LG-2　マルチエージェント強化学習における記憶に基づく貢献度判別
Memory Based Contributions Deciding in Multi-Agent Reinforcement Learning

保知 良暢† 大囿 忠親†† 新谷 虎松††
Yoshinobu BOCHI Tadachika OZONO Toramatsu SHINTANI
†名古屋工業大学大学院 工学研究科
Graduate School of Engineering, Nagoya Institute of Technology
††名古屋工業大学 知能情報システム学科
Dept. of Intelligence and Computer Science, Nagoya Institute of Technology

1 はじめに

マルチエージェントシステムに関する研究は盛んに行われており、実世界においても応用されている。しかしシステムを構成する1つ1つのエージェントについて実装を行なうことは大きな負担となる。また自由度の大きなシステムや環境に未知の要素がある場合には最適に実装することは困難である。これらを解決する方法として、強化学習が注目されている [Sutton 98]。個々のエージェントが自律的に強化学習をすることでシステムをボトムアップ的に構築することを目指し多くの研究がなされている。個々のエージェントが自律的に強化学習をする場合、報酬がエージェント群全体に対して与えられるならば各エージェントに対して報酬を分配する機構が必要となる。どのエージェントにどれだけの量を分配すべきか、という問題を報酬分配問題という。既存の報酬分配問題のアプローチとして、従来研究には強化学習に対してゲーム理論を取り入れた手法 [Lauer 00] や、間接報酬の概念を導入したものがある [Miyazaki 01]。しかしゲーム理論を用いた方法は、個々のエージェントの行動や状態に対して報酬を与えることが困難な状況では、利得表を用意することは困難である。また間接報酬を導入した方法では、報酬発生に対する貢献の度合いに応じて間接報酬を決定することが重要である。不適切な貢献度判別を行うと、実際には報酬発生に大きな貢献をしたエージェントでも小量の報酬を獲得するならば、協調的な行動を創発させることは困難である。

本論文では、報酬分配問題に注目し、記憶に基づいた貢献度判別手法を提案する。この手法においてエージェントに新たな知識等を追加する必要はなく、過去に経験した状態、行動、報酬を利用する。これにより個々のエージェントの状態と行動に応じた貢献度判別が可能となる。また実験により、提案手法では最適政策に収束することを示す。

2 記憶に基づく貢献度判別手法

2.1 マルコフ決定過程

マルチエージェントの世界を以下の組で定義する。

$$\langle n, S, A^1, A^2, \cdots, A^n, P \rangle$$

n はエージェントの数、S は世界の状態の有限集合、$A^i(i = 1, 2, \cdots, n)$ はエージェント i の行動の有限集合、P は状態遷移と報酬発生を表す関数である。各離散時間ステップ $t = 0, 1, 2, \cdots$ においてエージェント i は状態 s_t を観測し、行動 a_t^i を実行する。そして状態は s_{t+1} に遷移し、報酬 r_{t+1} が発生する。ここで状態遷移にはマルコフ性を仮定し、関数 P は以下で表される。

$$P(s_t, a_t^0, a_t^1, \cdots, a_t^n, s') = Pr(s_{t+1} = s', r_{t+1} = r | s, a_t^1, a_t^2, \cdots, a_t^n)$$

$Pr(s, r|s, a_t^1, a_t^2, \cdots, a_t^n)$ は、状態 s でそれぞれのエージェントが行動 $a_t^1, a_t^2, \cdots, a_t^n$ を実行したときに、状態が s' へ遷移し報酬 $r(\in \Re)$ が発生する確率を表す。

ContribusionsDeciding(Now_event,$Past\_events$)
　Now_event：事象 $\langle s_t, a_t^1, a_t^2, \cdots, a_t^n, s_{t+1}, r_{t+1} \rangle$
　$Past\_events$：過去の事象の集合
・$Num_i = 0 (i = 1, 2, \cdots, n)$
・$Past\_events$ に Now_event を追加
・$Past\_events$ 内の事象 $Event$ 全てにおいて繰り返し
　・エージェント $i (i = 1, 2, \cdots, n)$ 全てにおいて繰り返し
　　・$Event$ と Now_event とにおいて
　　　$s_t, a_t^i, s_{t+1}, r_{t+1}$ が同じ場合
　　・$Num_i = Num_i + 1$
・return $\langle Num_1, Num_2, \cdots, Num_n \rangle$

図 1: 貢献度判別アルゴリズム

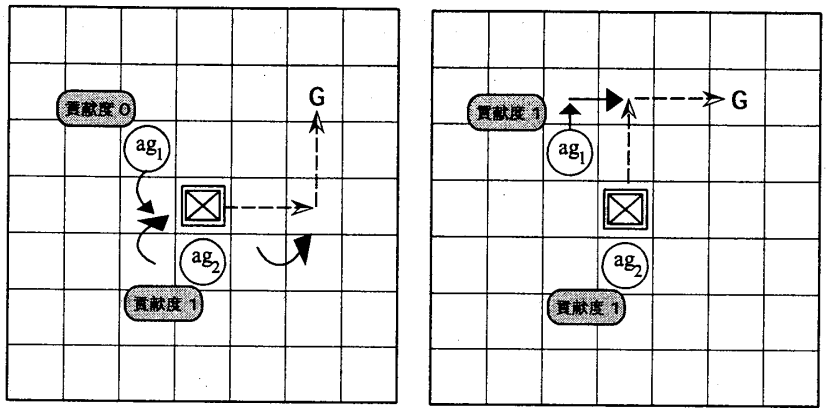
2.2 記憶に基づく貢献度判別と報酬分配手法

本論文では報酬発生時に記憶に基づき全エージェントの貢献度を判別するアルゴリズムを提案する。本提案手法では過去の事象を記憶可能であることを仮定する。1つの事象を以下で表す。

$$\langle s_t, a_t^1, a_t^2, \cdots, a_t^n, s_{t+1}, r_{t+1} \rangle$$

各離散時間ステップで1事象を経験する。このとき、学習が進むにつれて報酬発生に必要なエージェントや行動の情報が得られると考えられる。つまり、報酬発生に対して大きな貢献をしたエージェントや行動は過去の事象に多く現れる。提案手法では、各エージェントにおいて現在と同じ行動と状態遷移を行った過去の事象の数をそのエージェントの貢献度とする。アルゴリズムを図1に示す。提案するアルゴリズムは1つの事象と過去の事象集合を入力として受け取り、各エージェントの貢献度を出力する。

連絡先：保知良暢，名古屋工業大学知能情報システム学科新谷研究室，〒466-8555 名古屋市昭和区御器所町，TEL:052-744-3153, FAX:052-735-5477, bochi@ics.nitech.ac.jp



貢献度判別なし　　提案手法による貢献度判別

図 2: 倉庫管理シミュレーションの例

貢献度判別後、各エージェント毎に報酬を分配する。その手法を以下に示す。図 1 のアルゴリズムにより各エージェントの貢献度 $\langle Num_1, Num_2, \cdots, Num_n \rangle$ が得られる。このうち最も大きな値を持つ Num_i を Num_{max} とする。時刻 t におけるエージェント i の報酬分配量 r_t^i は以下とする。

$$r_t^i = r_{t+1} \cdot \frac{Num_i}{Num_{max}}$$

$$\text{where} \quad Num_{max} = \max_{1 \le i \le n} Num_i$$

そして各エージェントは上式によって与えられた報酬に基づき強化学習アルゴリズムを用いて学習する。

3 実験

倉庫管理シミュレーションにおいて実験を行う。2 次元世界上に強化学習を行うエージェント ag_1, ag_2 と荷物が存在する。エージェントは上下左右に移動でき、また荷物を押すことができる。各離散時間ステップにおいてエージェントは同時に行動を行うものとする。ここで荷物を目的位置 G まで運べば 0 以上の報酬が発生する。この問題において、1 エージェントのみで荷物を目的位置まで運ぶのに $L(L > 0)$ ステップ必要としたとき、複数エージェントとなったときに $M(M < L)$ ステップで運べるとき協調と呼ぶ。

エージェントと荷物が図 2 に示す位置関係にある場合を例に、貢献度判別無しと提案手法の結果を示す。実線の矢印はエージェントの行動を、点線の矢印は荷物の移動を表す。貢献度判別がない学習とは、実際に荷物を目的位置まで運んだエージェントにのみ報酬を与え、強化学習をさせる方法である。この場合 ag_2 が矢印の方向へ移動し、荷物を実際に目的位置まで運んだ。また ag_1 は下へ行動した。この結果 ag_0 から荷物を奪い取るような行動となった。これらの行動により、エージェント数が増えることで互いに競争的な立場となり、目標達成に必要なステップ数が増加した。

提案手法により貢献度を判別した結果、4 ステップで目標が達成さた。この動作が最も過去の事象において、頻繁に出現するようになった。実験では両エージェントの貢献度が共に 1 と判別され、この結果図 2 右に示す行動に収束した。つまりエージェント間は協調するようになり、目標達成に必要なステップ数が短縮された。

4 おわりに

本論文では記憶に基づいた貢献度判別手法を提案した。提案手法により、単純な計算にも関わらず最適な行動に収束することが確認できた。協調的な行動を創発するにはエージェント間での通信や交渉などが挙げられるが、状態爆発を引き起こす原因となることが報告されている [Tan 93]。実験で示したように、貢献度判別によりエージェント間の立場が操作が可能である。また提案手法により協調的な行動の創発が確認できた。しかしこの場合、実際に行動を行うエージェント以外に、これらを統括する機構が存在することとなる。またこの機構は全エージェントの状態を知る必要がある。

提案手法では過去の事象を記憶し、それらに基づいた報酬分配を行うため、学習時間に比例して計算量とメモリ量が増加する。しかしこれらを犠牲にしても最適な行動が学習できる可能性が上がることは重要であると考えられる。また、計算量とメモリ量の増加を実装上の工夫により抑制可能である。提案手法では学習困難な状況も考えられる。例えば状態遷移確率が学習中に変化したときには提案手法では即時に対応できない。これに対しては、過去の全ての事象を記憶するのでなく、過去数事象に限定して記憶するなどの工夫が必要である。本論文では実験により提案手法が最適な行動に収束したことを確認したが、全ての環境に対して保証しない。よって提案手法の理論的な考察が今後の課題である。

参考文献

[Sutton 98] Sutton, R. S. and Barto, A. G.: Reinforcement Learning—An Introduction—, The MIT Press (1998).

[Lauer 00] Lauer, M. and Riedmiller, M.: An Algorithm for Distributed Reinforcement Learning in Cooperative Multi-Agent Systems, *Proc. of the 17th International Conference on Machine Learning*, pp. 535–542 (2000).

[Miyazaki 01] Miyazaki, K. and Kobayashi, S.: Rationality of Reward Sharing in Multi-agent Reinforcement Learning, *New Generation Computing*, Vol. 91, No. 2, pp. 157–172 (2001).

[Tan 93] Tan, M.: Multi-Agent Reinforcement Learning:Independent vs. Cooperative Agents, *Proc. of the 10th International Conference on Machine Learning*, pp. 330–337 (1993).