6L-05

最遠の粒子を参照する粒子群最適化を用いた巡回セールスマン問題の解法

山田 悠希† 穴田 一†
東京都市大学 大学院総合理工学研究科

1. はじめに

自然界には，アリのフェロモンコミュニケーションを用いた採餌行動に代表されるように，単体では単純な知能しか持たないにも関わらず，複数の個体からなる「群れ」となり，相互に作用し合う事で高度な動きを創発する生物が存在する．このような「群れ」が創発する知能の事を群知能と呼び，現在もなお活発に研究が行われている．このような群知能を応用したアルゴリズムに，鳥や魚などに見られる群れ行動をモデルにした粒子群最適化(Particle Swarm Optimization, PSO) [1]がある．PSO は解空間上に配置された複数の粒子が互いに情報共有を行いながら解空間の探索を行うアルゴリズムで，主に実数値最適化問題において評価されており，短時間で良い解に到達するという特徴がある．私達は，この特徴が実数値最適化だけでなく，工業や経済の問題の多くに適用される組み合わせ最適化にも活かせるのではないかと考え，本研究では組み合わせ最適化問題の一つである巡回セールスマン問題 (Traveling Salesman Problem, TSP) への適用を研究目的としている．

本提案手法は，PSO のアルゴリズムの特徴を経路の重ね合わせを用いて表現し，それに加えて各粒子が最も遠い粒子の情報を参照できるように構築したものとなっている．具体的には，PSO で用いられている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解の情報を現在の解に重ね合わせた解の集合を用いて，解の更新を行うよう設計した．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を比較することで，その有効性を確認した.

2. 粒子群最適化

粒子群最適化(Particle Swarm Optimization, PSO)とは，1995 年に Kennedy らが提案した，魚や鳥などに見られる群行動を探索手法に応用した，実数値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，全粒子の中で最も適応度の高い粒子の情報を全粒子で共有し，より良い位置に近づくように速度と位置を更新する．PSO はこの操作を繰り返すことで，解空間を探索するアルゴリズムである．t イテレーション目の粒子 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．

$$x_i(t) = x_i(t-1) + v_i(t-1) \quad (1)$$

$$v_i(t) = wv_i(t-1) + c_1r_1\big(pbest_i - x_i(t)\big) + c_2r_2\big(gbest - x_i(t)\big) \quad (2)$$

ここで w, c_1, c_2 は [0,1] のパラメータ，r_1, r_2 は [0,1] の一様乱数，$pbest_i$ は粒子 i のそれまでの最良解，$gbest$ は全粒子中の最良解である．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
全粒子の位置と速度をランダムに設定し，各粒子 i の最良解 $pbest_i$ を現在位置に設定する．次に，全粒子の中で適応度が最も高い解を $gbest$ と設定する.

②位置の更新
(1)式に従い，各粒子の位置の更新を行う.

③適応度の評価
全粒子の適応度の評価を行う．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$pbest, gbest$ の更新
全粒子の $pbest$ と $gbest$ を更新する.

⑤速度の更新
(2)式に従い，各粒子の速度の更新を行う.

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，事前に設定したイテレーション数繰り返すことで解空間を探索する.

また，粒子間の情報共有を制限する PSO[2]も提案されている．この PSO では，各粒子は近傍にいる粒子とのみ情報共有をすることが出来る．この情報共有を制限した PSO における t イテレーション目の位置 $x_i(t)$ と速度 $v_i(t)$ の更新式は以下のようになる.

An Algorithm for Traveling Salesman Problem using Particle Swarm Optimization with reference to the farthest particle
Yuki Yamada†, Hajime Anada†,
†Graduate School of Integrative Science and Engineering, Tokyo City University

$$x_i(t) = x_i(t-1) + v_i(t-1) \tag{3}$$

$$v_i(t) = wv_i(t-1)c_1r_1\big(pbest_i - x_i(t)\big) + c_2r_2\big(lbest_i - x_i(t)\big) \tag{4}$$

ここで w, c_1, c_2 は [0,1] のパラメータ，r_1, r_2 は [0,1] の一様乱数，$pbest_i$ は粒子 i のそれまでの最良解，$lbest_i$ は粒子 i の近傍内の最良解である．

3. 提案手法

PSO は実数値最適化手法であるため，TSP に適用させるためには様々な工夫が必要である．本提案手法では，粒子群最適化における解の更新を，複数の解を重ね合わせた経路集合を用いた経路構築という形で表現している．また，本提案手法は近傍を導入した粒子群最適化を基にしたアルゴリズムとなっている．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保持している．この解は，各粒子の現在の解，それまでの各粒子の最良解，近傍の粒子の最良解，最遠の粒子の解，以上四つの解で形成されている経路集合をイテレーションごとに更新し，更新した経路集合を用いて解の更新を行う．本提案手法はこの操作を繰り返すことで，解空間を探索するアルゴリズムとなっている．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 i に解 x_i をランダムに設定し，各粒子の最良解 $pbest_i$ 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．

$$d_{ij} = \frac{1}{S_{ij}} \tag{5}$$

$$S_{ij} = \frac{|E_i \cap E_j|}{n}$$

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 は都市数である．距離 d_{ij} は x_i と x_j の異なる経路が多くなるほど長くなる．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 $lbest_i$，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

t イテレーション目における粒子 i の解の更新は $pbest_i$，$lbest_i$ に加え，最遠の粒子の解である x_i^f を現在の解 x_i に重ね合わせ生成した経路集合 $G_i(t)$ を用いて行われる．$G_i(t)$の更新式は次式で表される．

$$G_i(t) = \alpha G_i(t-1) + \{x_i + c_1 pbest_i + c_2 lbest_i + c_3 x_i^f\} \tag{6}$$

ここで，α，$c_1 \sim c_3 (1 > c_1 > c_2 > c_3)$ はパラメータを表している．この式は，PSO の「複数の粒子の情報を元に，現在の速度を更新する」という特徴を表現している．この経路集合を用いて解の更新を行う事で PSO の解更新を表現している．解更新ではまず，スタートとなる都市 a をランダムに選択する．そして，次に移動する都市を確率によって決定する． t イテレーション目において，粒子 i が都市 a から都市 b に移動する経路を選択する確率$P_i^{ab}(t)$は次式で表される．

$$P_i^{ab}(t) = \frac{w_i^{ab}}{\sum_{l=1}^{n} w_i^{al}} \quad (l \in N') \tag{7}$$

$$w_i^{ab} = \frac{G_i^{ab}(t)}{(d_{ab})^D}$$

ここで，N' は未訪問都市，D はパラメータを表している．$G_i(t)$の重みが大きく，距離が短い経路ほど，選択しやすくなるように設定している．また，経路集合の中に選択できる経路が存在しない場合，未訪問都市の経路候補の中から距離の逆数を用いたルーレット選択を用いて経路を選択する．この操作を巡回路が構築されるまで繰り返す．

③総経路長の計算

全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

全粒子の $pbest$, $lbest$ と $gbest$ を更新する．

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，事前に設定したイテレーション数繰り返すことで TSP の解空間を探索する．

結果，考察は発表時に述べる．

参考文献

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

[2] Russ C Eberhart, James Kennedy, et al. A new optimizer using particle swarmtheory. In Proceedings of the sixth international symposium on micro machineand human science, Vol. 1, pp. 39-43, New York, NY (1995).