粒子群最適化を用いた巡回セールスマン問題の解法

Algorithm of Traveling Salesman Problem using Particle Swarm Optimization

山田 悠希^{*1} Yuki Yamada 穴田 一*1 Hajime Anada

*1 東京都市大学 Tokyo City University #1

Particle swarm optimization (PSO) is a population based stochastic optimization technique inspired by social behavior of bird flocking or fish schooling. PSO has been applied to various combinatorial optimization problems belonging to nondeterministic polynomial-time hard (NP-hard) combinational problems. Of these, the traveling salesman problem (TSP) is one of the most important problems in the fields of technology and science. Insertion-based PSO strategy (IPSO) is an algorithm that improved the PSO for TSP. The update of the particle in IPSO is done by using the information of the own best solution and the solution of the nearby particles. However, IPSO has a problem that it converges to the local optimal solution before sufficient search. Therefore, we proposed to provide the information of the farthest particle and the randomly selected particle as well as nearby particles for the particle's update. We confirmed the effectiveness of our algorithm using several benchmark problems taken from the TSPLIB, which is a library of traveling salesman problem.

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) という問題がある。 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)[Kennedy 95] をTSP向けに改良した挿入操作 PSO戦略 (Insertion-based PSO strategy, IPSO)を提案した[本庄 16]. 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。

そこで本研究では解の更新時に、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 粒子の情報とランダムに選択した粒子の情報を与えることで、既 存手法よりも広範囲の探索を行える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、性能の向上を確認した.

2. 既存研究

2.1 粒子群最適化

粒子群最適化(Particle Swarm Optimization, PSO)とは、魚や 鳥などに見られる群行動を探索手法に応用した、最適化手法 の一つである. 解空間上に位置と速度を持った複数の個体(以 下,粒子と表記)をランダムに配置する. 各粒子の位置は問題の 解を表現しており、適応度の高い位置の情報を近傍の粒子と交 換し、その情報を基に、より良い位置に近づくように速度と位置 を更新する. この操作を繰り返すことで、解空間を探索する. t イテレーション目における粒子 i の位置 x_i(t) と速度 v_i(t) の更 新式は次式で定義される.

連絡先:山田悠希,東京都市大学,〒158-8557 東京都世田谷区玉堤 1-28-1

$$x_{i}(t) = x_{i}(t-1) + v_{i}(t-1)$$
(1)

$$v_{i}(t) = wv_{i}(t-1) + c_{1}r_{1}(pbest_{i} - x_{i}(t))$$
(2)

$$+c_2r_2(lbest_i - x_i(t))$$

ここで, w, c_1 , c_2 はパラメータ, $r_1 \ge r_2$ は [0,1] の一様乱数, pbest_i はその粒子 i のそれまでの最良解, lbest_i は粒子 i の 近傍の中のそれまでの最良解である. PSO のアルゴリズムの流 れは以下の通りである.

①初期設定

全粒子の位置と速度をランダムに設定し、各粒子 i のそれま での最良解 $pbest_i$ を現在位置に設定する.次に、設定した近 傍数 k を元に、各粒子 i と距離が近い k 個の粒子を粒子 i の 近傍に設定する.そして、各粒子 i の近傍の中で適応度が最も 高い解を近傍内の最良解 $lbest_i$ と設定し、全粒子の中で最も 高い適応度の位置を gbest と設定する.

②位置の更新

各粒子において,(1)式に従い,位置の更新を行う.

③適応度の評価 全ての粒子において適応度の計算と評価を行う.適応度は 問題に適した粒子ほど高くなるよう,評価関数を事前に設定し ておく.

④ pbest_iと gbest の更新

各粒子iが新たに得た解の適応度がこれまでの $pbest_i$ よりも 高かった場合, $pbest_i$ を更新する.近傍内で適応度が最も高い 解が $lbest_i$ よりも短かった場合, $lbest_i$ をその解で更新する. この操作を全粒子に行う.全粒子の中で最も適応度が高い解 がgbestよりも適応度が高かった場合,gbestをその解で更新 する.

⑤速度の更新

各粒子において、(2)式に従い、速度の更新を行う. 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、終了条件を満たすまで繰り返すことで解空間を探索する.

2.2 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. IPSO のアルゴ リズムの詳細な流れは以下の通りである.

①初期設定

各粒子に解 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.

$$d_{ij} = \frac{1}{S_{ij}}$$
(3)
$$S_{ij} = \frac{|E_i \cap E_j|}{n}$$

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 $x_i \ge x_j$ 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 $lbest_i$ と設定する。また、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 $i \circ pbest_i$ 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 $i \circ o$ $lbest_i$ 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. $p \ge l$ は以下の式で表される.

 $p = [c_1 r_1 (n+1)] \tag{4}$

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

ここで $c_1 \ge c_2$ は [0,1] を満たすパラメータ, $r_1 \ge r_2$ は [0,1] を満たす一様乱数, n は都市数である. $[c_1r_1(n + 1)]$ は $c_1r_1(n + 1)$ 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.

(5)

II pbest'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*' から都市 8 を削除 し, *pbest_i*' を再形成している.

III x_i'の形成

 x_i に pbest_i'と lbest_i'と共通する都市が含まれていれ ば、 x_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、 $x_i' = (1,2,3)$ となっている.

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 $I \sim V$ の操作を全粒子で行う.

③総経路長の計算

全ての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

粒子 i が新たに得た解 x_i' の総経路長が $pbest_i$ よりも短かっ た場合, $pbest_i \ge x_i'$ で更新する. 近傍内で総経路長が最も短 い解がこれまでの $lbest_i$ よりも短かった場合, $lbest_i$ をその解 で更新する. この操作を全粒子に行い, 全粒子の中で総経路 長が最も短い解がこれまでの gbest よりも短かった場合, gbest をその解で更新する.

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、終了条件を満たすまで繰り返すことで TSP の解空間を探索 する.



3. 既存手法の問題点

既存手法である IPSO は、各粒子 i のそれまでの最良解であ る $pbest_i$ と近傍の粒子の最良解である $lbest_i$ の部分経路を挿 入することで解 x_i の更新を行っている.しかし、 $pbest_i$ と $lbest_i$ は探索がある程度進むと、 x_i とほぼ同じような巡回路と なり、部分経路を挿入しても、元の解と全く同じ解ができる無駄 な更新が起こりやすい.この無駄な更新の影響で、新規の経路 の獲得が起こりにくく、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.

4. 提案手法

IPSOの解の更新時における部分経路の作成において,挿入 要素を抽出する解を $lbest_i$ から次式で表される x_i^{ins} に変更した. $(lbest_i with probability P_l$

$$x_i^{ins} = \begin{cases} x_i^f \text{ with probability } P_f \\ x_i^r \text{ with probability } P_f \end{cases}$$
(6)

 $(x_i^* with probability P_r)$

ここで、 $lbest_i$ は近傍内の最良解、 x_i^f は解空間上で一番遠い

粒子の解, x_i^r は自分以外の粒子の中からランダムに選択され た粒子を表しており, x_i^{ins} は lbest_i, x_i^f , x_i^r の三つの解から確 率的に選択される解である. また, P_l は lbest_i が選ばれる確率, P_f は x_i^f が選ばれる確率, P_r は x_i^r が選ばれる確率である. 最 遠の粒子の解 x^f とランダムに選択された粒子の解 x^r の部分 経路を挿入することで,既存手法では獲得できなかった経路を 獲得でき,各粒子が広範囲の探索を行うことが期待される.

5. 結果

提案手法の有用性を確認するため、TSPLIB に掲載されている TSP のベンチマーク問題である rd100, kroA150, kroA200, pr299 を用いて評価実験を行った.既存手法,提案手法ともに事前実験により決定された粒子数 m = 64,近傍数 k = 2, $c_1 = 0.9$, $c_2 = 0.1$ というパラメータを使用した.また,提案手法の x^{ins} において *lbest* を選択する確率 P_l , x^f を選択する確率 P_f , x^r を選択する確率 P_r はそれぞれ,事前実験により決定された, $P_l = 0.6$, $P_f = 0.3$, $P_r = 0.1$ を使用した.終了条件はrd100 と kroA150 は 20000 イテレーション, kroA200 と pr299 は 30000 イテレーションとした.各問題 50 試行の結果を表 1~4 に表す.また,表中で用いられている誤差率は,試行内で得られた最良解の最適解に対する誤差の割合を,最終更新イテレーションは最後に gbest を更新したイテレーションを表している.

表 1:rd100 の 5	0 試行の結果
---------------	---------

rd100(最適解 7910)	既存	提案	
最適解到達率(%)	88	98	
平均誤差率(%)	0.0099	0.0002	
誤差率の標準偏差	0.0601	0.0018	
平均最終更新 イテレーション	3210.4	1961.0	
平均時間(秒)	8.3	6.5	

提案	既存	kroA150(最適解 26524)
34	10	最適解到達率(%)
0.05	0.25	平均誤差率(%)
0.12	0.24	誤差率の標準偏差
4349.9	6951.2	平均最終更新 イテレーション
33.0	32.7	平均時間(秒)

表 2:kroA150 の 50 試行の結果

表 3:kroA200の 50 試行の結果

kroA200(最適解 29368)	既存	提案
最適解到達率(%)	2	20
平均誤差率(%)	0.50	0.25
誤差率の標準偏差	0.29	0.25
平均最終更新 イテレーション	13684.2	12206.1
平均時間(秒)	100.5	88.7

表	4:	pr299	\mathcal{O}	50	試	行(の結	果
~				~~	н •		- C /I'H	~

-		
pr299(最適解 48191)	既存	提案
最適解到達率(%)	0	2
平均誤差率(%)	1.13	0.63
誤差率の標準偏差	0.39	0.44
平均最終更新 イテレーション	21655.9	20767.4
平均時間(秒)	261.7	247.7

実験の結果,既存手法に比べ,最適解到達回数,平均誤差率,平均最終更新イテレーション,平均時間のほとんどの項目において提案手法の方が良いことを確認した.これは, x^f と x^r の部分経路を挿入することによって,*lbest*の挿入だけでは獲得できなかった経路の組み合わせを獲得できたことが要因だと考える.

続いて, pr299 の更新した粒子の割合の推移を表したグラフ を図1に表す.縦軸は更新した粒子の割合,横軸はイテレーションを表し,黒線は既存手法,赤線は提案手法の推移を表している.



グラフより,既存手法に比べ,提案手法の方が更新した粒子の 割合が全イテレーションにおいて多いことが分かる.これは,既 存手法の問題点である無駄な更新を減らし,効率の良い探索 が行えていると言える.

6. 今後の課題

今後の課題として、全粒子が持っていない新規の経路を獲 得することが挙げられる.提案手法では、全粒子が持っていな い新規の経路の獲得が難しく、最適解ではない解に収束しや すいという欠点がある.そこで、解の更新が一定のイテレーショ ン行われなかった場合、いくつかの粒子の経路を入れ替えると いった収束に対する刺激を与えることで、現在の提案手法よりも 最適解に到達しやすいアルゴリズムにしたいと考えている.

参考文献

- [Kennedy 95] J.Kennedy, R.C.Eberhart, : "Particle swarm optimization"IEEE International Conf. on Neural Networks, pp.1942-1948 (1995)
- [本庄 16] 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)