

球面上における巡回セールスマン問題ソルバーの開発と評価

小林 奎太[†]黒田 久泰[‡]愛媛大学工学部情報工学科[†] 愛媛大学大学院理工学研究科[‡]

1. はじめに

近年、グローバル化が進み、海外との交流が盛んになってきている。それと同時に海外へ出向く際には効率よく目的地を回る必要性も高まっている。本研究では、球面上での巡回セールスマン問題について考察する。

巡回セールスマン問題とは、複数個の都市が与えられたとき、それら全ての都市を一度ずつ巡り出発地に戻る巡回路の総移動コストが最小であるものを見つける組み合わせ最適化問題である[1][2]。この問題は実用上の応用も多く、配送計画やロボットの動作計画などに適用されている。しかし、巡回セールスマン問題は最適解を求める多項式時間アルゴリズムが見つからないとされている。そこで、短時間で近似解を求めることができる近似アルゴリズムが重要視されるようになった。近似アルゴリズムは巡回路を作成する構築法と、その巡回路に対して処理を行う改善法がある。

2. 構築法

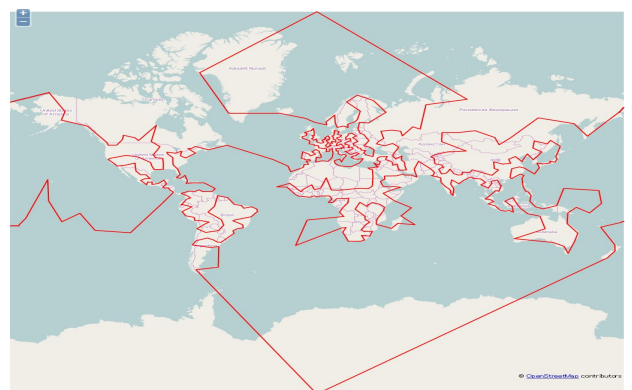
2.1 Nearest Neighbor 法

Nearest Neighbor 法は構築法の一つで、出発地点から最も距離が近い都市を選んでいく方法である。最も距離が近い都市を選んでいくので前半は効率よく都市を選択できるが、後半にコストの高い経路を選んでしまうという欠点がある。実装した Nearest Neighbor 法の手順を以下に示す。

- i. 都市の座標が入った配列を用意し、配列の先頭を注目都市とする。
- ii. 注目都市から最も近い都市を注目都市の次の位置にある都市と入れ替え、新たな注目都市とする。
- iii. 注目都市が配列の末尾の一つ前になるまで ii を繰り返す。
- iv. 配列内の都市の順番を訪問順として巡回路を作成する。

- v. i から iv を全ての出発点のパターンを試し、総移動コストが最も少なかったパターンを最終的な巡回路とする。

本稿では、最適解がわかっている 666 都市の問題(`gr666.tsp`)[3]を対象とした(図 1)。この問題の最適解は図 2 に示す巡回路となり巡回路の総距離は 294,358km である。Nearest Neighbor 法のみを用いた場合の結果は、総距離が 355,941 km となり、最適解との誤差は 20.92% となった。実行時間は 1.17 秒である。なお、プログラミング言語として C 言語(コンパイラは gcc 4.4.7)を用い、時間計測を行った PC の CPU は Intel Xeon E5-2660 2.20GHz である。

図 1 : 666 都市の問題(`gr666.tsp`)図 2 : 666 都市の問題(`gr666.tsp`)の最適解

Development and Evaluation of Traveling Salesman Problem Solver on Spherical Surface

[†]Keita KOBAYASHI

Computer Science, Faculty of Engineering, Ehime University

[‡]Hisayasu KURODA

Graduate School of Science and Engineering, Ehime University

2.2 Greedy 法

Greedy 法も構築法の一つで、最も距離の短い辺を選んでいく方法である。閉路ができたり、次数が2を超えたりしないように辺を選ぶ必要がある。Nearest Neighbor 法と同様に、前半は効率よく辺を選択できるが、後半にコストの高い辺を選んでしまうという欠点がある。実装した Greedy 法の手順を以下に示す。

- i. 都市の距離が入った配列と都市の座標が入った配列を用意する。
- ii. それぞれの都市にグループ番号を設定する。
- iii. 次数が2を超えないように距離の短い辺を選択する。この際、辺の端点と同じグループの場合、選択しない。
- iv. 選択した辺と連続している都市に同じグループ番号を割り当てる。
- v. 全ての都市を通るまで iii, iv を繰り返す。
- vi. 経路ができるように都市のソートを行う。

Greedy 法の結果は、総距離が 339,319km となり、最適解との誤差は 15.27% となった。実行時間は 0.469 秒である。

3. 改善法

3.1 Or-opt 法

Or-opt 法は改善法の一つで、巡回路の一部を他の位置に挿入して巡回路の改善を試みる方法である。元の巡回路として 2.2 節の Greedy 法で作成した巡回路を用いた。実装した Or-opt 法の手順を以下に示す。

- i. 構築法によって得られた巡回路の都市の座標が入った配列の先頭を注目都市とする。
- ii. 注目都市を他の都市の間に入れた場合、経路が改善されるならば注目都市をその位置に挿入する。
- iii. 注目都市を次の都市にする。
- iv. 経路が改善できなくなるまで ii, iii を繰り返す。

Or-opt 法の結果は、総距離が 323,372km となり、最適解との誤差は 9.86% となった。実行時間は 0.531 秒である。

3.2 Or-opt 法の高速化

Or-opt 法は改善の余地がないような都市も探索することになり、無駄な計算が入ってしまう。そこで、ある都市とその2つ先までの都市の経路に対して閾値を用いて、ある都市に Or-opt 法を試みるかどうかの判定を行うアルゴリズムを提案する。具体的な手順を以下に示す。

- i. 構築法によって作られた巡回路の都市の座標が入った配列の先頭を注目都市とする
- ii. 注目都市とその次の都市の経路と、注目都市から1つ目と2つ目の都市の経路を足したものから、注目都市と2つ目の都市の経路を引いたものが閾値を超えていれば注目都市に対して Or-opt 法を適用する。
- iii. 注目都市を次の都市にする。
- iv. 経路が改善できなくなるまで ii, iii を繰り返す。

巡回路における2都市間の距離の平均値を半分ずつにしていった値をそれぞれ閾値として用いた。その結果を表1と表2に示す。

表1: Nearest Neighbor 法+Or-opt 法の高速化

閾値 (km)	総距離 (km)	最適解との差 (%)	時間 (秒)
534.45	351,632	19.46	0.079
267.22	342,449	16.34	0.186
133.61	338,836	15.11	0.410
66.81	339,994	15.50	0.502
33.40	334,459	13.62	0.606
16.70	334,472	13.63	0.702
8.35	330,830	12.39	0.815
0	328,791	11.70	0.909

表2: Greedy 法+Or-opt 法の高速化

閾値 (km)	総距離 (km)	最適解との差 (%)	時間 (秒)
509.49	330,178	12.17	0.094
254.74	330,804	12.38	0.133
127.37	329,885	12.07	0.348
63.69	324,482	10.23	0.362
31.84	323,345	9.85	0.406
15.92	320,644	8.93	0.507
7.96	323,038	9.74	0.486
0.00	323,372	9.86	0.531

4. まとめ

Or-opt 法の高速化は Greedy 法においては適切な閾値をとることができれば、速度向上だけではなく精度向上も確認することができた。

参考文献

- [1] ウィリアム・J・クック, (訳)松浦俊輔: 驚きの数学 巡回セールスマン問題, 2013
- [2] B.コルテ, J.フィーゲン, (訳)浅野孝夫, 浅野泰仁, 小野孝男, 平田富夫: 組み合わせ最適化 第2版 理論とアルゴリズム, 2012
- [3] Martin Grottschel and Olaf Holland, "Solution of large-scale symmetric traveling salesman problems", *Mathematical Programming*, Vol.51, pp.141-202, 1991