

TSP の新しいアルゴリズム「粒子速度調整法」の提案

井出 恵介[†] 長谷川 智史[‡] 穴田 一[‡]

東京都市大学工学研究科[†]

東京都市大学知識工学部[‡]

1. 緒言

本研究では、TSP (Traveling Salesman Problem) を解く新しいアルゴリズム「粒子速度調整法」を提案する。従来の TSP のアルゴリズム(2-opt 法や Lin-Kernighan 法) [1]では経路に着目し、経路の操作を行うものがほとんどであった。しかし、提案アルゴリズムでは都市に見立てた粒子を操作することで経路を探索する。具体的には都市に見立てた粒子を円周上に置き、粒子に異なる速度を持たせ、順番を変えることで経路探索を行うアルゴリズムである。本研究ではこの提案アルゴリズムを用いて、20 都市の TSP を解くことに成功した。

2. 提案アルゴリズム

2.1 粒子速度調整法

まず図 1 のように円周上にランダムに都市に見立てた粒子を配置し、異なる速度を乱数を用いて与える。そして単位時間だけ粒子を進める。もし、進める前よりも経路距離が短くなった場合、進める前と変化した経路順が原因で短くなったことがわかる。そこでこの経路順を保存することにより経路距離が改善していくことができるのではないかと考えた。経路順を保存する方法として粒子の速度を調整する速度調整アルゴリズムを提案する。本研究では速度調整アルゴリズムとして 4 つの方法を試した。4 つの方法について下記に示す。

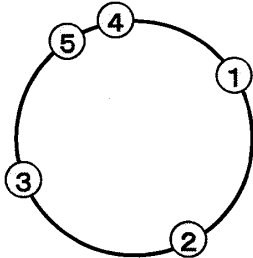


図 1 粒子モデル

- 平均法
保存したい粒子群の速度をそれらの平均値で更新する。
- 乱数法
保存したい粒子群の速度を同じ値の乱数で更新する。
- 乱数・位置幅法
保存したい粒子群の位置を等間隔に幅を狭めて固め、粒子の速度に同じ値の乱数を代入する。
- 速度ソート法
保存したい粒子群の速度を粒子の進行方向が昇順になるように並び替えを行う。そして次式を

用いて粒子速度を調整する。

$$V_i = \beta \{V_i - \alpha (V_i - V_{mean})\}$$

ここで V_i は粒子 i の速度、 V_{mean} は粒子群の平均速度、 α は粒子の速度を平均値に近づけるパラメータ、 β は速度の減衰率で、それぞれ 0 以上 1 以下の実数である。

本研究では速度調整アルゴリズムを用い経路探索を行う一連のアルゴリズムのことを粒子速度調整法という。粒子速度調整法の手順を下記に示す。

- Step1:初期値(粒子の位置、速度)を乱数で設定
- Step2:現時点における粒子の位置より経路を計算
- Step3:単位時間粒子を進め経路を計算
- Step4:Step2 の距離より Step3 の距離が長くなったら Step6 に移動
- Step5:速度調整アルゴリズムを用い粒子の速度を更新
- Step6:粒子の速度を基に粒子の位置を更新

Step2 から Step6 までを繰り返し、収束するまで計算する。

2.2 粒子結合アルゴリズム

2.1 の粒子速度調整法では 20 都市が求まらなかったため、粒子速度調整法の補助として粒子結合アルゴリズムを考えた。まず、粒子速度調整法を用いた同じプログラムを複数用意する。次に各々に異なる初期設定を与え、並列に計算する。もし、計算結果に共通の並びで、かつ最も多い並びを、一つの並びとして結合をする。たとえば、5-4 の並びが複数の計算結果の共通の並びで、他の並びより多い場合に 5-4 を一つの粒子と見立て、全ての粒子の位置、速度を初期設定し、粒子速度調整法を用いて再計算を行うアルゴリズムである。粒子結合アルゴリズムの手順を下記に示す。

- Step1:粒子速度調整法を並列に解が収束するまで計算
- Step2:Step1 の計算結果から最も多かった共通の経路順を抽出
- Step3:Step2 で抽出した経路順を一つの粒子として結合し Step1 に戻る

※Step3 から Step1 に戻る際に粒子の位置、速度は

A New Algorithm for the TSP

[†] Keisuke Ide, Research Division in Engineering, Tokyo City University

[‡] Satoshi Hasegawa, Hajime Anada, Faculty of Knowledge Engineering, Tokyo City University

乱数で再設定を行う。Step1 から Step3 を繰り返し収束するまで計算する。なお、粒子数は Step3 で結合することによって減少する。

2.3 斥力アルゴリズム

2.2 の粒子結合アルゴリズムでも 30 都市における TSP の厳密解が求まらなかった。そこで 2.1 の速度調整アルゴリズムについて考えてみる。速度調整アルゴリズムは経路距離が改善した都市の並び順を保存するために、経路距離が短くなったときに速度調整を行った。しかし、悪い並び順になったときは何もしてこなかった。そのため経路距離の改善が遅れているのではないかと考えた。そこでこの斥力アルゴリズムでは粒子速度調整法の改良として悪い並び順になったときにも速度調整することを考えた。次式を使い速度調整を行った。次式の狙いは悪い並び順を崩すことである。そこで平均との差分の分だけ平均から速度を離すアルゴリズムにした。

$$V_i = \beta \{V_i + \alpha (V_i - V_{mean})\}$$

ここで V_i は粒子 i の速度、 V_{mean} は粒子群の平均速度、 α は粒子の速度を平均値から遠ざけるパラメータ、 β は速度の減衰率で、それぞれ 0 以上 1 以下の実数である。

斥力アルゴリズムを加えて改良した粒子速度調整法の手順を下記に示す。

- Step1:初期値(粒子の位置、速度)を乱数で設定
- Step2:現時点における粒子の位置より経路を計算
- Step3:単位時間粒子を進め経路を計算
- Step4:Step2 の距離より Step3 の距離が長くなったら Step6 に移動
- Step5:粒子速度調整法を用い粒子の速度を更新, Step7 に移動
- Step6:斥力アルゴリズムを用い粒子の速度を更新, Step7 に移動
- Step7:粒子の速度を基に粒子の位置を更新

※Step2 から Step7 まで繰り返し、収束するまで計算する。

3. 結果・考察

まず、粒子速度調整法の性能を評価するために、20 都市の TSP(厳密解 3.12)を解いた。シミュレーション結果を下記に示す。

	最短	平均	分散
平均法	7.11	10.6	2.69
乱数法	6.02	7.78	0.627
乱数・位置幅法	5.77	7.71	0.457
速度ソート法	6.60	9.48	2.88

表 1:シミュレーション結果

(1×1 の中に都市数を 20, 厳密解は 3.12, 一周 100, 初期設定の位置は一様乱数 (0~100), 速度は一様乱数 (10~30), 一回の試行によるステップ数 1 万回, 試行回数を 50 回)

表 1 の結果より厳密解が 3.12 にも関わらず、計算した最短距離が厳密解の約 2 倍になり、20 都市の TSP を解くことができなかった。そこで同じ問題を粒子結合アルゴリズムで解いた。シミュレーション結果を下記に示す。

	最短	平均	分散
平均法	4.93	6.93	0.606
乱数法	3.12	4.53	0.399
乱数・位置幅法	3.12	5.12	0.640
速度ソート法	3.12	4.64	0.497

表 2:改善アルゴリズムを入れたシミュレーション結果

(1×1 の中に都市数を 20, 厳密解は 3.12, 一周 100, 初期設定の位置は一様乱数 (0~100), 速度は一様乱数 (10~30), 並列プログラム数は 10, 一回の試行によるステップ数 1 万回, 試行回数を 50 回)

表 2 の結果より平均法以外は厳密解を求めることができた。よって提案アルゴリズムで 20 都市の TSP を解くことができた。しかし、都市数を 30 都市に増やしてみたところ、厳密解を出すことができなかった。そこで斥力アルゴリズムを加えた改良粒子速度調整法を用い粒子結合アルゴリズムを行った。そのときの速度調整アルゴリズムは速度ソート法を用いた。

シミュレーションの設定を下記に示す。

- ・ 1×1 の中に都市数を 30
- ・ 厳密解は 3.12
- ・ 一周の長さ 100
- ・ 初期設定の位置は一様乱数 (0~100)
- ・ 速度は一様乱数 (10~30)
- ・ 並列プログラム数は 10
- ・ 試行回数は 10 回

現在、シミュレーション途中であるので結果は発表で述べる。今後の方針は有名な TSP のベンチマークを複数行い、性能の評価や優位性について研究していくことを考えている。

参考文献

- [1] 本橋瞬, 松浦隆文, 池口徹, IEICE technical report. Nonlinear problems 107(561) pp.43-48 20080321
 ※概要の枚数制限上, 一部の参考文献は記載していない