 速度ベクトルと位置ベクトルを更新する．

$$\begin{array}{l} \vec{v}_i^{t+1} \leftarrow w \times \vec{v}^t + c \times (\vec{x}_{lbest_i}^t - \vec{x}_i^t) \\ \vec{x}_i^{t+1} \leftarrow \vec{x}^t + \vec{v}_i^{t+1} \end{array} \tag{4}$$

位置ベクトル $\vec{x}_i^t$ は離散化された格子点 S_D 上に存在する．ただし, w と c は乱数要素を含まない決定論的なパラメータである．従来の PSO では, c は乱数要素を含むパラメータである．

STEP5: 記録された解がどの領域に存在しているかも判別する．

$$\begin{array}{l} \text{if } \|(\vec{x}_s^i - \vec{x}_i^t)^2\| > r \\ \vec{x}_S^i = \vec{x}_i^t \end{array} \tag{5}$$

ただし, 式 (5) はユークリッド距離を用いる．r は解の範囲を決める定数である．記録された粒子の位置情報を元にその評価値を参照し降順にソートする．最も値が小さい良い評価の点をはじめの解とし, 領域を設ける．すでに見つけた解とは別の領域の解だと判定されると, その粒子を新たな解と見なし, 新たに領域を設ける．次回の探索ではその新たな領域も考慮する．

STEP6: パーソナルベストが以下の式を十分に満たした時, その粒子の情報は初期化される．

$$\vec{x}_{pbest_i}^t < \frac{C_A}{2} \tag{6}$$

STEP7: 探索時間 t が時間 t_n に達するごとに現在の粒子の数だけ粒子の増加を行う．増加粒子は結合系の中間に増加され，増加粒子の情報は初期化されている．また，初期粒子数 $N_0 = \alpha$ とすると，増加粒子数は α である．また，粒子増加の概要を図 4 に示す．

STEP8: 最大探索時間 T_{max} に達するまで, $t \leftarrow t+1$ に更新し, **STEP2** に戻る．

3. 数値実験

アルゴリズムの性能の考察のため, 数値実験を行う．簡単のため次の基本的な問題を用いる．

$$F(\vec{x}_1,\vec{x}_2) \quad = \quad \cos(2\pi x_1) + \cos(2\pi x_2) + 2 \tag{7}$$

以下のパラメータ値を用いた．

$$w = 0.729, \quad c = 1.4779 \tag{8}$$

目的関数をを図 3 に示す．式 (7) の関数は, $[X_L, X_R]$ の範囲によって異なる数で格子点上に配置された最適値を持つ．F_1, F_2, F_3, F_4 における $[X_L, X_R]$ の値と解の個数は表 2 の関係である．

．図**??**における赤い範囲での中心が近似解基準 C_A を満た領域であり, それぞれの中点において最小値を取る．各パラメータは次のように設定した：近似解基準 $C_A = 0.05$, 最大探索時間 $t_{max} = 50$. 粒子は探索空間内に初期配置される．探索の後, 粒子は近似解にそれぞれ収束する．図 5 に粒子の探索過程を示す．図 2 に, 粒子の数を判定するために用いた, 1 つの最適解における領域を示す．その領域の範囲は半径 r で決まる．図 5(d) では, 解の個数として判定された解の位置と個数を示す．また，表 1 で粒子数と発見された解の個数の関係を示す．

式 (7) の問題では解の分布が均等である．この基本的な問題では, 正確な解の個数と位置を求めることができた．これは, 今後, アルゴリズムの性能を検討し, 改良していくための第一歩となると思われる．

4. むすび

PSO の，複数解の個数と位置を求める問題において粒子の数と解の個数について考察した．局所解への回避としてリング結合と情報の更新が行われない粒子に対しての初期化を用いた．また，粒子群全体の多様性を持たせるため粒子数の増加を導入した．以下の様な問題の考察は今後の課題である．トポロジーやパラメータの影響の検討 (最適化)，汎用性のある複数解の個数と位置探索アルゴリズムの開発，解の個数が多い問題や解の分布が複雑な問題への適応，工学的応用．

表 1 粒子の数，解の個数と見つかった解の個数の比較 (試行回数=10，#sol:存在する解)

N \ #sol	9	16	25	36
50	5.1	7.4	9.2	12.2
100	7.2	10.8	15.4	19.4

表 2 $[X_L, X_R]$ と解の個数の関係

$F(x)$	$F(1)$	$F(2)$	$F(3)$	$F(4)$
$[X_L, X_R]$	$[-2.0, 1.0]$	$[-2.0, 2.0]$	$[-3.0, 2.0]$	$[-3.0, 3.0]$
#sol	9	16	25	36

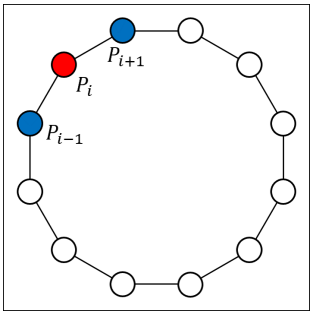

図 1 Ring topology.

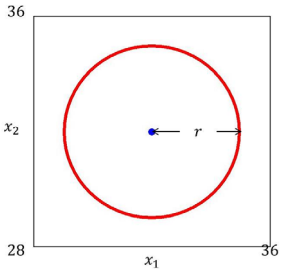

図 2 領域基準.

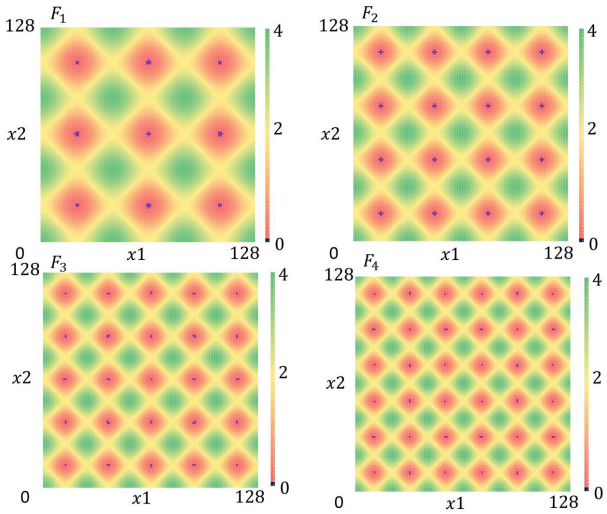

図 3 目的関数 (x_1, x_2:格子点数).

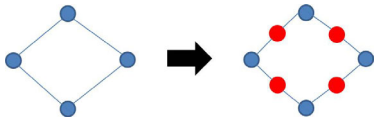

図 4 粒子増加.

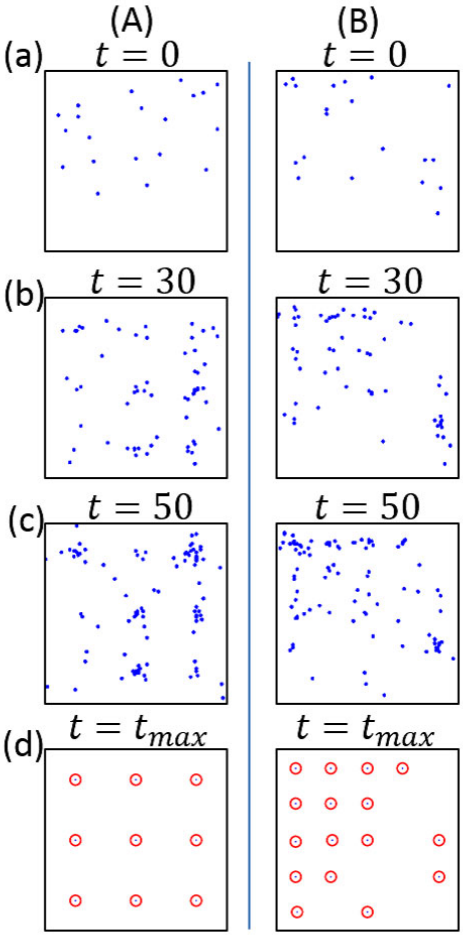

図 5 探索過程，A:$F(1)$　B:$F(3)$
((a)N=20，(b)N=60, (c)N=100).

参考文献

1) J. Kennedy and R. Eberhart, Particle Swarm Optimization, Proc of IEEE/ICNN, pp.1942-1948, 1995
2) A. P. Engelbrecht, Fundamentals of computational swarm intelligence, Willey, 2005.
3) M. P. Wachowiak, R. Smolikova, Y. Zheng, J. M. Zurada and, A. S. Elmaghraby, An approach to multimodal biomedical image registration utilizing particle swarm optimization, IEEE Trans. Evol. Comput., 8, 3. pp. 289-301, 2004.
4) R. A. Vural, T. Yildirim, T. Kadioglu and A. Basargan, Performance evaluation of evolutionary algorithms for optimal filter design, IEEE Trans. Evol. Comput., 16, 1. pp. 135-147, 2012.
5) A. B. van Wyk and A. P. Engelbrecht, Overfitting by PSO trained feedforward neural networks, in Proc. IEEE Congress Evol. Comput., pp. 2672 - 2679, 2010.
6) K. Kawamura and T. Saito, Design of switching circuits based on particle swarm optimizer and Hybrid Fitness Function, in Proc. Annu. Conf. IEEE Ind. Electron. Soc., pp. 1099-1103, 2010.
7) K. E. Parsopoulos and M. N. Vrahatis, On the computation of all global minimizers through particle swarm optimization, IEEE Trans. Evol. Comput., 8, 3, pp. 211-224, 2004.
8) D. Parrott and X. Li, Locating and tracking multiple dynamic optima by a particle swarm model using speciation, IEEE Trans. Evol. Comput., 10, 4. pp. 440-458, 2006
9) X. Li, Niching without niching parameters: particle swarm optimization using a ring topology, IEEE Trans. Evol. Comput., 14, 1. pp. 150-169, 2010.
10) T. Saito and E. Miyagawa, Growing-tree particle swarm optimizer with simple tabu search function, Proc. of NOLTA, pp. 376-379, 2009.
11) R. Sano, T. Shindo, K. Jin'no, and T. Saito, PSO-based multiple optima search systems with switched topology, in Proc. IEEE Congress Evol. Comput., pp. 3301-3307, 2012.
12) K. Maruyama, R. Sano and T. Saito, Deterministic discrete particle swarm optimizers with collision and insensitivity, Proc. NDES, pp. 245-248, 2012.
13) K. Maruyama and T. Saito, Deterministic Particle Swarm Optimizers with Collision for Discrete Multi-solution Problems, Proc. IEEE/SMC, pp. 1335-1340, 2013.