複数ターゲットによる階層型モジュラー強化学習結果からの知識獲得

Hierarchical Modular Reinforcement Learning method in Multi-target Problem and Its Knowledge Acquisition of State-Action Rules

◯ [1] 伊賀上　大輔,　　[2] 市村　匠
◯ [1]Daisuke Igaue,　[2]Takumi Ichimura
[1] 県立広島大学大学院総合学術研究科経営情報学専攻
[1]Graduate School of Management and Information Systems,
Prefectural University of Hiroshima
[2] 県立広島大学経営情報学部
[2]Faculty of Management and Information Systems,
Prefectural University of Hiroshima

Abstract: Hierarchical Modular Reinforcement Learning(HMRL), consists of 2 layered learning where Profit-Sharing works to plan a target position in the higher layer and Q-learning trains the state-action pair to the target in the lower layer. In this paper, we expanded HMRL to multi-target problem under the consideration of the distance between target. We try to extract the knowledge related to state-action rules by C4.5. The state-action decision is implemented by using the acquired knowledge.

1　はじめに

階層型モジュラー強化学習 ($HMRL$)[1] は，エージェントの目標位置を策定する $Profit\text{-}Sharing$ の上位階層と，目標位置までの実際の行動を学習する Q 学習の下位階層の 2 層からなる学習モデルである．本論文では，$HMRL$ を複数ターゲットの追跡問題に拡張する．学習結果から $C4.5$ を用いて知識獲得を行った．獲得知識を用いて推論システムを構築し，下位階層の行動選択に用いることで，上位階層の学習の収束の加速を確認した．

2　追跡問題

追跡問題は，グリッド内に存在する複数のハンターエージェントが，ターゲットエージェントを協調して包囲し，捕獲することを目的とした強化学習問題のタスクである．エージェントの行動は「上下左右に 1 マス分移動する行動」と,「現在のグリッドに留まる」の 5 つの行動が選択できる．また，グリッドの外に出た行動は取れず，1 つのマスに複数のエージェントが存在することは出来ない．捕獲条件において，何体のエージェントで包囲するかで問題の複雑度は変化する．本論文では，4 体のエージェントでターゲットを捕獲する問題である 4-エージェント追跡問題によるシミュレーションを行う．4-エージェント追跡問題では，4 体のハンターエージェントが協調して捕獲しなければターゲットエージェントの捕獲が難しく，また膨大な状態空間になるために [2]，次元の呪いの回避や学習速度の低下の問題に対処する必要がある．また，捕獲にはグリッドの端である壁を用いることもできる．捕獲条件を満たした時点で捕獲成功となり，捕獲に貢献したエージェントに対して環境から報酬が与えられる．図 1 は 4-エージェント追跡問題の捕獲条件の例である．

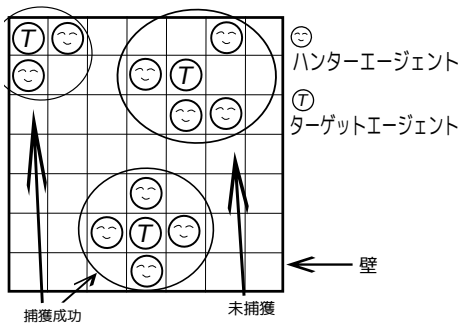

図 1: 7 × 7 のグリッドでの 4-エージェント追跡問題

従来研究の問題設定に加えて本論文では，2 体のターゲットエージェントをグリッド内に配置する．2 体の

捕獲時報酬は {0，100} で，ここで 0 の捕獲時報酬を持つターゲットを危険ターゲット，100 の捕獲時報酬を持つターゲットを安全ターゲットとする．報酬値の異なるターゲットが混在する場合，従来のように単純にターゲットに向かい捕獲するだけでは不十分で，報酬値が 0 である危険ターゲットの捕獲をしないために回避しつつ，報酬のより高い安全ターゲットを効率良く捕獲するような行動を選択しなければならないために，ハンターエージェントのタスクはより困難になる．

3 階層型モジュラー強化学習 [1]

階層型モジュラー強化学習は状態空間とタスクを分割することで，次元の呪いを回避し，学習性能を向上させている．従来手法において，上位階層はハンターエージェントごとに部分空間を分割し，4 つのモジュラーで全状態空間を表現した．今回，ターゲットが 2 体での追跡問題としたために，ハンターエージェントと，ターゲットエージェントのすべての組み合わせに状態空間を分割し，8 つのモジュラーで全状態空間を表現している．従来の状態表現を下式に示す．

$$(g, s_1, s_2, s_3, s_4) = \cup_e (e, g, s_e, s_\epsilon) \quad (e, \epsilon \in E, l \in L, e \neq \epsilon) \tag{1}$$

g がターゲットエージェントの位置，s がハンターエージェントの位置を示す．E はすべてのハンターエージェントの集合，L はすべてのターゲットエージェントの集合を示す．提案手法の状態表現を下式に示す．

$$(g_1, g_2, s_1, s_2, s_3, s_4) = \cup_e \cup_l (e, g_l, s_e, s_\epsilon) \quad (e, \epsilon \in E, l \in L, e \neq \epsilon) \tag{2}$$

3.1 モデル構造

階層型モジュラー強化学習では，上位階層の $Profit\text{-}Sharing$[3][4] で，各ハンターエージェントがどこに向かえばよいかのプランニングを行い，エージェントの目標位置の策定を決定する．下位階層の強化学習ではハンターエージェントの現在位置と上位階層で決定したハンターエージェントの目標位置の情報を元に Q 学習 [5] で行動選択を学習する．このように階層的に学習することで，目標達成のためのタスクが分割され問題の複雑さが軽減できる．また，それぞれ上位階層では行動を，下位階層では他のエージェントの状態を考慮しないことで，状態空間の次元数を削減できる．図 2 にモデル構造の例を示す．

学習の流れを図 3 に示す．上位階層と，下位階層の

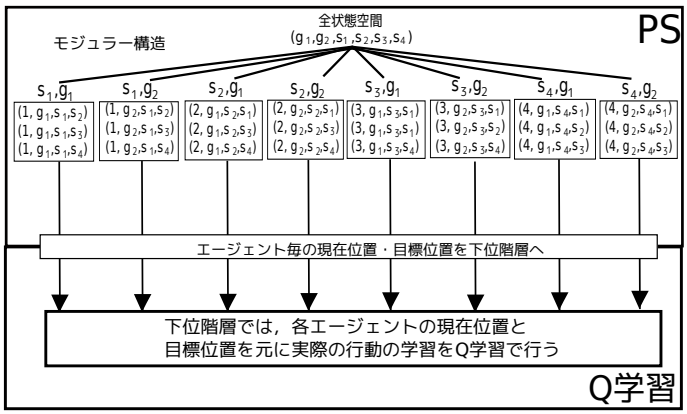

図 2: ターゲットが二体の時のモデル構造

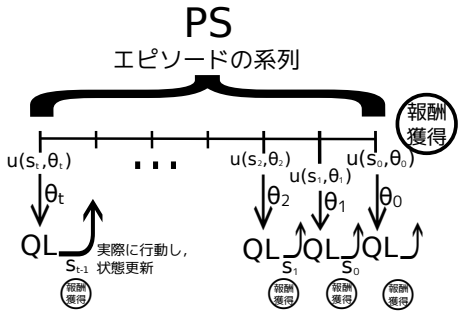

図 3: 階層構造のモデル

報酬獲得のタイミングは異なり，上位階層ではターゲットエージェントの捕獲に成功したときで，下位階層では，実際にハンターエージェントの行動選択を行い，上位階層から与えられた目標位置への遷移が成功したときに環境からの報酬が発生する．

3.2 ATField 関数

提案手法の上位階層学習では，報酬分配において，ターゲット同士の影響度を表現した ATField 関数を適用する．$ATF(gd)$ は ATField 関数を示し，当該ターゲットが他のターゲットから受ける影響度を求める．影響度は報酬分配時に忘却率にかかる．gd はターゲットエージェント間の距離である．

$$ATF = \begin{cases} \Phi = 0.0 & (if \quad gd \leq n_1) \\ \Phi = 1.0 & (if \quad n_1 < gd \leq n_2) \\ \Phi = 0.9 & (if \quad n_2 < gd) \end{cases} \tag{3}$$

n_1 は，近距離判定パラメタ，n_2 は遠距離判定パラメタとした．

3.3 上位階層による学習

上位階層は，モジュラー構造により分割された状態ごとに評価値を与え，評価値に従い目標位置をプランニングし，捕獲成功時に評価値を更新する．上位階層における評価値の更新式は，

$$\begin{aligned} u(e, g_l(i), h_e(i), h_\epsilon(i)) &= u(e, g_l(i), h_e(i), h_\epsilon(i)) \\ &+ k(e, g_l(i), h_e(i), h_\epsilon(i)) \end{aligned}$$

$$\begin{aligned} & k(e, g_l(i-1), h_e(i-1), h_\epsilon(i-1)) \\ &= \rho ATF(gd) k(e, g_l(i), h_e(i), h_\epsilon(i)) \\ & (i = 0, -1, \cdots, -m+1, \epsilon \neq e) \end{aligned} \tag{4}$$

である．$u(\cdot)$ は目標位置の評価値を求める関数を示し，$k(\cdot)$ は強化関数を示す．e は当該ハンターエージェント，ϵ は e 以外のエージェント，l はターゲットエージェント，$g_l(i)$ はターゲットエージェント l が報酬を得た時点を 0 としてステップ i 時点にいた位置，$h_e(i)$ はエージェント e がステップ i 時点にいた位置を示す．また下段の式は，評価値が強化される割合が報酬獲得ステップからさかのぼるごとに減衰することを示している．$\rho < 1$ は忘却係数である．$-m+1$ は更新対象のステップ数を示している．ハンターエージェントの目標位置は，

$$\begin{aligned} \theta_e = arg \max_v \sum_\epsilon \sum_l \frac{u(e, g_l, v, h_\epsilon)}{\mu^{|h_e - v|}} \\ (\epsilon \neq e, \mu \geq 1) \end{aligned} \tag{5}$$

で決定する．分母の μ は $\mu \geq 1$ の条件を持つパラメタで，当該エージェントと目標位置間の距離が指数乗されていることから，エージェントと目標位置間の距離が遠ければ遠いほど，評価値は低くなる．

3.4 下位階層による学習

下位階層では，上位階層で決定された目標位置へ移動するための具体的な行為を学習する．下位階層の Q 学習では，上位階層で決定された当該エージェントの目標位置と，当該エージェントの現在位置の情報を用いる．

$$\begin{aligned} & Q(s_e(t), a_e(t), \theta_e) = Q(s_e(t), a_e(t), \theta_e) + \\ & k(r(t) + \gamma \max_\eta Q(s_e(t+1), \eta, \theta_e) - Q(s_e(t), a_e(t), \theta_e)) \end{aligned} \tag{6}$$

Q は Q 値を示し，$s_e(t)$ は t 番目のステップのエージェント e の状態ベクトル，$a_e(t)$ はエージェント e の t 番目のステップで選択された行為，θ_e はエージェント e の目標位置である．また $r(t)$ はターゲットエージェント l に対する t 番目のステップの行動に対する報酬を表し，常時一定値を与えるものと設定する．k は学習のステップサイズパラメタ，γ は割引率である．更新式は現在の状態から次の状態に移ったとき，その Q 値を次の状態で最も Q 値の高い状態の値に近づけることを意味している．このことにより，報酬が更新ごとに伝播することになる．Q 学習においては，ターゲットエージェントや他のハンターエージェントの位置，ターゲットエージェント捕獲時の報酬などに依存せず，行動学習のみを行う．

4 知識獲得

学習後期の下位階層における学習データを元に教師データセットを生成し，$C4.5$ によって決定木を生成する [7]. 下位階層の学習から生成される教師データセットの前件部は，エージェントが知覚した入力信号から構成され，エージェントの現在位置から上位階層で設定された中間目標との差の X 成分と，Y 成分である．後件部は，エージェントが出力した行動 $(up, down, right, left, stay)$ である．また，下位階層の学習は，現在位置と目標位置の情報に対してとるべき行動がどのエージェントにおいても同じであるために，すべてのエージェントで一つの教師データセットを生成し，決定木を生成した．教師データは，十分に学習が進んでいる区間である 19,000 回から 20,000 回の区間について，下位階層の学習から教師データを抽出した．また，Q 学習の行動選択において探索的な行動選択であるランダム行動選択は教師データセットから除外している．

獲得知識を元に生成した If-Then ルールを読み込み，下位階層の行動選択を行う推論システムを適用する．一定の確率また，適合ルールが無い場合にはルールに依存せずランダムな行動選択を行う．

5 シミュレーション

シミュレーションは 7×7 のグリッドで，4-エージェント追跡問題を行う．シミュレーション開始時，ハンターエージェント，ターゲットエージェントはグリッド内にランダムに配置する．捕獲条件を満たし捕獲成功することを 1 試行の終了とする．1 試行が終了するとターゲットエージェントはランダムに再配置され，再度試行する．また，ターゲットエージェントごとに捕獲時の報酬は異なり，100 あるいは 0 の報酬が与えられる．

各ハンターエージェントは強化学習の行動選択に従って行動し，ターゲットエージェントはランダムな行動選択で，エージェントごとに順番に行動する．20,000回の試行を反復する間の学習を調査する．

5.1 シミュレーション結果

結果はそれぞれ10回シミュレーションを行い，平均を示している．また，表1の括弧内の数値はATF関数を適用しない場合の結果を示している．

表 1: 下位階層:Q学習　$n_1 = 2.0, n_2 = 5.0$

	安全ターゲットの捕獲確率	エピソード数	行動数
試行極初期 1-200	53.7% (53.1%)	703.5 (684.7)	999.1 (958.6)
試行初期 201-2000	73.4% (74.8%)	403.7 (355.2)	403.7 (358.5)
試行中期 2001-17000	89.7% (86.4%)	119.7 (101.3)	119.7 (104.9)
試行後期 17001-20000	90.6% (84.7%)	74.5 (62.3)	77.7 (66.6)

推論システムは，生成された49個のIf-Thenルールから構成した．

表 2: 下位階層:推論システム　$n_1 = 2.0, n_2 = 5.0$

	安全ターゲットの捕獲確率	エピソード数	行動数
試行極初期 1-200	56.5%	543.1	543.4
試行初期 201-2000	85.7%	194.0	195.6
試行中期 2001-17000	92.2%	63.0	69.3
試行後期 17001-20000	91.4%	46.3	54.6

5.2 考察

本論文で提案した$ATField$関数を報酬分配に適用した場合の$HMRL$が，適用しない場合の$HMRL$よりも安全ターゲットの捕獲確率が高い結果となった．しかし，エピソード数と行動数は若干増加した．また，推論システムを下位階層に適用した場合では上位階層から与えられた目標位置への行動を学習序盤から正確に出力したため，下位階層にQ学習を用いた時よりも推論システムを用いた方が，上位階層の学習の収束が加速した．

6 おわりに

本論文では，$HMRL$を複数ターゲットの追跡問題に拡張し，シミュレーションを行った．また，下位階層の強化学習結果から知識獲得を行い，推論システムを構築した．推論システムを用いることで，安全ターゲットの捕獲確率を低下させることなく，上位階層学習の収束を加速させることができた．今後は推論システムを用いて，*Android*アプリケーションなどの実システムへ適用したい．

参考文献

[1] 渡邊俊彦，和田竜也,「マルチエージェント追跡問題のための相対座標系に基づく階層型モジュラー強化学習」，バイオメディカル・ファジィ・システム，12(2)，pp.65-74，2010.

[2] 伊藤昭，金渕満,「知覚情報の粗視化によるマルチエージェント強化学習の高速化―ハンターゲームを例に―」，電子情報通信学会論文誌，Vol.J84-D1，No.3，pp.285-293，2001

[3] 宮崎和光，木村元，小林重信,「ProfitSharingに基づく強化学習の理論と応用」，人工知能学会誌，Vol.14，No5，pp.800-807，1999.

[4] 宮崎和光，荒井幸代，小林重信,「ProfitSharingを用いたマルチエージェント強化学習における報酬配分の理論的考察」，人工知能学会誌，Vol.14，No6，pp.1156-1164，1999.

[5] C.J.Watkins, and P.Dayan, “ Technical note:Q-Learning”, Machine Learning, Vol8，pp.58-68，1992.

[6] R.S.Sutton and A.G.Barto “ Reinforcement Learning”, MIT Press, 1998.

[7] D.Chapman and L.P.Kaelbling, :Input generalization in delayed reinforcement learning: An algorithm and performance comparisons,IJCAI-91,Vol.2,pp.726-731,1991

連絡先

〒 734-8558
広島市南区宇品東一丁目 1-71
県立広島大学 経営情報学部
市村　匠
E-mail: ichimura@pu-hiroshima.ac.jp