特　集　「マルチエージェント技術における新しい可能性」

マルチエージェント強化学習
—実用化に向けての課題・理論・諸技術との融合—

Multiagent Reinforcement Learning Frameworks: Steps toward Practical Use

荒井　幸代　　カーネギーメロン大学ロボティクス研究所
Sachiyo Arai　　The Robotics Institute, Carnegie Mellon University.
sachiyo@cs.cmu.edu

Keywords: multiagent reinforcement learning, stochastic games.

1. はじめに

マルチエージェント系のエージェント知識の設計は，系のもつ複雑さ故に非常に困難な問題であり，学習という適応的枠組みの導入，特に，陽な環境モデルや，教師信号をあらかじめ与える必要のない強化学習への期待は大きい．しかし，マルチエージェント系への適用においては，本来エージェント間の協調によって解決が期待できるはずの「不完全知覚問題」など，シングルエージェント系の深刻な課題がそのままボトルネックとなっているのが現状である．また，マルチエージェント系特有の，複数の学習，実行主体の存在に起因する「同時学習問題」や「報酬配分問題」も解決しなければならない．

以下，2章ではマルチエージェント強化学習の理解に必要な用語および代表的アルゴリズムを説明する．3章ではマルチエージェント系への強化学習適用における問題を「不完全知覚問題」，「同時学習問題」および「報酬設計問題」に限定して議論し，これらの各問題への理論的解析法について触れる．最後に実用化に向けて，他の技術との融合の可能性を議論する．

2. 準　備

2·1 用　語

図1に基本的な強化学習エージェントの枠組みを示す．エージェントは，環境認識器，行動選択器，学習器，および政策を保持する知識ベースに相当する部分を構成要素とする．時刻 t における環境の状態 $s\in\mathcal{S}$ を観測入力 $x\in X$ として受け，意思決定後，行動 $a\in\mathcal{A}$ を出力する．行動 a により環境は s' に状態遷移し，その状態の望ましさに応じた報酬 R がエージェントに与えられる．エージェントは報酬確認後，知識を更新し，再び状態観測入力を得る，というサイクルを繰り返す．

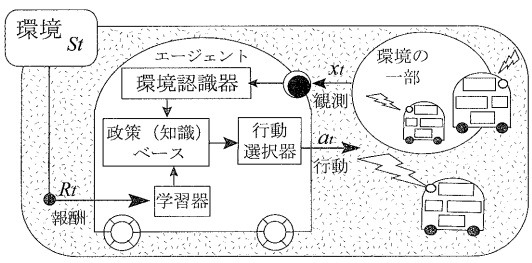

図1　強化学習エージェントモデル

§1　マルコフ決定過程

$(\mathcal{S}, \mathcal{A}, P^a_{ss'}, R)$ で定義される環境をマルコフ決定過程（Markov Decision Processes: MDPs）と呼ぶ．エージェントが状態 $s\in\mathcal{S}$ において行動 $a\in\mathcal{A}$ を実行したとき，確率 $P^a_{ss'}$ に従って s' へと状態遷移する．このとき，エージェントの状態 s における行動 a に対して確率的に与えられる実数値スカラー量を即時報酬（immediate reward）といい，その期待値 $R^a(s)$ は現在の状態と行動のみに依存する．

§2　部分観測マルコフ決定過程

環境がMDPsであっても，不完全な観測能力しか持たないエージェントは，状態遷移確率 $P^a_{ss'}$ が特定できない．このような場合，しばしばPOMDPs（Partially Observable MDPs）としてモデル化される．観測が不完全で，最適行動が状態 s_1, s_2 で相異なるにもかかわらず，これらを同一と見なしてしまう場合に生じる悪影響を不完全知覚問題（Perceptual aliasing problem）と呼ぶ．

§3　行動の最適性規範

MDPsにおけるエージェントの学習の目的は，最適な政策（policy）$\pi: \mathcal{S}\rightarrow\mathcal{A}$ を求めることである．最適政策とは，「すべての状態 $s\in\mathcal{S}$ において，state-value 関数 $V(s)$ を最大化する政策」を意味する．通常，$V(s)$ は無限期間の割引報酬の合計として定義され，これを最大化する政策（式（1））を最適政策とする．

$$\forall s\in\mathcal{S},$$

$$V^*_\infty(s)=\max_a\left(R^a(s)+\gamma\sum_{s'\in\mathcal{S}}P^a_{ss'}V^*_\infty(s')\right) \quad (1)$$

（ここで，γ $(0\leq\gamma\leq 1)$ は割引率（Discount rate）で，

将来見込まれる報酬に対して割り引く度合いを示すパラメータである.)

$V(s)$ が状態 s の value 関数であるのに対し，状態-行動対 (s, a) の value 関数 $Q(s, a)$ を Q 値と呼ぶ．Q 値は，「状態 s において行動 a を実行した後に，最適政策をとり続ける場合の無限期間の割引報酬の期待値」として式 (2) のように定義される．最適な Q 値が与えられれば，各状態 s において最大の $Q^*_\infty(s, a)$ をもつ行動選択が最適政策となる．

$$Q^*_\infty(s,a)=R^a(s)+\gamma\sum_{s'\in S}P^a_{ss'}\max_{a'\in A}Q^*_\infty(s',a') \quad (2)$$

以上のように状態遷移確率 $P^a_{ss'}$ と即時報酬 $R^a(s)$ が既知ならばダイナミックプログラミング (Dynamic Programming: DP) に基づく Value iteration や Policy iteration などを用いて最適政策を得ることができる．しかし，一般に，状態遷移確率や報酬の確率など環境のモデルを事前に知ることは難しい．そこで，次節では，これらの事前知識なしに適切な政策を獲得するためのアルゴリズムを紹介する．

2·2　強化学習アルゴリズム

代表的な強化学習アルゴリズムを図 2 のように分類する．本節では，状態-行動対の評価値を獲得するタイプのアルゴリズムのうち，マルチエージェント系に対しても利用されてきた Q-learning [Watkins 92]，そして，非マルコフ環境に頑健であるとして注目されている Monte-Carlo 法，Sarsa (λ)，Profit-Sharing について説明する．

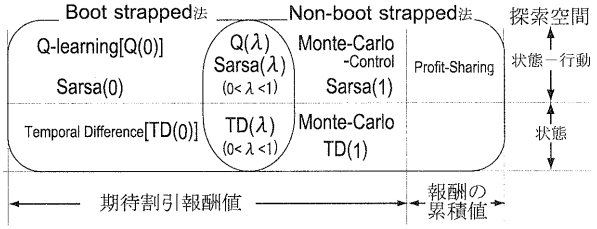

図 2　アルゴリズムの分類

§ 1　Q-learning

Q-learning の中でも広く利用されている Q (0) (One Step Q-learning) では，式 (3) のように，遷移先の観測 x_{t+1} の最大 Q 値 ($\max_{b\in A}Q_n(x_{t+1},b)$) に基づいて Q 値 ($Q(x, a)$) を推定するため，boot strapped な方法と呼ばれる [Sutton 98].

$$Q_{n+1}(x_t,a_t)\leftarrow(1-\alpha)\cdot Q_n(x_t,a_t)$$
$$+\alpha\left(r+\gamma\max_{b\in A}Q_n(x_{t+1},b)\right) \quad (3)$$

(ここで，a は非負の学習率，r は状態遷移後の報酬である.)

Q (0) は環境が MDPs であること，エージェントの観測が完全であること，および，学習率 α が適切に調整されることを前提に最適政策への収束性が保証されている．ただし，s_1 と s_2 を区別できない POMDPs 下であっても，両状態に対する最適行動が一致している場合には良い性能を示す．しかし，両状態の最適行動が相異なる場合には，この悪影響が他の Q 値の推定にも伝播してしまう．Q (0) における観測の不完全性の悪影響は文献 [宮崎 99a, Singh 94] に詳しい．なお，TD (0) も同様の性質を示す．

TD (0) や QL (0) に履歴を強化するための eligibility trace [Sutton 98] を導入した TD (λ) や Q (λ) では，λ を 1 に近づけることによって上記の悪影響の伝播を抑制することができる．

§ 2　Monte-Carlo 法

Q (0) では，遷移先の観測 x_{t+1} の最大 Q 値に基づいて Q 値を推定するのに対し，Monte-Carlo 法 (MC 法) では各時刻 t で実際に得られた報酬値に基づいて Q 値を推定する (図 3)．状態遷移先の Q 値を用いないことから non-boot strapped な方法と呼ばれる [Sutton 98].

Non-boot strapped 手法の長所として，(1) 状態遷移確率 $P^a_{ss'}$ が特定できない状況があっても，その悪影響は局所化され，非マルコフな環境や不完全知覚問題に対して頑健なこと，(2) 当面必要な (s, a) 集合の Q 値を保持すれば十分なこと，があげられる．一方，エピソード単位で一括更新する方法であるため，ゴールに到達するまで，Q 値が変化せずゴールに至るまでに「報酬につながらない行動」を繰り返す可能性があることが問題点としてあげられる．この問題はゴール到達に必要なエピソードが長いほど深刻となる*1.

Initialize, for all $x\in\mathcal{X}$, $a\in\mathcal{A}(x)$:
　$Q(x,a)\leftarrow$ arbitrary
　$\pi(x)\leftarrow$ arbitrary
　$Returns(x,a)\leftarrow$ empty list
Repeat forever

-1. Generate an episode using a soft-greedy method:
-2. For each (x,a) appearing in the episode:
　　$R\leftarrow$ return following the first*a occurrence of (x,a)
　　Append R to $Return(x,a)$
　　$Q(x,a)\leftarrow$ average($Return(x,a)$)
-3. For each x, in the episode:
　　$\pi(x)\leftarrow argmax_{a\in A}Q(x,a)$

*a　エピソード中に2度以上出現した (x, a) に対しての処理については，最初の出現時に得た報酬のみを対象とする方法 (First-visit) とすべての出現時の報酬を対象とする方法 (Every-visit) があるがいずれの方法でも MDPs における最適政策への収束性が保証されている．[Bertsekas 96, Singh 96] 参照.

図 3　A Monte-Carlo Control Algorithm [Sutton 98]

§ 3　Sarsa (λ) ($0\le\lambda\le 1$)

Sarsa ($\lambda = 0$) は Q (0) と同様に boot strapped 手法であるが，Q (0) が遷移先の観測 x_{t+1} の最大 Q 値を用い

*1　逐次更新法も提案されているが，別途メモリが必要となる.

るのに対して，Sarsa (0) では遷移先 x_{t+1} での実際に選択された行動の Q 値，$Q_n(x_{t+1}, a_{t+1})$（式（4））を用いる点で異なる*2.

一方，$\lambda = 1$ では，遷移先の Q 値を用いず，(x, a) が実際に獲得し得る報酬 $Q(x, a)$ を Replacing Eligibility Trace [Singh 96]（式（5））を用いて推定する．遷移先の $Q(x, a)$ を利用しないという点で non-boot strapped な方法であるが，MC 法と異なるのは，エピソード終了を待たずに Online で更新できる点にある．

Sarsa ($\lambda \approx 1$) の POMDPs 環境への頑健性は [Lanzi 00, Loch 98] に示されているが，Replacing Eligibility Trace 保持および更新に要するメモリ空間，計算量は膨大であり，マルチエージェント系への適用にはボトルネックとなる．

$$Q_{n+1}(x_t, a_t) \leftarrow (1-\alpha)\cdot Q_n(x_t, a_t) + \alpha(r + \gamma Q_n(x_{t+1}, a_{t+1})) \quad (4)$$

Replacing Eligibility Trace and Sarsa (λ):

1. $\delta_t \leftarrow r_t + \gamma Q_n(x_{t+1}, a_{t+1}) - Q_n(x_t, a_t)$
2. $\eta_t(x_t, a_t) = 1$
3. $\forall (x \neq x_t, a \neq a_t)\,;\, \eta_t(x, a) = \gamma\lambda\,\eta_{t-1}(x, a)$
4. $\forall (x, a)\,;\, Q_{n+1}(x, a) \leftarrow Q_n(x, a) + \alpha\cdot\delta_t\cdot\eta(x, a)$ (5)

§ 4 Profit-Sharing

上記に紹介した手法は DP に基づいて発展してきたが，進化的計算論の分野で研究が進められてきた Classifier System における Credit assignment の手法も強化学習の枠組みとして位置づけられる．これらの手法で獲得されるルール（観測入力と行動の対：(x, a)）の重みは，環境に適応するための評価値であり，DP に基づく方法における Q 値とは性質が異なる．

初期に提案された Backet brigade 法 [Holland 86] や Profit-Sharing [Grefenstette 88] では，獲得ルールの最適性，あるいは合理性の保証に関する理論的議論はされておらず，遺伝的アルゴリズム（GA）との組合せで利用することを前提として，不適切なルールは GA によって淘汰される枠組になっている．

その後，[Dorigo 94] で Backet brigade 法と Q-learning の等価性が示され，Profit-Sharing [Grefenstette 88] に対しても，[宮崎 94] がその合理性に関する理論を与え，GA との併用を行わずに合理的なルールが獲得できる枠組みとして注目されてきた．図4に，宮崎による無効ルール抑制定理（式（7））を適用した Profit-Sharing を示す．

時刻 t で発火したルールへの強化値を決定する強化関数 $f(t, R, T)$ が式（7）を満たすとき，獲得される政策は「単位ステップ当たり正の報酬が与えられる」という合理性が保証される．発火したルール (x, a) の強化値

*2 Sarsa の由来は実際の遷移 $s \to a \to r \to s \to a$ に基づく．

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$: $W(x,a) \leftarrow$ arbitrary
Repeat forever
-1. Generate an episode using a soft-greedy method:
-2. For each (x,a) appearing in the episode:*a

$$W_{n+1}(x_t, a_t) \leftarrow W_n(x_t, a_t) + f(t, R, T) \quad (6)$$

$$\forall t = 1, 2, .., T \qquad L\sum_{j=0}^{t} f(j, R, T) < f(t, R, T) \quad (7)$$

*a 式(6)で R はゴール到達時に得る報酬，W はルール (x,a) の重み，T はエピソード長，n は更新回数を示す．また，式(7)における L は有効ルールの最大個数を示す．

図4 無効ルール抑制定理を利用した Profit-Sharing

が遷移先に無関係に独立に更新される non-boot strapped 手法であるため，POMDPs に対して頑健であるが，MDPs における最適性は保証されていない．また，エピソード単位の一括更新法であるため，MC 法と同様の問題点を抱えると同時に，式（7）を満たす強化関数はゴールから遠いルールの重みの更新が困難なこと，が問題点としてあげられる．この問題はエピソードが長いほど深刻である．

MC 法や Sarsa (1) と異なる点は，これらが Widrow-Hoff Rule に基づいて (x, a) の Q 値を推定する方法であるのに対して，Profit-Sharing は強化値を上乗せしていく方法であるため，(1) ステップサイズ α などのパラメータ調整を必要としないこと，そして，(2) ユニークな Q 値の特定ができない不完全知覚領域においても有効な確率的政策が獲得できること [荒井 98]，が長所としてあげられる．「上乗せ」かあるいは「Q 値の推定」かに関する議論は [Sutton 98] に詳しい．

3. マルチエージェント強化学習の課題

本章ではマルチエージェント強化学習において考慮すべき問題のうち，(1) 不完全知覚問題，(2) 同時学習問題，および (3) 報酬配分問題，の三つを取り上げて説明する．

3・1 不完全知覚問題

一般に大規模な状態空間を対象とするマルチエージェント系では不完全知覚問題は免れない．不完全な知覚による状態遷移確率 $P^a_{ss'}$ の非定常性は環境の状態遷移確率そのものの変化ではなく，不完全な知覚しかもたないエージェントの「誤った状態遷移の認識」によるものであることに注意したい．

単純に他のエージェントを環境の一部とみなし，情報の共有が期待できない場合には，Non-boot strapped な方法である MC 法や Profit-Sharing，あるいは Sarsa ($\lambda > 0$) を利用するのが望ましい．ただし，報酬がエピソード単位で与えられるタスクの場合には，時間計算量，空間計算量の側面から Profit-Sharing が有望である．[Arai 00] は Profit-Sharing を各エージェントに

適用し，陽に定義された通信による情報の交換を行うことなしに，マルチエージェント間で不完全知覚問題を解消し協調行動を創発できたことを報告している．

3·2　同時学習問題

学習途中で定常な政策をもたないエージェントが複数存在する系において，自己の行動による状態遷移先を特定することは難しい．ここでは，この難しさに起因する問題を同時学習問題と呼ぶ．

一般に，他エージェントの入出力前後の環境の状態変化を完全に観測することができないため，自己の行動による真の遷移先を認識することは難しい．なぜなら，遷移先 s' は必ずしも自己の行動 a のみによるものではなく，他エージェントの行動との連帯行動（joint actions）による場合が多いからである．自分以外の他エージェントの政策が定常であれば問題はないが，他エージェントも学習によって時々刻々と政策を変化させるため，環境の状態遷移確率 $P^a_{ss'}$ が特定できないために学習に悪影響を及ぼす原因となる．したがって，理論上は状態遷移先の Q 値を利用する boot strapped な方法は適切ではない．

他エージェントを環境の一部とみなし，通信なしの同時学習下で Q-learning をロボットの協調タスクに適用した [Mataric 97, Sen 94] では収束解を得ており，[荒井 98] では不完全知覚問題のない追跡問題に対して「獲物がエージェントの捕獲行動と独立に行動する場合」には Q-learning は同時学習下でも最終的に最適解を得るという実験結果を示している．しかし，これらの実験の問題設定はいずれも共通のゴールをもつエージェント間の協調タスクで，各エージェントが独立に獲得する報酬の最大化が全エージェントの報酬の最大化につながる問題であることに注意しなければならない．最近の話題である電子商取引における価格設定 [Tesauro 99] をはじめとする競争的関係にあるエージェント間での同時学習問題は特に重要である．

3·3　報酬配分問題

マルチエージェント系ではしばしば，エージェント間の連帯行動（joint actions）に対する報酬は定義できても，各エージェントの個別の行動に対する報酬を定義することは難しい．ここでは，この難しさに起因する問題を報酬配分問題と呼ぶ．適切に報酬が配分されなければ，各エージェントの正しい学習のみならず，系全体の望ましい挙動は期待できない．

(1) サッカーゲームにおけるゴールまでのパスの例を考える：
(例) Agent1 → Agent2 →..., AgentG ⇒ GOAL
ここで，直接報酬に寄与した AgentG のみに報酬を与えると，AgentG 以外は全く学習しないことは明らかである．

(2) 2 人のハンターで捕獲する追跡問題において，冗長な 2 人のハンターを混在させた場合を考える（図 5）：ここで直接捕獲に寄与しなかった 2 人にも報酬を与えると，捕獲に結び付かない行動（初期位置の周辺をうろつくなど）が強化されてしまい「すべてのハンターが冗長な行動をとるようになる」結果，冗長エージェントの存在しない場合に比較して捕獲の効率が低下している．報酬配分問題は強化学習によるマルチエージェントシステム系の挙動を単に創発現象と捉えるのではなく，工学的な利用へと引き上げるための重要な問題である．

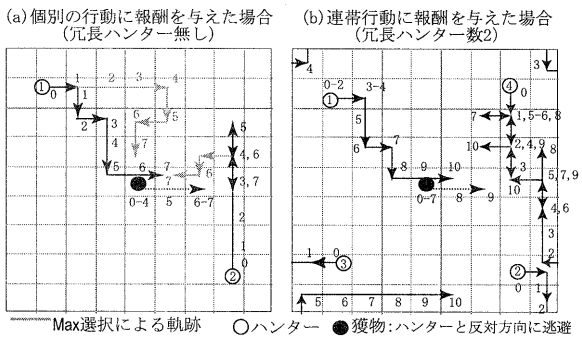

図 5　連帯行動への報酬の悪影響の例

4. 理論的枠組み

本章では，3 章であげた「同時学習問題」および「報酬配分問題」に対して解決の緒を与える理論的枠組みを紹介する．なお，「不完全知覚問題」に関する理論を含めた議論は [木村 97] に譲る．

4·1　同時学習問題への理論

Stochastic Games: 同時学習を扱うための理論的枠組みとして Stochastic Game を紹介する．Stochastic Games（SG）はゲーム理論研究において最初に登場した概念である．SG は，「単一エージェント，複数状態」を扱う枠組みの MDPs と「複数エージェント，単一状態」を扱う枠組みである Marix Games（MG）を総括して扱えるため，「複数エージェント，複数状態」からなるマルチエージェント強化学習の理論的解析に利用できる．SG は $(n, \mathcal{S}, \mathcal{A}_{1...n}, P^{\mathcal{A}}_{ss'}, R_{1...n})$ で定義される．n はエージェント数，$\mathcal{S}$ は環境の状態集合，$\mathcal{A}_i$ はエージェント i が選択可能な行動集合，$\mathcal{A}$ は全エージェントの連帯行動集合空間（joint action space），$\mathcal{A}_1 \times ... \times \mathcal{A}_n$ である．また，$P^{\mathcal{A}}_{ss'}$ は連帯行動 $\mathcal{A}$ による遷移確率，R_i は，各エージェント i に与えられる報酬である $(\mathcal{S} \times \mathcal{A} \rightarrow \mathbf{R})$*3.

Minimax-Q: [Littman 94] は SG の零和ゲームの例として 1 対 1 のサッカーゲームを取り上げ Minimax Q-learning [Littman 94] を提案した．また，[Hu 98] は

*3 ここでは個々のエージェントに対して報酬 R_i が定義されることに注意．

Minimax Q-learningをSGの非零和ゲームを扱える枠組みに拡張し，「均衡点が唯一であること」，「他エージェントの行動とその報酬を情報として与えられること」を条件としてNash均衡解に収束することを示した．しかし，Minimax-Qでは均衡点に収束するものの，最良解あるいはゴール達成に結び付く解に収束するとは限らない．

[Bowling 01]は「ゴール達成に結び付く解を得ること」をマルチエージェント強化学習における合理性であると定義し，この合理性と同時に収束性も保証するアルゴリズム *WoLF Policy Hill Climbing* (*PHC*) を提案している．

4·2 報酬配分問題への理論

[宮崎 99b]は，Profit-Sharingを用いた場合において，系全体の単位行動当たりの期待獲得報酬が正となるための各エージェントへの報酬配分に関する必要十分条件を与えている（式（8））．ここでは，ある時刻で $(n'-1)$ $(0<n'\leq n)$ 個のエージェントがそれぞれある特別な感覚を得ているときに，n' 個目のエージェントがある特別な感覚を得たとき，その n' 個目のエージェントに**直接報酬** R $(R>0)$ が与えられ，それ以外の $(n-1)$ 個のエージェントに**間接報酬** μR $(\mu\geq 0)$ が与えられるものとする．

n' の値，および，ある特別の感覚が何であるかは一般には未知である．そのため，外界から報酬が与えられる直前に行動を出力したエージェントが報酬獲得に貢献したことは明らかであるが，それ以外のエージェントに関しては，報酬獲得に貢献したのか，妨害したのか，あるいは無関係なのかはいっさいわからない．したがって，μ を適切に設定することが非常に重要となる．

$$\mu<\frac{M-1}{M^{T}\left(1-\left(\frac{1}{M}\right)^{T_0}\right)(n-1)L} \qquad (8)$$

（ここで，T は直接報酬を獲得したエージェントの最大エピソード長，T_0 は直接報酬を獲得したエージェント以外のエージェントの強化区間，M は競合数，L は同一感覚入力下に存在する有効ルールの最大競合数である．）

この定理は，他エージェントからの間接報酬による副作用を抑制するために導出された定理である．一般には，間接報酬が合理的ルールを強化するような加速効果も十分考えられる．この定理は，そのような加速効果に関しては何も保証していないが，報酬が全く得られなくなるという最悪の事態を回避したうえで，システム設計者に，間接報酬による加速効果を期待させるものとして非常に意味がある．

5. 他技術との融合

理論的解析を進める一方，マルチエージェントシステムを実用レベルに引き上げるためには他技術との融合が不可欠である．幸い，強化学習は「状態入力表現，政策表現に制約がない」ことから，ニューラルネット，人工知能，および進化的計算等の諸技術との融合が比較的容易に可能である．

3章に示した三つの問題は，少なくとも協調的エージェント間において，契約ネットや黒板モデルを介した入力情報の共有によって解決の途が開かれており，[Schneider 99, Tan 93]は情報の共有による性能や効率が上がることを示している．また，競争エージェント間においては，他エージェントのモデルの推定や保持が考えられるが，いずれの場合も，状態空間の爆発問題の悪化を招き得る．

ここでは状態空間の爆発問題に焦点を絞り，(1) 状態表現を連続化し，政策を関数近似表現する方法，(2) 階層表現の利用，そして (3) 他の機械学習法の導入可能性について簡単にまとめる．

(1) 関数近似表現には，小脳モデルであるCMAC，ニューラルネット，ファジイなどがあげられる．しかし，CMACやニューラルネットの併用は強化学習の利点である「教師信号が不要なこと」，「遅れのある報酬が扱えること」，「知識（政策）表現の透明性」を損なうことにもつながり，エージェント間の情報共有も不透明化する恐れがある．

(2) 階層的表現は，問題のモジュール化という点においてマルチエージェントシステムになじみやすい．[Parr 98]による政策の階層化や，Valueの階層化に基づいたMaxQ [Dietterich 98]はマルチエージェント系においても利用されている[Makar 00]．ここでは，各モジュールをエージェントと見立てた階層化における上位から下位エージェントへの報酬の伝播，あるいは同一レベルエージェント間での報酬分配の設計が重要な鍵となる．この報酬配分問題に対し，[Humphrys 96]は遺伝的アルゴリズムを用いて報酬の評価を行い，適切な報酬設計に成功している．

(3) ほかの機械学習の導入については，決定木を用いた状態や政策空間の構造化 [Boutilier 96]による環境認識に要する照合時間の大幅な短縮や，EBLの導入による学習の効率化の実現 [Dietterich 95]，あるいは，CBLを導入した獲得政策の再利用など，さまざまな帰納学習方法との融合が期待できる．

6. ま と め

本稿では強化学習の基礎を概観し，マルチエージェント強化学習の理論的課題と接近法の一部を紹介した．強化学習はエージェント設計に対して期待できる接近法であるが，実用化に向けては他技術との融合は必須であり，今後も理論および実験，両面からの議論や検証が必要である．

謝　辞

本稿をまとめるにあたり，終始アドバイスをいただきました名古屋工業大学の新谷虎松先生，大学評価・学位授与機構の宮崎和光先生，カーネギーメロン大学Michael Bowling氏には心から感謝します．

◇ 参 考 文 献 ◇

[Arai 00] Arai, S. and Sycara, K.: Effective Learning Approach for Planning and Scheduling in Multi-Agent Domain,In *Proceedings of the 6th International Conference on Simulation of Adaptive Behavior,* pp. 507-516 (2000)

[荒井 98] 荒井幸代, 宮崎和光, 小林重信: マルチエージェント強化学習の方法論— Q-learning と Profit Sharing による接近, 人工知能学会誌, Vol. 13, No. 5, pp. 609-618 (1998)

[Bertsekas 96] Bertsekas, D. P. and Tsitsiklis, J. N.: Neuro-Dynamic Programming, *Athena Scientific* (1996)

[Boutilier 96] Boutilier, C.: Planning, Learning and Coordination in Multiagent Decision Processes, In *Proceedings of the 6th Conference on the Theoretical Aspects of Rationality and Knowledge,* pp. 195-210 (1996)

[Bowling 01] Bowling, M. and Veloso, M.: Rational and Convergent Learning in Stochastic Games, to appear in *Proceedings of the 17th International Conference on Artificial Intelligence* (2001)

[Dietterich 95] Dietterich, T. G. and Flann, N. S.: Explanation-Based Learning and Reinforcement Learning: A Unified View, In *Proceedings of the 12th International Conference on Machine Learning,* pp. 176-184 (1995)

[Dietterich 98] Dietterich, T. G.: The MAXQ Method for Hierarchical Reinforcement Learning, In *Proceedings of the 15th International Conference on Machine Learning,* pp. 118-126 (1998)

[Dorigo 94] Dorigo, M. and Bersini, H.: A Comparison of Q-learning and Classifier Systems, In *Proceedings of the 3rd International Conference on Simulation of Adaptive Behavior,* pp. 248-255 (1994)

[Grefenstette 88] Grefenstette, J.: Credit Assignment in Rule Discovery Systems Based on Genetic Algorithms, *Machine Learning 3,* pp. 225-245 (1988)

[Holland 86] Holland, J. H.: Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems,In R.S.Michalsky, et al. (eds.), *Machine Learning: An Artificial Intelligence Approach,* Vol. 2, pp. 593-623, Morgan Kaufman (1986)

[Hu98] Hu, J. and Wellman, M. P.: Multiagent Reinforcement Learning: Theoretical Framework and an Algorithm, In *Proceedings of 5th International Conference on Machine Learning,* pp. 242-250 (1998)

[Humphrys 96] Humphrys, M.: Action Selection methods using Reinforcement Learning, In *Proceedings of the 4th International Conference on Simulation of Adaptive Behavior,* pp. 135-144 (1996)

[木村 97] 木村 元, and Kaelbling, L. P.: 部分観測マルコフ決定過程下での強化学習, 人工知能学会誌, Vol. 12, No. 6, pp. 822-830 (1997)

[Lanzi 00] Lanzi, P. L.: Adaptive Agents with Reinforcement Learning and Internal Memory, In *Proceedings of the 6th International Conference on Simulation of Adaptive Behavior,* pp. 333-342 (2000)

[Littman 94] Littman, M. L.: Markov Games as a Framework for Multi-agent Reinforcement Learning, In *Proceedings of the 11th International Conference on Machine Learning,* pp. 157-163 (1994)

[Loch 98] Loch, J. and Singh, S. P.: Using Eligibility Traces to Find the Best Memoryless Policy in Partially Observable Markov Decision Processes, In *Proceedings of the 15th International Conference on Machine Learning* (1998)

[Makar 00] Makar, R. and Mahadevan, S.: Hierarchical Multi Agent Reinforcement Learning, In *Advances in Neural Information Processing Systems 12,* pp.345-352 (2000)

[Mataric 97] Mataric, J.: Reinforcement Learning in the Multi-Robot Domain, *Autonomous Robots,* Vol. 4, No. 1, pp. 77-83 (1997)

[宮崎 94] 宮崎和光, 山村雅幸, 小林重信: 強化学習における報酬割当の理論的考察, 人工知能学会誌, Vol. 9, No, 4, pp. 104-111 (1994)

[宮崎 99a] 宮崎和光, 荒井幸代, 小林重信: POMDPs 環境下での決定的政策の学習, 人工知能学会誌, Vol. 14, No. 1, pp. 148-156 (1999)

[宮崎 99b] 宮崎和光, 荒井幸代, 小林重信: Profit Sharing 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, No. 6, pp. 1156-1164 (1999)

[Parr 98] Parr, R. and Russell, S.: Reinforcement Learning with Hierarchies of Machines, In *Advances in Neural Information Processing Systems 10,* pp. 1043-1049 (1998)

[Schneider 99] Schneider, J., Wong, W., and Moore, A.: Distributed Value Functions, In *Proceedings of the 16th International Conference on Machine Learning,* pp. 371-378 (1999)

[Sen 94] Sen, S., Sekaran, M., and Hale, J.: Learning to Coordinate without Sharing Information, In *Proceedings of the 12th National Conference on Artificial Intelligence,* pp. 426-431 (1994)

[Singh 94] Singh, S., Jaakkola, T., and Jordan, M.: Learning Without State-Estimation in Partially Observable Markovian Decision Processes, In *Proceedings of the 11th International Conference on Machine Learning,* pp.284-292 (1994)

[Singh 96] Singh, S. and Sutton, R. S.: Reinforcement Learning with Replacing Eligibility Traces, *Machine Learning 22,* pp. 1-37 (1996)

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

[Tan 93] Tan, M.: Multi-agent Reinforcement Learning: Independent vs. cooperative agents, In *Proceedings of the 15th International Conference on Machine Learning,* pp. 330-337 (1993)

[Tesauro 99] Tesauro, G. and Kephart, J. O.: Pricing in Agent Economies with Multi-agent Q-learning, Generating Cooperative Behavior by Multi-Agent Reinforcement Learning, *Workshop on Decision Theoretic & Game Theoretic Agent,* see also, `http://www.research.ibm/infoecon/researchpapers.html` (1999)

[Watkins 92] Watkins, C. and Dayan, P.: Technical note: Q-learning, *Machine Learning 8,* pp. 55-68 (1992)

2001年5月14日　受理

著 者 紹 介

荒井　幸代（正会員）

1984年慶應義塾大学工学部計測工学科卒業．（株）ソニー，米U.C.Berkeleyを経て，1998年東京工業大学大学院総合理工学研究科知能科学専攻博士後期課程修了．博士（工学）．現在，米Carnegie Mellon University, Robotics Institute研究員．マルチエージェント間意思決定に関する研究に従事．AAAI学会会員．