3F4-05

非マルコフ性を有する環境のモデリング

A Modeling for Non-Markov Decision Process Environment

金澤勇樹　　栗原正仁
Yuki Kanazawa　Masahito Kurihara

北海道大学大学院工学研究科
Graduate School of Engineering, Hokkaido University

In this paper, we propose a learning system which combines the profit sharing with macro-rules coping with the non-markov probrem in a simple and effective way. A macro-rule is a rule sequence that combines the rules applied continuously, and it is used in order to reprensent the rule sequence acquired by agent's experience. By dynamic construction and use of macro-rules, the proposed system aims to improve the profit sharing system by adapting it to non-markov environment. By experiments, we show that the proposed system can be effective for some problems with hidden state.

1. はじめに

実ロボット制御問題やマルチエージェント問題において自律的に環境に適応する学習方式として，強化学習（Reinforcement Learning）が知られている。強化学習とは，報酬という外界からの入力を手がかりに学習を行う機械学習の一種であり，さまざまな強化学習法が提案されている。それらの中でも利益共有法（Profit Sharing）は非マルコフ性を有するような問題に対しても対処可能であると考えられているが，学習が困難である状況もまた想定される。

そこで本研究では，非マルコフ性を有する問題に対してできるだけ簡便かつ有効に対処する手法として，利益共有法とマクロルールを組み合わせた学習システムの提案を行う。マクロルールは連続して適用されたルールを結合したものであり，経験によって得られたルール系列を表現するために用いる。提案システムではマクロルールを学習の進行にあわせて動的に生成し，それらを順次利用することにより適応能力の改善を図る。

本研究では，非マルコフ性を有する問題の一例として，隠れ状態がある迷路問題を取り扱う。隠れ状態がある問題とは，環境側では異なる状態であるはずがエージェント側では同一の状態であると知覚してしまうといった問題である。このような問題に対して，提案システムを適用した実験を行い，その有効性について検証を行う。

2. 準備

2.1 利益共有法

ある状態と，その状態において実行可能な行動の対はルールとして記述される。状態 x で行動 a を選択する "if x then a" というルールを $R(x,a)$ と書く。また，初期状態あるいは報酬を得た直後の状態から次の報酬までのルール系列をエピソードという。

利益共有法ではエピソード単位で学習を行う。報酬が得られた時点で，エピソード中のルールは以下の式にしたがって強化される。

$$w(x_i,a_i) \leftarrow w(x_i,a_i) + f(r,i) \qquad (1)$$

連絡先: 金澤勇樹，北海道大学大学院工学研究科，
yukika@main.eng.hokudai.ac.jp
栗原正仁，同上，
kurihara@main.eng.hokudai.ac.jp

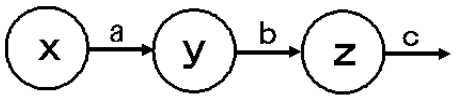

図 1: マクロルール

ここで $w(x_i,a_i)$ はエピソード中の i 番目のルールの重み，r は得られた報酬値，f は強化関数であり，一般に等比減少関数が用いられる。

2.2 マクロルール

マクロルールは，エージェントによって連続して適用されたルール系列を表現するものとして扱う。これを用いることによって，エージェントの経験によって得られたルール系列を環境の部分的な特徴として保持することが可能になる。また，連続したルール系列を一括して保持することで，単一のルールとルール系列を同等なものとして扱うことができると考えられる。

本研究では，マクロルールを以下のように定義する。

$$M_k = R_1 R_2 \cdots R_{n_k} \quad (n_k \geq 2) \qquad (2)$$

$$M = \{M_k\} \qquad (3)$$

ここで，M_k は k 個目のマクロルール，R はルール，n_k は個々のマクロルールの構成要素であるルールの数，M はマクロルールの集合を意味している。図 1 は $R(x,a)R(y,b)R(z,c)$ なるマクロルールを表している。

また，マクロルールには重みが付加される。個々のマクロルールの重みを $w(M_k)$ と記述することにする。マクロルールにおける重みは単一のルールの重みと同様に，エージェントによる行動決定時の選択基準となり，重みの更新もまた行われる。

マクロルールはエージェントの探索によって得られたルール系列から生成される。そのため，獲得したルール系列からどのようにして環境の特徴を抽出し，マクロルールとするかが問題となる。また，マクロルールはエージェントの探索の結果得られた決定的な状態遷移を表現していると考えられるため，エージェントに対して行動決定の際の決定的指針として用いることができる。

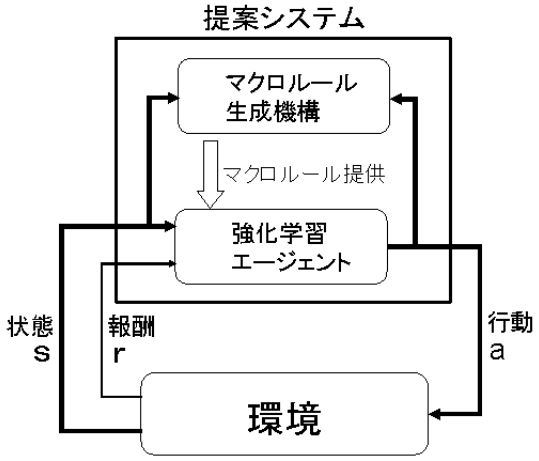

図 2: 提案システム

3. 提案学習システム

3.1 設計方針

本研究で提案する学習システムの概念図を図 2 に示す。

提案学習システムでは，行動決定機構として従来の強化学習法の枠組みがあり，それに付随する形でマクロルールを生成する枠組みが備え付けられている。マクロルールは学習の進行にあわせて逐次生成され，生成されたマクロルールはエージェントの内部状態表現として反映されることになる。具体的には，エージェントの探索によって蓄積されたエピソードから，複数共通するルール系列をマクロルールとして抽出し，環境の決定的な遷移構造表現として利用する。

マクロルールを用いることで，即時的な知覚情報だけではなく，過去の経験から抽出された環境の情報もまた同時に利用することができると考えられる。

3.2 学習システムの動作

提案学習システムの強化学習部分は，利益共有法を用いる。エージェントは各ルールの重みに準じたルーレット選択によって行動を決定する。ここで選択したルールと，先頭のルールが同じマクロルールが存在する場合，選択ルール及び存在するマクロルールの重みに準じたルーレット選択を再度行うことにより，マクロルールを適用するかどうかの決定を行う。マクロルールを適用する際には，構成要素であるルールを先頭ルールから順に選択することにする。

報酬が得られた時点で，各ルール及び各マクロルールの重みを更新する。単一ルールの重みに関しては式 (1) を用いて更新を行う。また、初期状態（あるいは以前報酬を得た直後の状態）から報酬を得るまでにマクロルールが利用されていれば，各マクロルール重みの更新を行う。

マクロルール重みの更新方法は、さまざまな形式が考えられる。例として以下の式を用いると，マクロルール重みは単一ルールの重みと同様に更新を行うことになる。

$$w(M_k) \leftarrow w(M_k) + f(r, i) \tag{4}$$

また，以下の式を用いると，マクロルールの構成要素であるルールの数によって更新の度合いを変化させることができると考えられる。

$$w(M_k) \leftarrow w(M_k) + \sum_{j=i}^{n_k} f(r, j) \tag{5}$$

3.3 マクロルールの適用方法

提案システムにおいて，マクロルールは学習の進行にあわせて動的に生成，拡張，削除といった操作が行われる。

マクロルールの生成

マクロルールは以下の手続きによって生成される。

Step1: 探索中に発生したエピソードを蓄積する。

Step2: 一定数 E 個のエピソードが蓄積された時点で，蓄積エピソード中においてルール系列 $R_1 \cdots R_n (n \geq 2)$ が N 個以上存在すれば，そのルール系列をマクロルールとして生成する。

生成された各マクロルールの重みの初期値は、

- 先頭のルールの重み
- 構成要素であるルール重みの相加平均
- 構成要素であるルール重みの相乗平均

などが考えられる。本研究では，場合に応じて使い分けることを許すことにする。

マクロルールの結合

あるエピソードにおいて，マクロルールが行動決定時に選択されていた場合，選択されたマクロルールと部分的に共通しており，かつ前後のルール系列を含むマクロルールが存在すれば，それらを結合して新たにマクロルールを生成する。具体的な手続きを以下に示す。

Step1: $R_1 \cdots R_n$ を選択されたマクロルールとする。ここで，$R_{p1} \cdots R_{pn}$（$pn \geq 1$）を選択されたマクロルール直前までのルール系列，$R_{s1} \cdots R_{sn}$（$sn \geq 1$）を選択されたマクロルール直後からのルール系列とする。

Step2: このとき，$R_{pi} \cdots R_j (1 \leq pi \leq pn, 1 \leq j \leq n)$ あるいは，$R_j \cdots R_{sk} (1 \leq j \leq n, 1 \leq sk \leq sn)$ なるマクロルールが存在していれば，それらを結合し，それぞれ $R_{pi} \cdots R_j$，$R_j \cdots R_{sk}$ というマクロルールを生成する。

マクロルールの削除

マクロルールによって行動選択を行う際，そのルール系列から外れるようなことがあれば，そのマクロルールを削除する。ここでルール系列から外れるとは，構成要素であるルールを選択した結果，次のルールの状態と異なる状態が知覚された場合や，ルールが選択できない場合を意味している。

4. 実験

実験では，文献 [McCallum 95] を参考にして，隠れ状態がある迷路問題を対象とした。

4.1 実験設定

図 3 の迷路環境を実験環境とした。エージェントはランダムに 4 箇所の角から出発し，ゴール G にたどり着いた時点で報酬を得る。報酬を得た場合を成功とする。エージェントは上下左右の壁の有無を状態として知覚する。これにより，環境側では異なる状態がエージェント側では同一の状態であるとみなしてしまうことになる。図 3 のマスの数は，左壁から左回りにそれぞれ 1，2，4，8 と数値付けした場合の和を意味しており，同じ数のマスが同一の状態であると認識されることになる。

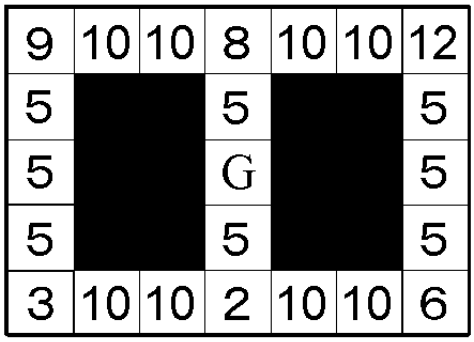

図 3: 実験環境

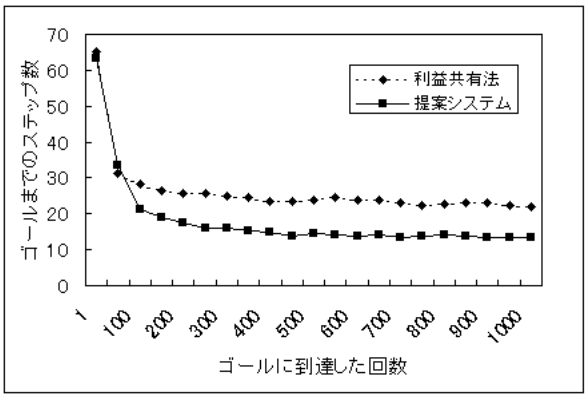

図 4: 実験結果

エージェントは上下左右の 4 方向に 1 マス移動することができる。ただし，壁がある方向に移動することは許されない。1 回行動を試みるごとに 1 ステップ経過したとする。エージェントの目的はゴールまでのステップ数を最小化することである。各ルールの重みの初期値は 1.0, 報酬値は 1.0, 強化関数は等比減少関数とした。また、マクロルールの重みの初期値は先頭のルールの重みとし，(4) 式を用いて重みの更新を行った。

マクロルール生成のために蓄積されるエピソード数 E を 50, 生成基準となるルール系列の数 N を 10 とした。また，エピソードから生成されるマクロルールのルール系列長を 2 と固定し，結合することができるマクロルールを 1 エピソードにつき 1 つとした。

1000 回の成功を 1 回の学習とし，利益共有法のみを用いた場合と，提案システムを用いた場合をそれぞれ 1000 回ずつ計測した。

4.2 実験結果および考察

図 4 にゴールに到達した回数と，ゴールまでのステップ数の関係を示す。また，1000 回目の成功時におけるそれぞれの平均ステップ数を表 1 に示す。

図 4 より，ゴールに到達した回数が 50 回までは両手法とも同様の軌跡でゴールに到達するまでのステップ数の減少が見られる。その後，提案システムのおいてマクロルールが生成され始めることによって，両者の平均ステップ数に違いが出ており，利益共有法のみの場合よりも提案システムの方がステップ数が少なく抑えられていることが確認できる。また，表 1 より，1000 回目の成功時において平均ステップ数の差が大きく出ていることが確認できる。

これらのような結果が生じた原因として，利益共有法のみの場合では，隠れ状態において重みの均等化が生じ，行動選択がおよそランダムに決定されてしまったためにステップ数が比較して多くなってしまったと考えられる。一方，提案システムではマクロルールの利用によって，順次エージェントの内部構造を構築することで，隠れ状態における状態を同一視してしまうという問題を避け，効率の良い適応が行われたためであると考えられる。

しかしながら，提案システムにおいても最小のステップ数（5 ステップ）にまで到達していない。このように最適なステップ数に収束しない理由として，ゴール到達に貢献しないマクロルールもまた生成され続けるため，それらのマクロルールが一定の割合で選択されていることが考えられる。また，ゴール到達に貢献しないマクロルール同士が結合されることにより，この短所がより強調されてしまうことが考えられる。このため，

表 1: 1000 回目の成功時における平均ステップ数

Profit Sharing	21.892
提案システム	13.232

提案システムは，利益共有法のみの場合と比較してある程度有効な結果が得られているとはいえ，さらなる改善の余地があると考えられる。

5. 関連研究

本研究で使用したマクロルールは，マクロオペレータの概念に基づいて構成したものである。近年，強化学習の分野においてマクロオペレータの重要性が認識されてきている。マクロオペレータは，時系列的に連続して適用されるルール群を結合したもので，これによって高次のルールを表現可能となっている。 [Precup 98] では，既存の強化学習アルゴリズムを変えることなくそのまま適用可能な option と呼ばれるマクロオペレータを提案している。また [嶋田 01] では，動的環境における適応能力を改善するためにマクロオペレータの生成，分割，結合を動的に行うシステムを提案している。本研究では，ルール系列と単一のルールを同等のものとして扱うという観点から，マクロルールと記述することにした。

また，従来より強化学習の分野ににおいて状態空間を自律的に構成を行う研究が行われている。現在に至る経過の情報を利用するために，短期記憶を表現する方法として， [McCallum 95] では Utile Suffix Memory (USM) と呼ばれる決定木構造の状態表現を利用している。そこでは，現在までに得られた知覚，行動，状態の短期記憶に基づいて，内部状態表現を環境との相互作用に即して逐次構成することにより，隠れ状態を有する問題において良い結果が得られている。また， [井上 02] では USM と類似した決定木構造を用い，自律的状態空間構成を伴う行動獲得を実移動ロボットに対して適用を行っている。[McCallum 95] では，基準になっている強化学習法が環境同定型アプローチである Q–learning を用いていることに対し，本研究では経験強化型アプローチである利益共有法を用いている。基準となる強化学習法の違いのため両手法を単純に比較することはできないが，本提案システムでは利益共有法の学習アルゴリズムを崩すことなく適用可能であるといった利点があると考えられる。

6. まとめ

本研究では，非マルコフ性を有する問題に対してできるだけ簡便かつ有効に対処する手法として，利益共有法とマクロルールを組み合わせた学習システムを提案した。また，例として隠れ状態がある迷路問題においてその有効性を検証した。提案システムは利益共有法のみの場合と比較してはある程度有効であることが確認できたが，さらなる改善の余地が必要であるといった結果が得られた。

今後の課題としては，提案システムの改善はもちろん，TD(λ)法を基にしてマクロルールを導入した学習システムの構築や，提案システムにおける他の非マルコフ性を有する問題に対して応用的な適用方法などを考えている。

参考文献

[Chirisman 92] Lonnie Chirisman, “Reinforcement Learning with Perceptual Aliasing”, Tenth National Conference on AI, (1992).

[Precup 98] D. Precup, R. S. Sutton, “Theoretical results on reinforcement learning with temporally abstract options”, Proc. 9th Intl. Conf. Machine Learning, pp.382-393, (1998).

[McCallum 95] R. Andrew McCallum, “Instance-based state identification for reinforcement learning”, In advance of neural information processing systems, (1995).

[McCallum 95] R. Andrew McCallum, “Instance-based utile distinctions for reinforcement learning with hidden state”, Proc. Intern. Conf. on Machine Learning, pp.387-396, (1995).

[井上 02] 井上庸介, 大田順, 新井民夫, “部分観測環境下での自律的状態空間構成を伴う　実移動ロボットのナビゲーション行動獲得”, 日本ロボット学会学術講演会, (2002).

[嶋田 01] 嶋田総太郎, 安西祐一郎, “マクロオペレータの部分的再利用による強化学習システムの　動的環境への適応能力の改善”, 電子情報通信学会, D-I Vol.J84-D-I, No.7 pp.1076-1088, (2001).