

粒子群最適化を用いた巡回セールスマン問題の解法

山田 悠希† 穴田 一†
 東京都市大学 大学院総合理工学研究科

1. はじめに

工業や経済の問題の多くは、最も効率が良い組み合わせを求める、組み合わせ最適化問題に帰着することができる。その中に、与えられた全ての都市を巡る最短経路を求める巡回セールスマン問題 (Traveling Salesman Problem, TSP) がある。本庄らは、最適化問題に用いられるアルゴリズムの一つである粒子群最適化 (Particle Swarm Optimization, PSO) [1] を TSP 向けに改良した挿入操作 PSO 戦略 (Insertion-based PSO strategy, IPSO) [2] を提案した。IPSO は、解空間上に配置された各粒子がそれまでの最良解と、近傍の粒子の最良解の情報を基に、解の更新を繰り返すことで解空間の探索を行うアルゴリズムである。しかし、この IPSO には探索が十分に行われないうちに、局所解に陥ってしまうという問題点がある。

そこで本研究では、既存手法で用いられた各粒子のそれまでの最良解と近傍の粒子の最良解の情報に加え、解空間上で最も遠い粒子の解の情報を現在の解に重ね合わせた解の集合を用いて、解の更新を行うアルゴリズムを構築した。そして、TSPLIB に掲載されているベンチマーク問題を用いて既存手法と提案手法の比較をすることで、その有効性を確認した。

2. 既存手法

2.1 挿入操作 PSO 戦略

本庄らが提案した IPSO は、PSO に基づき TSP の解空間の探索を行うアルゴリズムである。まず、解空間上に複数の粒子を配置する。これらの粒子は、それぞれ巡回路である解を持っており、各粒子のそれまでの最良解と近傍の粒子の最良解から抽出した部分経路を、各粒子の現在の解に挿入することで解の更新を行い、これを繰り返すことで、解空間を探索する。アルゴリズムの流れは以下の通りである。

①初期設定

各粒子 i に解 x_i をランダムに設定し、各粒子のそれまでの最良解 $pbest_i$ を現在の解 x_i に設定す

る。粒子 i と粒子 j 間の距離 d_{ij} を以下のように定義し、全粒子間の距離を計算する。

$$d_{ij} = \frac{1}{S_{ij}} \quad (1)$$

$$S_{ij} = \frac{|E_i \cap E_j|}{n}$$

ここで、 E_i は粒子 i が持つ解 x_i の経路の集合、 $|E_i \cap E_j|$ は E_i と E_j の共通している経路の本数、 n は都市数を表している。距離 d_{ij} は x_i と x_j の共通の経路が多くなるほど短くなる。次に、設定した近傍数 k を元に、粒子 i と距離が近い k 個の粒子を粒子 i の近傍に設定する。各粒子 i の近傍の中で総経路長が最も短い解を近傍内の最良解 $lbest_i$ と設定し、全粒子の中で最も総経路長が短い解を全粒子の最良解 $gbest$ と設定する。

②解の更新

解 x_i は $pbest_i$ の部分経路である $pbest_i'$ と $lbest_i$ の部分経路である $lbest_i'$ を総経路長が最も短くなるように挿入することで更新される。粒子 i の解の更新の詳細は以下の通りである。また、9都市の TSP の解の更新の例を図 1 に示す。図 1 の例の $x_i = (1,4,7,5,6,9,8,3,2)$ は都市 1 → 都市 4 → … → 都市 3 → 都市 2 と都市を巡り、都市 1 に戻る巡回路を表している。

I 部分経路の作成

粒子 i の $pbest_i$ から、 p 本の連続する経路をランダムに抜き出し、部分経路 $pbest_i'$ とする。また、粒子 i の $lbest_i$ から、 l 本の連続する経路をランダムに抜き出し、部分経路 $lbest_i'$ とする。 p と l は以下の式で表される。

$$p = [c_1 r_1 (n + 1)] \quad (2)$$

$$l = [c_2 r_2 (n + 1)] \quad (3)$$

ここで c_1 と c_2 は $[0,1]$ を満たすパラメータ、 r_1 と r_2 は $[0,1]$ を満たす一様乱数、 n は都市数である。 $[c_1 r_1 (n + 1)]$ は $c_1 r_1 (n + 1)$ の整数部分を表している。図 1 の例では $pbest_i' = (5,4,8,7)$ と $lbest_i' = (8,9,6)$ を抜き出している。

II $pbest_i'$ の再形成

$pbest_i'$ と $lbest_i'$ に共通した都市が含まれていれば、 $pbest_i'$ から該当した都市を削除し、残った都市で総経路長が最も短くなるよう部分経路を再形成する。図 1 の例では都市

8が共通しているため、 $pbest'_i = (5,4,8,7)$ から都市8を削除し、 $pbest'_i = (5,4,7)$ を再形成している。

III x'_i の形成

x_i から $pbest'_i, lbest'_i$ と共通する都市を削除し、残った都市で総経路長が最も短くなるよう巡回路を再形成し、 x'_i とする。図1の例では、 $pbest'_i$ と $lbest'_i$ にある都市4,5,6,7,8,9を x_i から削除し、 $x'_i = (1,3,2)$ としている。

IV $pbest'_i$ の挿入

$pbest'_i$ を x'_i に総経路長が最も短くなるよう挿入する。図1の例では都市1と都市3の間に $pbest'_i$ を挿入している。

V $lbest'_i$ の挿入

$lbest'_i$ を x'_i に総経路長が最も短くなるよう挿入する。図1の例では都市5と都市3の間に $lbest'_i$ を挿入している。

以上の I ~ V の操作を全粒子で行う。

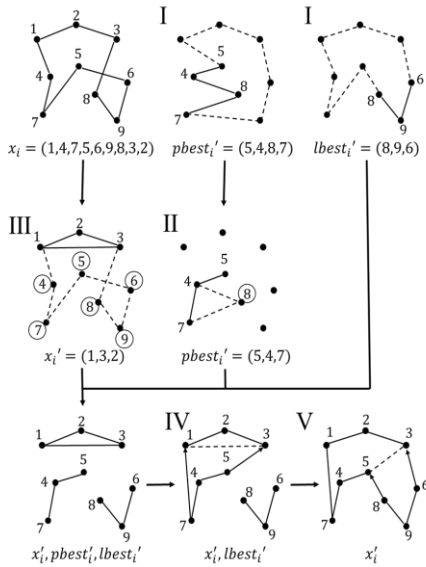


図1: 解の更新の例

③総経路長の計算

全粒子が持つ巡回路の総経路長の計算を行う。

④近傍の更新

全粒子間の距離を再計算し、近傍を更新する。

⑤ $pbest, lbest, gbest$ の更新

全粒子の $pbest, lbest$ と $gbest$ を更新する。

初期設定を①で行い、②から⑤までの操作を1イテレーションとし、事前に設定したイテレーション数繰り返すことでTSPの解空間を探索する。

2.2 既存手法の問題点

既存手法において、解 x の更新は各粒子の今ま

での最良解 $pbest$ と近傍の最良解 $lbest$ を用いて行われる。近傍数が2である場合、近くにいる2粒子のうち良い粒子の解が $lbest$ となるため、それぞれの粒子が参照できる粒子の情報が少ない。これでは、離れた粒子との組み合わせを試さないうちに探索が終了してしまい、全粒子の初期解の周辺を探索する多点探索とは言えない。

3. 提案手法

提案手法における解の更新は、各粒子 i のそれまでの最良解 $pbest_i$ 、近傍の粒子の解である $lbest_i$ 、最遠の粒子の解である x_i^f を現在の解 x_i に重ね合わせた経路集合 G を用いて行われる。まず、ある都市 i をランダムに選択する。そして G に含まれる経路から、次式で定義される確率 P_{ij} で次の経路 ij を選択する。

$$P_{ij} = \frac{w_{ij}}{\sum_{l=1}^n w_{il}} \tag{4}$$

$$w_{ij} = \frac{c}{(d_{ij})^D}$$

$$C = C_1 + C_2 + C_3 + C_4$$

$$C_1 = \begin{cases} c_1 & x_i \text{に含まれる場合} \\ 0 & \text{otherwise} \end{cases}$$

$$C_2 = \begin{cases} c_2 & pbest_i \text{に含まれる場合} \\ 0 & \text{otherwise} \end{cases}$$

$$C_3 = \begin{cases} c_3 & lbest_i \text{に含まれる場合} \\ 0 & \text{otherwise} \end{cases}$$

$$C_4 = \begin{cases} c_4 & x_i^f \text{に含まれる場合} \\ 0 & \text{otherwise} \end{cases}$$

ここで、 n は都市数、 $D, c_1 \sim c_4$ はパラメータを表している。(4)式は G に含まれる経路のうち、距離が短く、複数の解に含まれる経路を選択しやすくなるように設定している。 G に選択できる経路が存在しない場合、未訪問都市の経路の中から距離情報を用いたルーレット選択を用いて経路を選択する。この操作を繰り返すことで巡回路を構築していく。

結果、考察は発表時に述べる。

参考文献

[1] J.Kennedy, R.C.Eberhart, : "Particle swarm optimization"IEEE International Conf. on Neural Networks, pp.1942-1948 (1995).
 [2] 本庄将也, 飯塚博幸, 山本雅人, 古川正志, "巡回セールスマン問題に対する粒子群最適化の提案と性能評価", 日本知能情報ファジィ学会誌, vol.28, no.4, pp.744-755 (2016).