4K-01

カッコウ探索と2opt法のハイブリッド化による巡回セールスマン問題の解法

戸田敬太　熊谷洋佑　藤井昭宏　田中輝雄

工学院大学

1. はじめに

進化的計算であるカッコウ探索 (CS : Cuckoo Search)[2]は本来数値計算のアルゴリズムであるが，本研究では巡回セールスマン問題 (TSP : Traveling Salesman Program)[1]に適応した．

CS はカッコウの繁殖行動である托卵にレヴィフライトを組み合わせたアルゴリズムである．CS は世代が進むにつれ各解が似通った解になってしまう性質がある．

本研究では，解の精度の向上を目的とする．

そこで，従来の CS よりも精度の高い解を得るため，CS に局所探索法である 2opt 法[3]を組み合わせた．本研究では，初期生成段階のみに実行した CS-2opt と，世代毎に実行した CS-2opt-Every を提案する．

2. カッコウ探索

2.1 カッコウの繁殖行動

カッコウは托卵という特殊な繁殖行動を持っている．托卵は自分の卵を他の種の鳥に育てさせる繁殖行動である．もし宿主の鳥が自身のものではない卵を発見したら，宿主の鳥はその卵を捨てる．カッコウは托卵を見破られないようにするため，卵の色などを仮親の卵に似せている．

2.2 レヴィフライト

レヴィフライトは単純にランダムに移動するブラウン運動とは違い，ほとんどが規則性のない短距離の移動だが，時折長距離の移動をするランダムウォークである．この移動距離は指数が$-1 \sim -3$のべき乗分布に従う．

Cuckoo Search with 2-opt method for traveling salesman program

Keita Toda，Kumagai Yosuke，Fujii Akihiro，Tanaka Teruo

Kogakuin University

2.3 カッコウ探索

CS は上記のカッコウの繁殖行動とレヴィフライトを組み合わせたアルゴリズムである[2].

CS の基本的なステップは図 1 の擬似コードのように要約することができる．

```
1   begin
2     n を巣の数とし，巣に卵を 1 個ずつ生成 x_{i(i = 1,2,…,n)}^0
3     t (世代数) = 1
4     while (t < 最大世代数 または (その他基準)
5       レヴィフライトを利用し，ランダムに選んだ
          巣の卵からカッコウの卵 x_i^t を作る
6       ランダムに卵 x_j^{t-1} を選択
7       if (x_i^t が x_j^{t-1} に比べ改善)
8         x_i^t によって x_j^{t-1} を置き換える
9       end if
10      卵を並び替え，最良な卵を発見
11      一定数の悪い巣が放棄され，新しい巣を構築
12      t = t + 1
13    end while
14  end
```

図 1 カッコウ探索（CS）の擬似コード

3. 巡回セールスマン問題への適応

3.1 巡回セールスマン問題

TSP とはセールスマンが所定の複数の都市を 1 回だけ巡回する場合の最短経路を求める組合せ最適化問題である．この問題は都市の数が多くなるに従い，探索経路が指数関数的に増大し，NP 困難な問題として知られている[1].

3.2 2opt 法

2opt 法は TSP の代表的解法の 1 つである[3]. 巡回路の中から 2 都市間を結ぶ 2 つを選び，巡回路長が短くなるように入れ替える．これを入れ替えができなくなるまで繰り返す．大域探索を行うことはできず，得られる解の質は高くないが，その代わり高速なアルゴリズムである．

表 1 各解法での最良な解（最短距離からの増分率）

問題	都市数	最短距離	巡回路長										
			CS			CS-2opt			CS-2opt-Every			2opt	
			Best	Average	世代数	Best	Average	世代数	Best	Average	世代数	Best	Average
eil51	51	426	437 (2.6%)	454 (6.6%)	5×10^3	431 (1.2%)	434 (1.9%)	5×10^3	430 (0.9%)	430 (0.9%)	10^2	438 (2.8%)	445 (4.5%)
a280	280	2579	2811 (9.0%)	2867 (11.2%)	10^5	2644 (2.5%)	2685 (4.1%)	10^5	2591 (0.5%)	2620 (1.6%)	5×10^2	2750 (6.6%)	2799 (8.5%)
pr1002	1002	259045	280867 (8.4%)	288162 (11.2%)	10^6	276923 (6.9%)	278585 (7.5%)	10^6	265831 (2.6%)	267931 (3.4%)	5×10^4	279618 (7.9%)	282513 (9.1%)

3.3 CS の適応

CS の流れは図 2 の擬似コードに従う. 目的関数は巡回距離の最小化である．レヴィフライトを掛け合わせる過程は辺の入れ替えによって表現した. レヴィフライトでいう移動距離を, TSP の巡回路の変化の度合いと置き，2 辺の入れ替えは短い変化，入れ替えを行う辺を 3，4，5…と増やすごとに，変化量が増えるとした．今回の実装では最大 16 辺の入れ替えとした. レビィフライトのべき乗の指数は−2とする.

4. 提案手法

従来の CS よりも精度の高い解を得るため, CS に局所探索法である 2opt 法[3]を組み合わせた．初期生成段階のみに実行した CS-2opt と，世代毎に実行した CS-2opt-Every を提案する.

CS-2opt では擬似コード 2 行目の初期集団の生成の際に, CS-2opt-Every では図 2 の 7 行目で新しい解に置き換わった際に実行した.

CS-2opt-Every の 2opt 法は全ての生成解に実行するわけではないので，計算量が莫大にはならずに, 少ない世代数で優れた解を求められる.

5. 実験と評価

本研究は, TSPLIB[4]のベンチマークを用いて性能評価を行う．各問題の全ての都市は接続している．今回の実験では，一定の世代数で処理を打ち切った．1 つにつき 10 回計測を行った．問題は小規模〜中規模都市を使用する.

実験は CS と CS-2opt 及び CS-2opt-Every の精度比較を行う.

表 1 に TSP の問題と各結果を示す.

実行環境は CPU に Intel Xeon E5-2643 を使用した．コンパイラには gcc 4.1.2 を使用した.

CS-2opt は CS に比べ, Best が 1.4〜6.5 ポイント，Average が 3.7〜6.5 ポイントという改善ができた．Best と Average を比べてみると，CS は差があるが，CS-2opt では近い値が出ている．このことから，CS-2opt は安定して良い解が求められると考えられる.

CS-2opt-Every は CS に比べ，Best が 1.6〜8.5 ポイント，Average が 5.6〜9.6 ポイントという改善ができた．世代数が少ない理由は他 2 つに比べ，早い段階で収束が起こったからである.

6. おわりに

本研究では，TSP に対して従来の CS と提案手法である CS-2opt の精度を比較した.

CS-2opt において，初期生成段階で精度を上げることにより精度が上がった．また，毎世代 2opt を実行することにより，さらなる精度向上ができた．CS-2opt-Every は従来の CS に比べ，少ない世代数で最良値を求められた.

今後の課題として, 改良法に 2opt 法以外を適応，またレヴィフライトの移動距離の設定の反復毎の修正などがある．さらに，大域性をより上げるために遺伝的要素を絡めていきたい.

また都市数が増えると世代数が比例して増加するので，膨大な時間がかかるという問題もある. そのため世代数を減らす改良が必要である.

参考文献

[1] 山本 芳嗣，久保 幹雄，“巡回セールスマン問題への招待”，朝倉書店 (1997).

[2] Xin-She Yang, Suash Deb, “Cuckoo Search via Levy Flights”, Nature & Biologically Inspired Computing, no.37, pp. 210-214 (September 2009).

[3] G. A. Croes, “A Method for Solving Traveling-Salesman Problems”, Operations Research, Vol.6 No.6, pp. 791-812 (1958).

[4] TSPLIB http://elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib/tsplib.html