人工知能 中間レポート

電気情報工学科 3 年 C 課程 1TE20262G 石山 遼

1 概要

1.1 実装対象

NP 困難とされている巡回セールス問題 (TSP) に強化学習 (Q 学習) を応用し，生成された経路の分析や，パラメータを変更し特徴や経路長について述べる．

1.2 TSP への応用方法

Q 学習をベースとした更新式を用いて閉経路の経路長を最小とすることを目的として実装を行う．

TSP を解くにあたり，Q 学習を離散グラフで解釈する必要がある．状態 S_t についてはグラフのノード番号とし，次の状態 S_{t+1} においては，S_t に隣接し未だ到達していないノードの集合 N から選ばれるものとする．したがって，行動 a_t も集合 N のある状態へ移動することを指す．

1.2.1 報酬

報酬については，文献 [4] を参考にした．前提として，初期状態 S_{init} から探索を開始し，t において全ての状態を到達したとき，$S_{t+1} = S_{init}$ となるように行動を選択し，閉路を生成する．

上記を踏まえ価値関数は以下のように定義される．経路長はノード間のユークリッド距離としており，$length(a, b)$ は a-b 間の経路長を意味する．

$$r_t = \begin{cases} -length(S_t, S_{t+1}) & (S_{t+1} \neq S_{init}) \\ \frac{100}{length(S_t, S_{t+1})} & (S_{t+1} = S_{init}) \end{cases} \tag{1}$$

式 (1) では，集合 N が空でないとき，報酬 (r) は経路長 (S_t, S_{t+1}) の負の値をとる．集合 N が空であれば，$S_{t+1} = S_{init}$ となる．閉路を完成させるとき，経路長に反比例した正の報酬が与えられる．

1.2.2 行動選択

$\epsilon - greedy$ を採用し，ランダム変数を r(0 以上 1 未満の実数)，試行回数を $episode$ として以下のように行動選択される．

$$a_t = \begin{cases} random(N) & (r < \frac{\epsilon}{(episode+1)}) \\ \min_{s \in N} length(S_t, s) & (otherwise) \end{cases} \tag{2}$$

$r < \frac{\epsilon}{(episode+1)}$ という条件については，文献 [1] を参考にした．エピソード数が小さい時，ランダム選択が行われる確率が高く，後半のエピソードについては $greedy$ に選択されやすくなる．エージェントはエピソード前半においては，初期値 0 の Q テーブルを埋めるような行動選択をし，後半においては貪欲に隣接距離が近いものを選択していく．

Q 学習における更新式は式 (3) を用いており，$\max_{p \in N} Q(S_{t+1}, p)$ については経路の整合性を取るために集合 N に含まれている状態へ移動する Q 値最大の行動しか選択されない．

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) + \alpha[r_{t+1} + \gamma \max_{p \in N} Q(S_{t+1}, p) - Q(S_t, a_t)] \tag{3}$$

1.2.3 最終的な方策

TSP においては，全てのノードを通過し閉路となる必要がある．ここで，任意のエピソード数を終了した Q テーブルを用いて，目的の方策は定義される．Q テーブルを参照した以下の方策で最終的な閉路を生成する．

1. 初期状態 S_{init} から価値最大の状態 S へ移動
2. 状態 S に関する集合 N(隣接かつ未到達状態からなる集合) を定義
3. N が空集合ならば S_{init} へ移動し閉路完了
4. N が空集合でなければ S から価値最大の状態 $S_n \in N$ へ移動
5. S に S_n をセットし，Step.2 へ

2　実行結果

ソースコードについては github リポジトリ (https://github.com/ishiyeahman/RL-test/blob/main/RL_test.ipynb) を参照していただきたい．ノード数 35 個のグラフにおいて，300 エピソード実行したときの実行結果を図 1 および図 2 に示す．

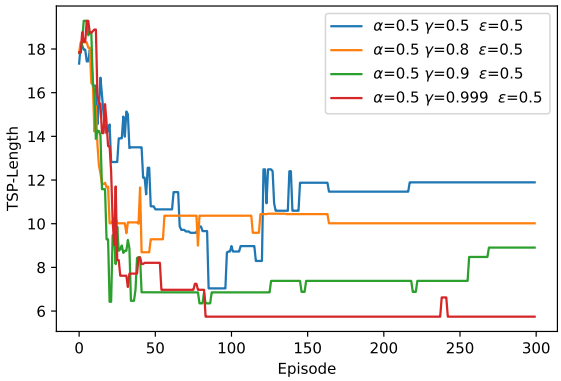

図 1: γ-パラメータ調整による経路長変化

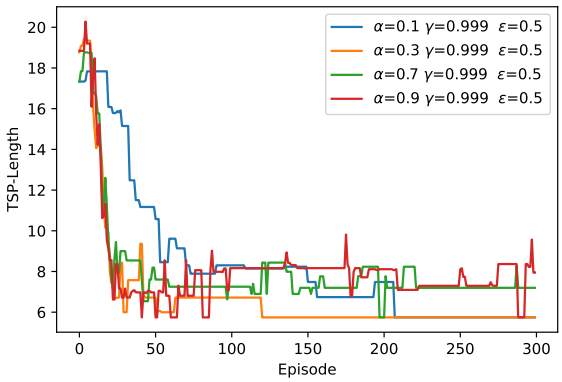

図 2: α-パラメータ調整による経路長変化

3　考察

割引率 γ については，$\gamma = 0.999$ で最良の経路が算出された．また，図 1 より割引率の大きさに伴って，収束値が変化していることができる．今回，閉路を生成するときに生じる正の報酬 (式 (1) を参照) をもとに Q 値が向上するため，長期的な報酬の影響を十分に評価するパラメータが最良となることと一致した．

学習率 α については，値が大きくなるほど経路長が不安定になっており，Q テーブルの値が都度変更されていることが読み取れる．したがって，$\alpha = 0.1, 0.3$ においては揺らぎの少ない学習過程が確認でき，最終的な経路も比較的短い．

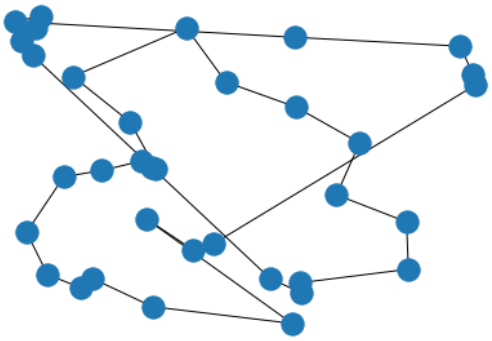

図 3: 生成された経路

図 1 実行時において生成された最良経路を図 3 に示す．明らかにエッジが交差しており，冗長な経路が収束後も存在していることが認められる．したがって，今回用いたパラメータ設定では，高い精度で最小コストの閉路を生成することは困難であると言える．これについては，参考文献 [4] でも同様な状況が見られた．Q 学習を用いた TSP への応用については改良点が多く存在すると言える．TSP への Q 学習応用は価値関数の設定が難しく，最適な報酬を与える方法十分に検討すべきである．

遺伝的アルゴリズム等であれば，部分的な最良経路を組み合わせる (交差) や突然変異を行う性質があるため，世代数および，個体数が多ければある程度妥協できる解を算出しやすい．しかしながら，今回の手法では，エピソード後半につれてランダムな選択が発生しないため，局所解が算出されやすいと考えられ，改善すべき点である．

最終的な方策の決定に関しても，初期状態から隣接したノードを連続的に選択し閉路を形成したものの，Q テーブルの学習状態を総合的に判断したような方策ではないため，改良の余地がある．本課題においては，Q 学習を TSP に応用したが，Q 学習のパラメータ調整は一般的に設定が難しく，グリッドサーチなどでチューニングされることが多い．しかし，上記のパラメータ設定はどの値もある程度収束しており，強化学習としての観点からは，その有用性が確認できるものとなった．今後は，ボルツマン選択を始めとした各種行動選択手法，パラメータチューニングを吟味することで妥当な解を算出できるよう引き続き取り組みたい．

参考文献・引用文献

[1] Python による AI プログラミング入門（O'REILLY，2019）

[2] Reinforcement learning for the traveling salesman problem with refueling
(URL:https://link.springer.com/content/pdf/10.1007/s40747-021-00444-4.pdf)

[3] Traveling salesman problem with reinforcement learning
(URL:https://medium.com/betacom/travelling-salesman-problem-with-reinforcement-learning-eac425be87aa)

[4] The traveling salesman and the Q-agent
(URL:https://github.com/rdgreene/sa_tsp/blob/master/The%20traveling%20salesman%20and%20the%20Q-agent.pdf)