3A3-4

マルチエージェント巡回清掃における自律的戦略の過学習とその一解消手法

Method of Solving the Overlearning in Autonomous Strategy Learning for Multi-agent Continuous Cleaning

杉山歩未　　菅原俊治
Ayumi Sugiyama　Toshiharu Sugawara

早稲田大学基幹理工学研究科情報理工・情報通信専攻
Department of Computer Science and Communications Engineering, Waseda University

In this paper, we propose the method to solve the overlearning in multi-agent continuous cleaning by discarding the learning result. We already proposed a strategy learning method so that agents individually identify the appropriate patrolling strategy in the coordinated cleaning tasks. However, we found that, in some environments, the performance of cleaning degraded because many agents overly select a certain strategy. This biased strategy selection, which we assume that is a kind of overlearning, results in another conflicts, and thus reduces the performance of the system by their interferences. Therefore, we propose the method to avoid this performance degradation by autonomously discarding the learning results. We experimentally examined our proposed method can avoid the overlearning by introducing it to agents and showed that it can keep the performance higher than the previous method.

1. 序論

近年，ロボット技術の発展により，生活の負担を減らすための介護，警備，清掃ロボットなど，ロボットが活躍する機会が増えてきている．例えば，清掃ではごみが時間とともに蓄積し，警備の領域では問題の早期発見のために，各場所をある頻度以上で巡回する必要がある．しかし，ロボットの移動速度や活動時間の制限から，作業範囲が広域になると複数台での協調作業が必要不可欠である．

ロボットによる清掃と警備の巡回の研究は多くある．例えば，[1] では，複数ロボットが自律的に環境を分割し，協力して全体をカバーする手法が提案されている．しかし，本研究と異なり，同一の領域内で複数ロボットが行動し，協調することは考慮していない．[2] では，各ロボットが環境の情報を取得しながら，自律的に行動計画を学習，決定することで，結果として各ロボット間で協調が行われている．しかし，これらのようなマルチエージェント巡回清掃において，各エージェントが自律的に探索戦略を学習する場合，多くのエージェントが同時に，特定の戦略を適切と判断し，システム全体で戦略が極端に偏る一種の過学習状態となり，その結果エージェント同士の競合が発生して，効率が低下することがある．本研究では，[2] の手法を拡張し，エージェントが自律的に過学習を解消する手法を提案する．

本稿の構成は以下の通りである．次節で [2] にもとづいてエージェントと環境をモデル化し，[2] で提案された戦略学習について説明する．次に予備実験として，学習とともに効率が低下する現象を述べる．第 4 節で提案する学習法を述べ，第 5 節で過学習による効率低下を防げることを示す．

2. 問題の定義

2.1 環境の定義

本研究で使用するモデルは [2] と同様である．以下簡単に説明する．エージェントの集合を $R = \{r_1, \ldots, r_n\}$ とする．離散時間を導入し，その単位は 1 ステップとする．1 ステップでエージェントは移動とごみの回収を行う．環境を有向グラフ $G = (V, E)$ で表わし，エージェントはこの環境 G 内を清掃する．$V = \{v_1, \ldots, v_x\}$ はノードの集合であり，エージェントやごみは頂点 v 上に存在する．E はエッジ e の集合であり，頂点 v_i と v_j をつなぐエッジを $e_{i,j}$ と表す．簡易化のためエッジの長さは全て 1 とする．

環境におけるごみの発生を確率的に表現する．各頂点 v の 1 ステップあたりのごみの発生確率を P_v とする．この発生確率の違いにより，ごみの偏りを表せる．エージェントはこの P_v を予め知っているとする．これを未知としてマップ作成などの手法によって獲得することも可能だが，本研究では探索戦略の学習に主眼を置くため，既に学習は終わったとして既知とした．時刻 t における頂点 v のごみの量を $L_t(v)$ とおくと，時刻 $t+1$ における頂点 v のごみの量 $L_{t+1}(v)$ は，

$$L_{t+1}(v) \leftarrow \begin{cases} L_t(v) + 1 & (\text{ゴミが発生した場合}) \\ L_t(v) & (\text{ゴミが発生しなかった場合}) \end{cases} \quad (1)$$

と蓄積する．ただし時刻 t においてエージェントが頂点 v を通過すると，ごみは回収され，$L_t(v) = 0$ となる．

エージェント r_i はバッテリをもち，その容量が 0 になる前に基地 v^i_{base} に戻り充電を開始する．このアルゴリズムの説明は割愛するが，詳細については [2] を参照されたい．

なお，本研究では，エージェント間での通信は行わず，他のエージェントの現在の戦略やごみの回収量など，内部の情報を知ることはできないとする．しかし，自分を含む全エージェントの現在位置は知ることができると仮定した．これは，環境のセンサー等とインフラでモニタリングでき，それをブロードキャストすることで容易に確認ができると想定したためである．この仮定により，エージェントは各頂点が最後に清掃された時間 t^v_{visit} が分かり，ごみの発生確率 P_v から，時刻 t における各頂点のごみの期待値 $EL_t(v)$ を

$$EL_t(v) = P_v(t - t^v_{visit} - 1) \quad (2)$$

と計算できる．

連絡先: 杉山歩未，a.sugiyama@isl.cs.waseda.ac.jp

2.2 エージェントの探索戦略

エージェント r_i は，次に進むべき目標頂点 v^i_{target} の選択と，そこへ至る経路を生成する．この生成された経路に従い，v^i_{target} に到着すると，再び新たな目標頂点と経路を生成する．以上の行動を繰り返すことで継続的な巡回清掃を行う．

2.2.1 目標選択戦略

エージェント r_i は，以下の 4 つの目標選択戦略と，以下で述べる学習により，適切な戦略を選ぶ．

1. ランダム法

   環境上のすべての頂点からランダムに 1 つの頂点を選ぶ．

2. 貪欲法

   各頂点のごみの期待値 $EL_t(v)$ のうち，上位 N_g 個の頂点から，ランダムに 1 つの頂点を選びそれを目標とする．ランダム性を加えるのは目標頂点を分散させるためである．

3. 斥力法

   他のエージェントから離れた頂点を目標とする．すべての頂点 v からランダムに N_{rep} 個の頂点を選び，その中で全てのエージェントから最も遠い頂点を目標とする．

4. 戦略的目標決定法

   近隣にごみがあると判断したときに，近隣を優先的に巡回する．このために，近隣の清掃を優先させるための閾値 $EV^i_{threshold}$ を学習する．エージェント r_i が現在の自分の位置との距離が d_{rad} 以下の頂点の集合を近領域 V^i_{area} と定義する．ここで d_{rad} は正の定数である．このとき，近領域内の各頂点のごみの量の期待値の平均を EV^i_t とする．この EV^i_t と，あらかじめ定義した閾値 $EV^i_{threshold}$ を比較し，$EV^i_t > EV^i_{threshold}$ の時は近領域内から新たに目標頂点を貪欲法で決定する．目標に移動し，清掃が終了した後，EV^i_t を再度求める．$EV^i_t < EV^i_{threshold}$ のときは，環境全体から目標頂点 v^i_{target} を貪欲法で決定する．その後 V^i_{area} を更新し，その V^i_{area} の評価値を EV^i_{t+1} とし，$EV^i_{threshold}$ を以下の学習式で更新する．

$$EV^i_{threshold} \leftarrow EV^i_{threshold} + \alpha(EV^i_{t+1} - EV^i_{threshold}) \tag{3}$$

   ここで $\alpha(0 < \alpha < 1)$ は割引率である．

2.2.2 サブゴール型経路計画法

経路生成は，現在地から目標頂点への最短経路をダイクストラ法や $A^\star$ アルゴリズム等で生成する．さらに効率化をさせるために，最短経路近隣のごみ存在量の期待値が高い頂点をサブゴールとして経由しつつ，目標頂点へ移動する経路計画法を採用する．これをサブゴール型経路計画法と呼ぶ．エージェントはこのサブゴール型経路計画法により，移動前に目標頂点までの経路を生成する．

2.2.3 学習型目標決定法

エージェントが自律的に，環境構造や他のエージェントの動きによって，2.2.1 で述べた目標選択戦略から適切だと思われるものを学習する手法であり，[2] の提案手法である．学習には強化学習を用いる．エージェント r_i は，戦略を決定しそれにしたがって清掃をするため，学習対象は選択する目標選択戦略であり，その目標選択戦略によって得られた報酬 u により，その行動価値 $Q^i(a)$ を以下の式で更新する．

$$Q^i(a) \leftarrow (1-\alpha)Q^i(a) + \alpha u \tag{4}$$

このステップを繰り返すことで最適な行動を学習できる．行動 a の戦略は上記のランダム法，貪欲法，斥力法，戦略型決定法のような，いくつかの目標選択戦略から選択するものとする．報酬 u は目標頂点までの経路で 1 ステップあたりに回収できたごみの量の平均値とする．具体的には，目標を決定した位置からその目標までの移動距離を d_{travel}，目標を決定した時刻から目標頂点に到達するまでの時刻の範囲を T_{travel} とすると，報酬 u は，

$$u = \frac{\sum_{t \in T_{travel}} EL_t(v^i_t)}{d_{travel}} \tag{5}$$

となる．ここで行動 a の目標選択戦略の選択は，ε-greedy 法によって行う．

2.3 評価指標

本研究では，清掃効率の評価指標として，ある期間 t_0 から t_e 内で環境中に存在する，ごみの存在時間の総和 D_{t_0,t_e} を用いる，これを

$$D_{t_0,t_e} = \sum_{v \in V} \sum_{t=t_0}^{t_e} L_t(v) \tag{6}$$

と定義する．D_{t_0,t_e} の値が小さいほど効率が良いと言える．

3. 予備実験

本論文では，まず，学習型目標決定法により過学習が発生する状況を再現する．

3.1 本研究における過学習

学習型目標決定法により，各エージェントが自律的に学習を行うと，ある環境で多くのエージェントが同じ目標選択戦略を適切と判断し，類似した行動をとるものが増えすぎる．その結果システム全体の効率が低下する．このように協調するグループが同じ学習を行ない，同じ戦略に偏る現象をエージェントグループの過学習と考える．

3.2 実験環境

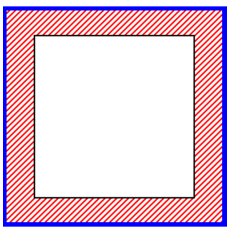

図 1: 環境 (a)

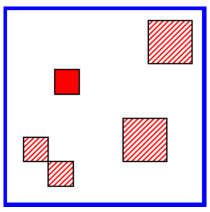

図 2: 環境 (b)

エージェントが清掃を行う仮想環境を，101×101 の 2 次元グリッドとする．各頂点は (x, y) と表し，それぞれ $(-50 \leq x, y \leq 50)$ とする．エージェントは充電基地 $v_{base} = (0, 0)$ からスタートする．

本実験では，図 1，図 2 に示すように，ごみの溜まりやすさに違いのある環境を想定した．環境 (a) は周囲にややごみが溜まりやすい環境である．環境 (b) は，ごみの溜まりやすい個所が何カ所かブロック状に存在する環境である．図中において，

表 1: 各目標決定法のパラメータ

目標決定法	パラメータ	値
貪欲法	N_g	5
斥力法	N_{rep}	100
戦略的目標決定法	α	0.1
	d_{rad}	15
学習型目標決定法	α	0.1
	ϵ	0.05

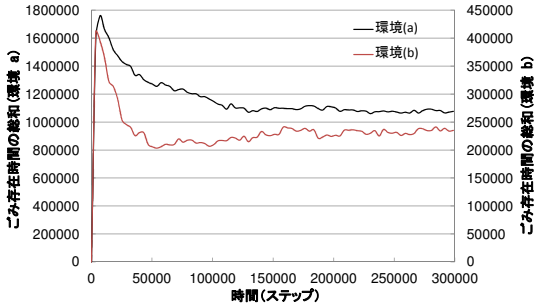

図 3: 学習型目標決定法（既存手法）の清掃効率推移

ごみの発生確率は以下のように設定した．

$$P_v = \begin{cases} 10^{-3}\ (\text{塗りつぶし部}) \\ 10^{-4}\ (\text{斜線部}) \\ 10^{-6}\ (\text{上記以外}) \end{cases} \tag{7}$$

実験は 1 回の試行を 300000 ステップとする．実験結果は 20 回の試行の平均値である．エージェント数は 20 で，全てのエージェントは前述した学習型目標決定法により目標選択戦略を決定する．各目標決定法のパラメータは表 1 に示す．本実験では，途中でエージェントを導入することはせずに，試行の最初に全エージェントを投入している．評価指標として式 (6) のごみの存在時間の総和を T ステップごとに記録している．本実験では $T = 3600$ とした．

3.3 実験結果と考察

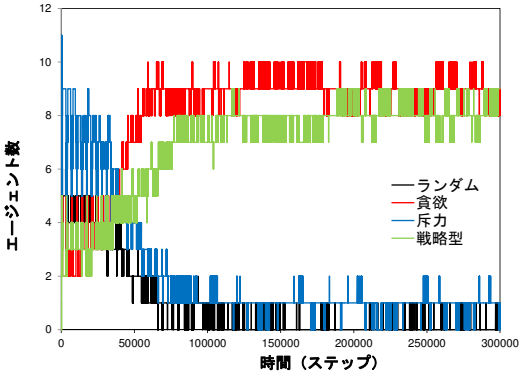

図 4: 各目標選択法を選択したエージェント数の推移（環境 b，既存手法）

表 2: 学習破棄手法（提案手法）のパラメータ

パラメータ	値
k_0	5
k_1	10
N_{th}	5

ごみ存在時間の総和の推移を図 3 に示す．環境 (a) では初期の状態から時間が経つにつれて効率が向上（ごみ存在時間の総和は減少）し，その後ほぼ一定に推移する．環境（b）では環境（a）と同様に途中まで効率は向上している．しかし，54000 ステップをピークに効率は低下し始め，最終的には 16%ほど効率は低下している．

この原因を調べるために，それぞれの環境で各時刻で選択されている目標選択戦略の台数の推移を図 4 に示す．戦略型目標決定法は貪欲法をベースとした戦略のため，ほぼ全てのエージェントが実質的に貪欲法を選択する目標選択戦略に偏っている．環境（b）は環境 (a) に比べ，ごみが多く発生する場所が連続せずに，ブロック状に点在している．そのため，貪欲法のような同じ場所に集まりやすい戦略を多くのエージェントが偏って選択し，同じブロックを多くのエージェントが目標とし，集まり過ぎることで，効率の低下を招いたと考えられる．このことから，ごみの発生しやすい場所が連続しておらず，その面積が小さいことと，同じような場所を目標とする戦略を多くのエージェントが選択することが過学習による効率低下を引き起こす要因となったと考えられる．そのため，過学習への対処には，戦略の偏り過ぎを防ぐか，多くのエージェントが選択しても同一の場所に集まり過ぎない目標選択戦略を用意することが必要である．ここでは，戦略の偏りを自律的に防ぐことで過学習に陥らないような手法を提案する．

4. 提案手法

本研究で提案する学習破棄手法は，学習型目標決定法を拡張したものであり，エージェントが自己の効率をモニタリングすることで，自律的に過学習状態を推定し，それまでの学習結果を一部破棄することで過学習状態を解消する手法である．

エージェントは T ステップごとに $D_{0,T}$，$D_{T+1,2T}$，$D_{2T+1,3T}$，$\cdots$ を記録し，過去 k_0T ステップと k_1T ステップで回収したごみの移動平均 $C^t_{k_0,T}$ と $C^t_{k_1,T}$ を求める．ここで，$k_0 < k_1$ は自然数であり，整数 k，n と時刻 $t(= nT)$ に対し，

$$C^t_{k,T} = \frac{D_{(n-k)T+1,(n-k+1)T} + \cdots + D_{(n-1)T+1,nT}}{k} \tag{8}$$

とする．このとき $C^t_{k_0,T} < C^t_{k_1,T}$ が続くと，下降傾向にあると考えられる．そこで閾値 $N_{th}(> 0)$ を導入し，$C^t_{k_0,T} < C^t_{k_1,T}$ が連続して N_{th} 回続いたとき，それまでの学習結果を破棄する．ここで，学習結果の破棄とは，全ての Q 値を初期化し，全ての $D_{(n-1)T+1,nT}$ を忘却する．

5. 評価実験

3.2 節と同様の実験環境で，予備実験（既存手法 [2]）の学習型目標決定法と提案手法との比較実験を行った．提案手法で導入した各パラメータの設定を表 2 に示す．

清掃効率の推移を比較した結果を図 5 に示す．過学習の影響がみられない環境 (a) では，既存手法と提案手法に差はほと

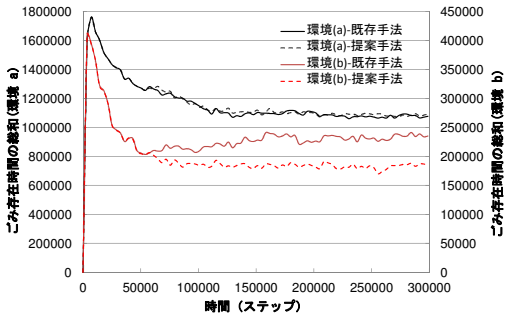

図 5: 既存手法と提案手法の清掃効率推移の比較

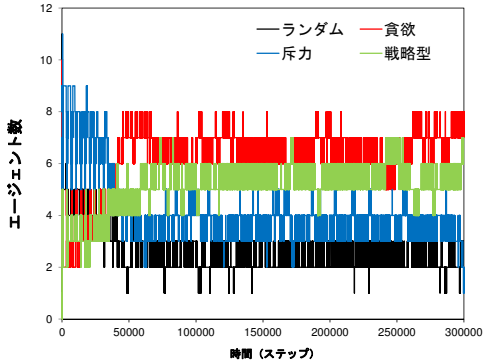

図 6: 各目標選択法を選択したエージェント数の推移（環境 b，提案手法）

んどない．過学習による効率低下が起きている環境 (b) では，既存手法で発生していた効率の低下は起こらず，最終的には提案手法が既存手法に比べ 21.5%ほど効率が向上した．

効率に変化のみられた環境 (b) において，提案手法で各目標選択戦略の選択された台数の推移を図 6 に示す．図 5 の結果から，適切な条件で学習結果を破棄することで，過学習が発生する環境において，効率を向上できている．また，過学習が発生しない環境においては，学習結果を破棄しても効率を大きく低下させないことも確認できた．図 6 からは，貪欲法と戦略型目標決定法が選択される台数は多いものの減少し，また，相対的にランダム法と斥力法を選択する台数が増えた．提案手法によって戦略の偏りが緩和できたことが確認できる．これらの結果から，ごみの回収量の推移を利用して過学習状態を推定し，状態に応じて学習結果を一部破棄することで，戦略の偏りを緩和し，過学習が解消されることで，競合を防ぎ，効率を向上されられることが分かった．

過学習を発生しない環境で学習を破棄しても効率が低下しないのは，破棄の条件を効率が低下傾向にあるとしたことが理由と考えられる．過学習が原因でなくとも，何らかの原因で効率が低下しているなら，学習結果を破棄して他の行動をとらせることで，ε-greedy 法のような現在の行動よりも最適な行動を探すという働きをしていると考えられる．過学習が発生しないときは効率にほぼ影響せず，発生するときは効率を向上できるということは，システムの導入段階では分からない予期せぬ過学習に，リスクなく対応できると考えられ，この観点からも本研究の提案手法は有効と考えられる．

6. 結論と今後の課題

本研究では，マルチエージェントシステムによる複数ロボットの巡回清掃において，過学習状態を解消する一手法を提案した．過学習状態をエージェントが自律的に解消する手段として，エージェントが自己の効率の悪化から，過学習状態を推定し，学習結果を一部破棄する学習破棄手法を提案した．評価実験の結果，提案手法によって過学習を解消し，清掃効率を向上させることが確認でき，提案手法の有効性を示した．

今回は通信範囲の制限や遅延，通信の不安定性を考慮し，エージェント間の通信ができない状態を想定していた．しかし，通信が行えるロボットは普及しており，通信によってより正確な情報を各エージェントが共有することで，エージェントはより適切な行動を取れる可能性もある．そのため，エージェント間の通信が可能なモデルの導入が考えられる．また，今回は環境はすでに既知で，変化がない場合を考えたが，実際には障害物の発生や，環境中のごみの発生確率が変わる場合もある．こういった動的なモデルへの対応も考えられる．

参考文献

[1] M. Ahmadi and P. Stone. A Multi-Robot System for Continuous Area Sweeping Tasks. In *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation.* pages 1724-1729.

[2] K.Yoneda and C.Kato and T.Sugawara. Autonomous Learning of Target Decision Strategies without Communications for Continuous Coordinated Cleaning Tasks. In *IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology* volume 2. 2013. pages 216-223.