情報処理学会研究報告
IPSJ SIG Technical Report

Vol.2009-MPS-76 No.44
Vol.2009-BIO-19 No.44
2009/12/18

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††

本研究は,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.本アルゴリズムは,今までのアルゴリズムと全く違うアルゴリズムである.今までの経路決定アルゴリズム(2-opt 法や Lin-Kernighan 法)は経路に着目してきたが,提案するアルゴリズムでは都市に着目したアルゴリズムを考えた.都市を粒子と見立て,粒子に速度を持たせることで経路探索を行った.本研究では 4 つの速度調整アルゴリズムと粒子結合アルゴリズムを提案した.さらにアルゴリズムの性能を確かめるため 20 都市の TSP を解き評価を行った.

A New Solution Algorithm of the TSP

Keisuke Ide†, Satoshi Hasegawa†† and Hajime Anada††

The purpose of this study is a new solution algorithm of the TSP. However, most of the research on connecting and changing of path. So we focus attention on order of cities. Our algorithm dissolved the TSP in 20 cities. The results in the present investigation suggest that our algorithm could solve the TSP in many cities.

1. はじめに

TSP(Traveling Salesman Problem)は全ての都市を 1 回ずつまわる最短経路を求める問題である.この問題は都市数が多くなると経路の組み合わせ爆発がおき［1］,計算困難となり様々な解法が提案されている.解法は厳密解法と近似解法に大別される.厳密解法は必ず最短経路を計算できるが,探索時間が長いという欠点がある.近似解法は厳密解法と違い,短い時間である程度の精度の解を求めることができる.しかし,厳密解法のように厳密解を求められるという保障はない［2］.近似解法の分野では PSO(Particle Swarm Optimization)と SA(Simulated Annealing)を組み合わせて解くアルゴリズム[3]や ACO(Ant Colony Optimization)と GA(Genetic Algorithm)を組み合わせて解くアルゴリズム[4]などがある.また,経路探索のアルゴリズムとして 2-opt 法や Lin-Kernighan 法などがよく研究されている[5].

本研究では TSP を解く新しいアルゴリズムを提案する.今までの経路探索アルゴリズムは 2-opt 法や Lin-Kernighan 法など,経路に着目点を置いたアルゴリズムが主に研究されてきた[6].しかし,本研究で提案するアルゴリズムは都市に着目点をおき,経路を探索するアルゴリズムとなっている.

提案アルゴリズムでは,円周上に都市を意味する速度のある粒子を置き,粒子の並び順から経路を求めるアルゴリズムである.まず,粒子の位置や速度を乱数で設定する.そして，粒子が移動し前回の経路よりも距離が改善された場合,前回と変わった経路順が原因で距離が短くなったと考えられる.そこで,変更した経路順を保存するため,４つの速度調整アルゴリズムを提案した.また,速度調整アルゴリズムだけでは解が思うように厳密解へ近づかなかったため,粒子結合アルゴリズムを考えた.粒子結合アルゴリズムとは,並列に速度調整アルゴリズムを収束するまで計算し,複数の計算結果から同じ経路順が出た場合,経路を構成している粒子群をひとつの粒子にまとめ,再度位置や速度を乱数で設定し並列に計算する.そして,並列に計算している結果が同じになるまで繰り返す.

本研究では,20 都市の TSP を解き各アルゴリズムの性能を比較した.その結果,平均法以外はすべて厳密解を求めることができた.

† 東京都市大学大学院工学研究科
Graduate School of Engineering, Tokyo City University
†† 東京都市大学知識工学部
Faculty of Knowledge Engineering, Tokyo City University

Vol.2009-MPS-76 No.44
Vol.2009-BIO-19 No.44
2009/12/18

2. 提案アルゴリズムの概要

2.1 TSP とは

TSP は，すべての都市を一度ずつ回り，出発した都市に戻ってくる距離が最小になる経路を求める問題のことである.本研究では 2 都市の行きと帰りの道では距離が変化することがない問題を研究対象にする.つまり,ユークリッド空間での対称型巡回セールスマン問題を考える.この問題では逆順も同一経路とみなすので,総組み合わせは(N-1)!/2 通りになり,都市数が増えると組み合わせ爆発が起こる.

2.2 提案アルゴリズムの基本モデル

図 1 は経路順を意味している.円周上にある粒子は都市を意味しており,それぞれの粒子は右方向に速度を持っている.図 1 は,右回りに 1→2→3→5→4 という経路を表している.

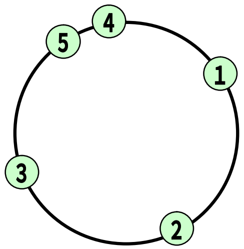

図 1 粒子モデル

提案アルゴリズムの手順を以下に示す.

Step1：初期値(粒子の位置,速度)を乱数で設定.
Step2：一定時間粒子を進め試行経路を計算.
Step3：Step2 を行う前の距離よりも短くなったら Step5 に移動.
Step4：粒子速度調整法を用い粒子の速度を更新.
Step5：粒子の速度を基に粒子の位置を更新.

Step2 から Step5 までを繰り返し収束するまで計算する.

2.3 粒子速度調整法

2.2 の Step3 で出てきた Step2 を行う前の距離よりも短くなった場合,経路順が変化したことが原因で経路が改善したと考えられる.そこで,変化した経路順を保存していくことを考えた.

本研究では,経路を保存する方法として,変化した経路の粒子の組(粒子群)の速度を更新する 4 つの方法を提案する.

2.3.1 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.

2.3.2 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.

2.3.3 乱数・位置幅法

保存したい粒子群の位置関係を狭く等間隔に固め,粒子の速度に同じ乱数の値を代入する.

2.3.4 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に,値が大きく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.

$$V_i = \beta\{V_i + \alpha(V_{mean} - V_i)\} \quad (1)$$

ここで V_i は都市 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 は速度の減衰率である. α ならびに β は,0 以上 1 以下の実数が入る.

2.4 粒子結合アルゴリズム

本研究では粒子速度調整法を用いて速度の更新を行っている.しかし,それだけでは厳密解に近づかなかったため,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.アルゴリズムの手順について下記に述べる.

Step1:粒子速度調整アルゴリズムを並列に解が収束するまで計算.
Step2:Step1 の計算結果から同じ経路順を抽出.
Step3:Step2 で抽出された経路順をひとつの粒子として結合.
Step4:再度各々の粒子に対し乱数を振りなおし計算

Step2 から Step4 を繰り返し収束するまで計算する.なお,Step4 の粒子数は Step3 で結合した数により変動をする.

3. 結果・考察

4 つの粒子速度調整法を比較するため,20 都市の TSP(厳密解 3.12)を解いた.なお,シミュレーション条件を下記に示す.

- 試行回数は 50 回
- 一試行に対してステップ数は 10000 回
- 円一周の距離 100
- 初期設定の位置は一様乱数(0～100)

- 初期設定速度は一様乱数(10～30)

計算結果を表 1 に示す.

表 1　計算結果

	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 を見ると全てのアルゴリズムにおける解が,厳密解の約 2 倍以上になった.さらに早い段階で局所解に収束し,厳密解になっていないことが確認できた.そこで粒子結合アルゴリズムを用いて計算結果を表 2 で示す.

表 2　粒子結合アルゴリズムを入れた計算結果

	最短	平均	分散
平均法	4.93	6.93	0.606
乱数法	3.12	4.52	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 をみると平均法以外は厳密解を出すことができた.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.

粒子結合アルゴリズムを入れることにより,局所解に落ちることを避けることができたと考える.その理由は 2 つある.初めに,並列で計算された結果の共通経路順は,厳密解の一部である可能性が非常に高いからである.次に,粒子を結合することによって探索する解空間が狭くなっているため,問題は簡単になり局所解に陥りにくくなる.

4.　今後の課題

今回,性能を比較する際 20 都市のみで検証した.本提案アルゴリズムが有効か都市数を増やしての検証の必要性がある.また，都市数が増えることにより収束までの時間という点も考慮し 4 つの提案アルゴリズムに関して優劣をつけようと考えている.

5.　参考文献

[1] A.V エイホ,J.E.ホップクロフト,J.D.ウルマン,データ構造とアルゴリズム,培風館, 1987.

[2]増井忠幸,百合本茂,片山直登,ロジスティクスの OR,槙書店, 1998.

[3]高橋良英,拡張遺伝子交叉オペレータ交代法による巡回セールスマン問題の解法, IEEJ Transactions on Electronics, Information and Systems, 128(12), pp.1820-1832, 2008.

[4] Fang.L., Chen. P., Liu.S., Particle Swarm Optimization with Simulated Annealing for TSP, WSEAS Transactions on Information Science and Applications, Volume 4, Issue 6, June 2007, Pages 1157-1162, 2007.

[5]本橋瞬,松浦隆文,池口徹,Lin-Kernighan アルゴリズムをカオス駆動する巡回セールスマン問題の解法,IEICE technical report, Nonlinear problems,107(561),pp.43-48,2008

[6] Wen-liang, Zhong, JunZhang, A Novel Discrete Particle Swarm Optimization to Solve Traveling Salesman Problem, 2007 IEEE Congress on Evolutionary Computation, 2007.