知能と複雑系　131－36
(2003. 1. 31)

マルチエージェント系における方策勾配法
――追跡問題――

石原 聖司　　五十嵐 治一

近畿大学工学部
〒739-2116 広島県東広島市高屋うめの辺１番
E-mail: {ishihara, igarashi}@hiro.kindai.ac.jp

あらまし　マルチエージェント系における行動学習法として方策勾配法を用いる強化学習方式を提案する．本方式では，自律分散的な行動方式を採用することにより，マルチエージェント系の行動決定問題を各エージェント内で定義されたある目的関数の最小化問題に帰着させる．目的関数は，状態の価値，状態・行動ルール，ポテンシャル等の項によって表現することができる．本方式による学習則は，これらの項に含まれるパラメータを，ボルツマン分布による確率的方策から得られる系の行動計画に対する評価関数の期待値が最大となるよう確率的勾配法によって更新するものである．本方式の適用例として，追跡問題に対する実験の結果を合わせて報告する．実験の結果，本方式によって良好な方策が得られることを確認した．さらに，本方式は，方策中への行動制約や目的の変更追加，ヒューリスティクスの利用に柔軟に対応できることを示した．

キーワード　強化学習，方策勾配法，追跡問題，マルチエージェント系

Policy Gradient Method in Multi-Agent Systems
――Pursuit Problem――

Seiji ISHIHARA　and　Harukazu IGARASHI

School of Engineering, Kinki University
1 Takaya-Umenobe, Higashi-Hiroshima, Hiroshima, 739-2116 Japan
E-mail: {ishihara, igarashi}@hiro.kindai.ac.jp

Abstract　We propose a method using the policy gradient for reinforcement learning in multi-agent systems. In our approach, motion planning problems in multi-agent systems are formulated as problems that each agent selects its actions to minimize each objective function independently. The objective function can be defined by a state-value function, the sum of weight parameters of state-action rules, and heuristic potentials. The functions include some parameters. The parameters are updated stochastically in order to maximize the expectation of the reward based on a history of states and actions in each episode. The results of experiments for the pursuit problem showed that our method can make short episode plans as Q-learning does, and can easily deal with limitations such as time-window restrictions imposed on the episode length and heuristic knowledge such as an attractive potential to the target.

Keyword　reinforcement learning,　policy gradient method,　pursuit problem,　multi-agent system

1. はじめに

マルチエージェント系における行動学習方式として強化学習を用いる研究が行われている[1][2][3]. このようなマルチエージェント系での強化学習においては，マルチエージェント系特有の難しさがある.

その中でも，以下の点が特に難しい問題であるとされている：①状態爆発，②同時学習，③シーソー現象，④報酬配分. まず，①の状態爆発の問題は，エージェントの増加による状態空間の増大を意味している[1]. エージェント群の行動の組合わせ数はエージェント数のべき乗に比例することから，状態爆発という問題の深刻さが理解できる.

この問題を解決するには，エージェント系全体の行動決定を一括して行う“集中方式”をあきらめて，個々のエージェントが自己の行動を独立して決定する“自律分散方式”を採用するのが有力な手法である. この方式を採用すると，自分以外の他のエージェントは環境とみなすことになるが，それぞれのエージェントが独立に学習を行うために，個々のエージェントにとっては環境が時間変動することになり，Q学習等で前提としている環境の定常性が失われてしまう. これが上記②であげた同時学習問題である[3].

また，他のエージェントの状態を観測することなく，あるいは観測していても何の取り決めもなく一斉に行動してしまうと，望ましくない発振が生じてしまう場合もある（③のシーソー現象）[1]. さらに，系全体の状態や行動に報酬を与えたい場合，自律分散方式では学習の途中でだれにどれだけの報酬を与えるかを決定しなければならないという④の報酬配分問題が生ずる[3].

自律分散方式における②～④の問題を解決するのは容易ではないが，我々は集中方式と自律分散方式との関係を数理的に明らかにすることが，まず，これらの問題解決には必要であると考えた.

そこで，本研究では，マルチエージェント系の行動学習方式として，強化学習の一方式である“方策勾配法”(Policy Gradient Method)を用いる. まず，この学習法をオンライン型行動決定という仮定の下で議論する. 次に，集中方式と自律分散方式との関係を明確にし，自律分散方式における学習則を導出した. さらに，本手法の適用例として追跡問題を取り上げ，いくつかの実験を行った. 一部の実験ではQ学習を適用した実験も行い，結果を比較した.

2. マルチエージェント系における行動決定

一般的なマルチエージェント系における行動計画問題を考える. エージェント数を N, 時刻 t におけるエージェント i の状態を $s_i(t)$, 行動を $a_i(t)$, エージェント系全体の行動を $a(t)=\{a_i(t)\}$ $(i=1,\cdots,N)$で表す. また，系全体の1エピソード（長さ L）の行動計画を$\{a(t)\}$ $(t=0,\cdots,L-1)$で表す.

2.1. オンライン型行動決定

ここでは，エージェントのある時刻における行動は，その時刻ごとに（方策 π により）決定する方式を採用する. すなわち，

$$\pi(\{a(t)\}) = \pi(a(0))\pi(a(1))\cdots\pi(a(L-1)) \tag{2.1}$$

が成立することを仮定する. これを“オンライン型行動計画法”と呼ぶ. この仮定は実時間での行動決定や，環境の動的変化への対応に適している. さらに，**2.3.**以下での理論的な取り扱いが易しくなるという利点がある.

2.2. 目的関数と方策

時刻 t におけるエージェント系全体の目的関数を $E(a(t);s(t),\{\theta_{ij}\})$とし，時刻 t における系全体の行動を決定する方策 $\pi(a(t);s(t),\{\theta_{ij}\})$を以下のボルツマン型の分布関数で定義する.

$$\pi\left(a(t);s(t),\{\theta_{ij}\},T\right) \equiv \frac{e^{-E(a(t);s(t),\{\theta_{ij}\})/T}}{\sum_a e^{-E(a;s(t),\{\theta_{ij}\})/T}} \tag{2.2}$$

ただし，θ_{ij}は，目的関数 E に含まれるエージェント i に関する j 番目のパラメータである.

2.3. 方策勾配法によるパラメータの学習

強化学習においては，行動価値関数 $Q^{\pi}(s,a)$や状態価値関数 $V^{\pi}(s)$を通して間接的に方策 π を学習する場合が多いが，方策中のパラメータを確率的勾配法により直接学習する方式がある. WilliamsのREINFORCEアルゴリズム[4]や木村らの確率的傾斜法[5]などである. 本稿ではこれらの学習を総称して“方策勾配法”(policy gradient method)と呼ぶ.

今，行動列$\{a(t)\}$は系の行動計画を表し，評価関数 $r(\{a(t)\})$により評価値が与えられるとする. この評価値の期待値 $V(\pi)= E[r(\{a(t)\})]$を最大化するように，方策勾配法を適用する.すなわち，式(2.1)の仮定と式(2.2)の方策とを用いると，

$$\frac{\partial V}{\partial \theta_{ij}} = \frac{\partial}{\partial \theta_{ij}} \sum_{\{a(t)\}} \pi(\{a(t)\}) r(\{a(t)\}) \tag{2.3}$$

$$= E\left[r(\{a(t)\}) \cdot \sum_{t=0}^{L-1} e_{ij}(t) \right] \tag{2.4}$$

となり，確率的勾配法により次の学習則を得る.

$$\Delta\theta_{ij} = \varepsilon \cdot r(\{a(t)\}) \cdot \sum_{t=0}^{L-1} e_{ij}(t) \tag{2.5}$$

ここで，$\varepsilon(>0)$は学習係数，$e_{ij}(t)$は適正度[4] で，

$$e_{ij}(t) \equiv \frac{\partial}{\partial \theta_{ij}} \ln \pi = -\frac{1}{T}\left(\frac{\partial E}{\partial \theta_{ij}} - \left\langle \frac{\partial E}{\partial \theta_{ij}} \right\rangle_T \right) \tag{2.6}$$

により計算できる．ただし，$<\cdots>_T$ は式(2.2)の分布による期待値操作である．パラメータの更新は，エピソード終了時ごとに行う．

3. 自律分散的な計画方式

3.1. 方策における近似と学習則

処理時間の観点からは，各エージェントごとに,各々の目的関数 $E_i(a_i(t);s_i(t))$ と方策 $\pi_i(a_i(t);s_i(t))$ とにより，独立に計画を立てる方が望ましい（自律分散方式）．そこで，各時刻 t において以下の近似を用いる．

$$\pi\left(a(t),s(t)\right)\approx\prod_i\pi_i\left(a_i(t),s_i(t)\right) \tag{3.1}$$

ただし，

$$\pi_i\left(a_i(t),s_i(t)\right)\equiv\frac{e^{-E_i(a_i(t);s_i(t),\{\theta_{ij}\})/T}}{\sum_{a_i}e^{-E_i(a_i;s_i(t),\{\theta_{ij}\})/T}} \tag{3.2}$$

この近似を用いると，式(2.6)の適正度は，

$$e_{ij}(t)\equiv\frac{\partial}{\partial\theta_{ij}}\ln\pi_i=-\frac{1}{T}\left(\frac{\partial E_i}{\partial\theta_{ij}}-\left\langle\frac{\partial E_i}{\partial\theta_{ij}}\right\rangle_T\right) \tag{3.3}$$

となるが，学習則(2.5)はそのまま成り立つ．なお，以下では簡単のため，s_i，a_i，π_i，E_i，θ_{ij} などの記号におけるエージェントに関する添え字 i は省略する．

3.2. 目的関数の例 1：状態の価値

状態 s の価値を表すパラメータ $\theta(s)$により，エージェントの目的関数を以下のように表す．

$$E\left(a;s,\{\theta(s)\}\right)\equiv-\sum_{s'}\theta(s')\delta_{s',u(a;s)} \tag{3.4}$$

ただし，$u(a;s)$は，状態 s において，行動 a を選択したときに予想される遷移先の状態を表す．本稿では簡単のために，遷移先の状態 $s'\equiv u(a;s)$は決定論的に 1 つの状態に定まるものと仮定する．もし，遷移先の状態が確率的にしか定まらない場合は,その遷移確率 $P^a_{s,s'}$ を式(3.4)の右辺の $\delta_{s',u(a;s)}$の代わりに用いればよい．なお，$\delta_{s,s'}$は，$s=s'$ならば 1，$s\neq s'$ならば 0 をとる関数である．

目的関数が式(3.4)で表されているとき，パラメータ $\theta(s)$ の学習則(2.5)は，

$$\Delta\theta(s)=\varepsilon\cdot r\sum_{t=0}^{L-1}\frac{1}{T}\left[\delta_{s,u(a(t),s(t))}-\sum_a\delta_{s,u(a;s(t))}\pi\left(a;s(t)\right)\right] \tag{3.5}$$

となる．この学習則の意味を考えると，次のように解釈できる．

(1)エピソード中，状態と行動の対として，(s,a)が出現すれば，$s'=u(a;s)$の重み $\theta(s')$を，$\varepsilon\cdot r[1-\pi(a;s)]/T$ だけ増加させる．

(2)出現した状態 s において，選択しなかった行動 $b(\neq a)$ による遷移先の状態 $s'=u(b;s)$ の重み $\theta(s')$ は，$\varepsilon\cdot r\cdot\pi(b;s)/T$ だけ減少させる．

(3)それ以外の状態の重みは更新しない．

したがって，報酬値の高いエピソード中に選ばれた状態・行動対は選択されやすくなり，他の行動は抑制される．かつ，エピソード中に出現しなかった状態の重みは更新しない．また，この学習則によりすべてのパラメータの更新量がゼロとなるのは，すべての状態 s において，取りうるべき行動 a が一意的に定まる（i.e.その行動の選択確率が 1 となる）場合である．

3.3. 目的関数の例 2：状態・行動ルール

目的関数 E を，“状態 s ならば行動 a をとる”という IF-THEN 型ルールの重み $\theta(s,a)$の和，

$$E\left(a;s,\{\theta(s,a)\}\right)\equiv-\sum_{s'}\sum_{a'}\theta(s',a')\delta_{s,s'}\delta_{a,a'} \tag{3.6}$$

で表すと，学習則(2.5)は，

$$\Delta\theta(s,a)=\varepsilon\cdot r\sum_{t=0}^{L-1}\frac{1}{T}\delta_{s,s(t)}\left[\delta_{a,a(t)}-\pi\left(a;s(t)\right)\right] \tag{3.7}$$

となる．**3.2.**の場合と同じように学習則の意味を考えると，

(1)エピソード中，状態と行動の対として，(s,a)が出現すれば，そのルールの重み $\theta(s,a)$を，$\varepsilon\cdot r[1-\pi(a;s)]/T$ だけ増加させる．

(2)出現した状態 s において，選択しなかった行動 $b(\neq a)$ に関するルールの重み $\theta(s,\ b)$は，$\varepsilon\cdot r\cdot\pi(b;\ s)/T$ だけ減少させる．

(3)それ以外の状態・行動対に関するルールの重みは更新しない．

となっている．したがって，報酬値の高いエピソード中に出現した状態・行動対の重みは増大し，他の行動は抑制される．かつ，出現しなかった状態に関するルールの重みは更新しない．また，この学習則による学習が停止するのは，全ての状態 s において，取りうるべき行動 a が一意的に定まる（i.e.その行動の選択確率が 1 となる）場合である．

3.4. 目的関数の例 3：ポテンシャル

移動ロボットの行動決定の場合，ポテンシャル法という手法がよく用いられる．これは，例えば，ゴールからは引力が，他のロボットや障害物からは斥力が働くと考え，これらの力を何らかのポテンシャルの形で与える方式である．ここでは，エージェントの目的関数が，次のようにいくつかのポテンシャルの線形和で与えられる場合を考える．

$$E\left(a;s,\{\theta_j\}\right)\equiv\sum_j\sum_{s'}\theta_jU_j(s')\delta_{s',u(a;s)} \tag{3.8}$$

ここで，$U_j(s)$は状態 s の j 番目のポテンシャル関数であり，θ_jはそのポテンシャル関数を用いる重みである．このとき，学習則(2.5)は，

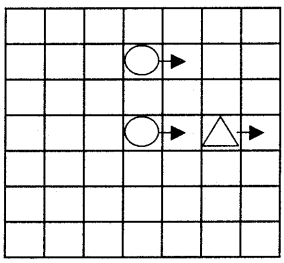

図 1 追跡問題の例．ハンター“○”が獲物“△”を追跡する．各矢印は方策によって選択される行動を表す．

Fig.1 Example of pursuit problem. Circle agents pursue a triangle agent. Each arrow shows a direction of action selected by a policy.

$$\Delta\theta_j = -\varepsilon \cdot r \sum_{t=0}^{L-1} \frac{1}{T}\left[\sum_{s'} U_j(s')\delta_{s',u(a(t),s(t))} - \sum_a \sum_{s'} U_j(s')\delta_{s',u(a(t),s(t))}\pi(a;s(t))\right] \tag{3.9}$$

となる．

また，目的関数として，例 1，例 2，例 3 であげた 3 つの関数を線形結合により合成した関数を用いることも可能である．実際，次章で述べる追跡問題では，例 2 の状態・行動対の IF-THEN 型ルールと，例 3 のポテンシャル型の知識（ヒューリスティクス）とを加え合わせた関数を目的関数とする場合も取り扱っている．

4. 追跡問題への適用

4.1. 追跡問題とは

図 1 に示すように，2 次元のグリッド上において，追跡役の複数のエージェント（ハンター“○”）が逃亡役の 1 つのエージェント（獲物“△”）を捕らえるまで追いかけるというのが追跡問題である．追跡問題においては，各エージェントの配置が状態 s，マスの移動が行動 a に相当する．

本研究では文献[2]に従って，追跡問題を，

- 全ハンターが獲物の上下左右いずれかに隣接することを目標とする．
- 目標達成時に全ハンターに報酬を与える．
- 各エージェントは決まった順序で行動する．
- 各エージェントは上下左右いずれかの方向へ 1 マスの移動，もしくは，停止の 5 つの行動パターンから 1 つを選択する．
- 複数のエージェントが同時に 1 つのマスを占有することはできない．

と定義する．

4.2. 実験条件

4.1.で定義した追跡問題について，次に示す条件の下，方策勾配法を用いた 3 つの異なる行動計画実験を行った．

- 2 次元のグリッドを 7×7 の格子状トーラスとする．
- エージェントの初期配置はランダムとする．
- ハンターの視界を 7×7 とする．
- ハンターの数を 2 とする．
- 各ハンターは式(3.2)に従い確率的に行動を決定する．
- 獲物はランダムに行動を決定する．

初期状態から目標達成時までを 1 エピソードとし，目標達成に要した時間ステップ数をエピソード長 L とする．なお，L>1000 となる試行については，学習データとしてカウントしないものとした．次の **4.3.**から **4.5.**において，各実験の内容と結果を示す．

4.3. 実験 1：状態・行動ルールを用いて最短エピソードを求める場合

状態・行動ルールを用いた式(3.6)で表される目的関数の下，ルールの重み $\theta(s,a)$を方策勾配法によって自律分散的に学習する実験を行った．目標達成時，$r=1/L^2$ の報酬値を全ハンターに与えることにより，より短いエピソードをもたらす方策の導出を目指した．

実験は，ルールの重みの更新を 10 万エピソード繰り返す試行を 1 セットとし，計 5 セット行った．なお，温度 T および学習係数 ε の値については，予備実験の結果に基づきいずれも 0.2 に設定した．さらに，ルールの重みの初期値はすべて 0.1 とした．

一方，方策勾配法に対する比較として，Q 学習による自律分散的な学習の実験を同様に 5 セット行った．この際，文献[2]に基づいて，報酬値 r=1，温度 T=0.2，学習係数 α=0.04，割引率 γ=0.9 とそれぞれ設定した．さらに，Q 値の初期値はルールの重み同様 0.1 とした．

エピソード長に関する 5,000 エピソード毎の平均値（平均エピソード長）の推移を，方策勾配法については図 2，Q 学習については図 3 にそれぞれ示す．ただし，平均エピソード長に関する 5 セット間の平均値を折れ線で結ぶと共に，その最大値と最小値をエラーバーで表示した．これらの図からわかるように，方策勾配法は，Q 学習とほぼ同じエピソード長に収束した．

上記の各試行実験について，学習の結果得られたパラメータ（ルールの重み（方策勾配法），Q 値（Q 学習））を固定した上で，さらに 1 万エピソード分の行動計画実験を行った．ただし，方策勾配法および Q 学習共に温度はかなりの低温（T=0.01）とした．つまり，この実験は，10 万回の学習後に得られた方策が，決定論的にどういう計画を最良とするのかを，1 万個のデータを用いて調べるというものである．この実験の結果得られた 1 万エピソード分の平均エピソード長に関する 5 セット間の平均値は，方策勾配法が 5.5，Q 学習が 5.3 であった．このことからわかるように，両者は，本稿で定義した追跡問題についてほぼ同等の方策学習能力を持つといえる．

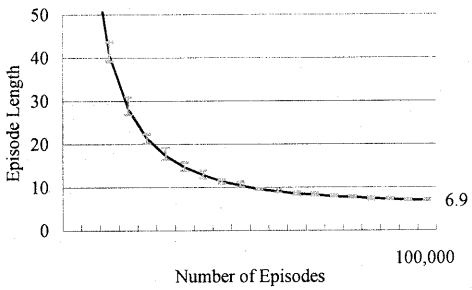

図 2 方策勾配法による学習.

Fig. 2 Learning by policy gradient method.

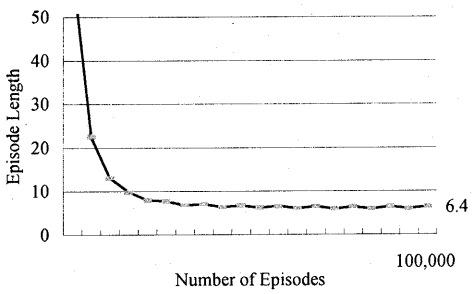

図 3 Q 学習による学習.

Fig. 3 Learning by Q-learning.

4.4. 実験 2：ポテンシャル項を追加して最短エピソードを求める場合

状態・行動ルールを用いた式(3.6)で表される目的関数に式(3.8)で表されるポテンシャル項を追加した目的関数,

$$E(a;s,\{\theta(s,a)\},\theta) \equiv -\sum_{s'}\left[\sum_{a'}\theta(s',a')\delta_{s,s'}\delta_{a,a'} - \theta U(s')\delta_{s',u(a;s)}\right] \tag{4.1}$$

の下, ルールの重み $\theta(s,a)$およびポテンシャル関数の重み θ を方策勾配法によって自律分散的に学習する実験を行った. これらのパラメータに対する学習則は, 式(3.7)および式(3.9)に示した通りである. 実験 1 同様, 目標達成時に $r=1/L^2$ の報酬値を全ハンターに与えることにより, より短いエピソードをもたらす方策の導出を目指した. また, ポテンシャル関数 $U(s)$については,

$$U(s) \equiv |\mathrm{X}-x| + |\mathrm{Y}-y| \tag{4.2}$$

とした. ここで, (x,y)および(X, Y)は, 状態 s におけるハンターの座標および獲物の座標をそれぞれ表す. 式(4.2)では, ハンターと獲物との間が近づくほどポテンシャルが低くなるように設定してある. つまり, このようなポテンシャル項を追加した場合, 遷移先におけるハンターと獲物との間が近い行動ほど選択されやすくなる.

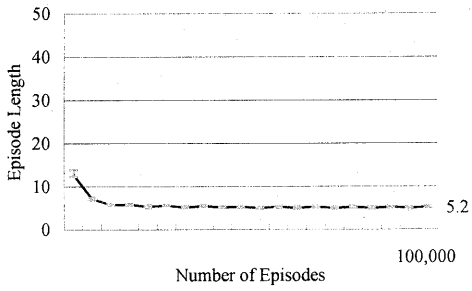

図 4 ポテンシャル項を使用した方策勾配法による学習.

Fig. 4 Learning by policy gradient method using a potential term.

ルールの重みとポテンシャル関数の重みの更新を 10 万エピソード繰り返す実験を, 実験 1 同様に計 5 セット行った. なお, 温度 T および学習係数 ε の値は実験 1 同様それぞれ 0.2 に設定した. また, ルールの重みとポテンシャル関数の重みの初期値についてもそれぞれ 0.1 とした.

エピソード長に関する 5,000 エピソード毎の平均値（平均エピソード長）の推移を図 4 に示す. ただし, 平均エピソード長に関する 5 セット間の平均値を折れ線で結ぶと共に, その最大値と最小値をエラーバーで表示した. 図 2 と図 4 との比較から, ポテンシャル項を追加した方が明らかに収束の速度が速いことがわかる.

上記の各試行実験について, 実験 1 同様, 学習の結果得られたパラメータ（ルール重み, ポテンシャル関数の重み）を固定し, さらに 1 万エピソード分の行動計画実験を温度 T=0.01 として行った. その結果得られた 1 万エピソード分の平均エピソード長に関する 5 セット間の平均値は 4.9 であった. このことから, 目的関数(4.1)の使用により, ポテンシャル項がない場合や一般的な Q 学習より短いエピソードを実現する方策を学習できたといえる.

4.5. 実験 3：エピソード長に時間窓制約がある場合

式(2.5)に示したように, 方策勾配法による学習では, 目標達成までの行動列$\{a(t)\}$に基づいて報酬を与える. そのため, エピソード長に時間窓制約のある場合にも容易に対応できる. ここでは, 例題として, 目標達成までのエピソード長が, $10<L<30$ のように制限された場合を考える. そこで,

$$r = \begin{cases} 1 & \text{if } \ 10 < L < 30 \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

のように時間窓制約のある報酬関数を使用し, 実験 1 同様に状態・行動ルールを用いた式(3.6)で表される目的関数の下, ルールの重み $\theta(s,a)$を方策勾配法によって自律分散的に学習する実験を行った.

ルールの重みの更新を 10 万エピソード繰り返す実験を，実験 1 同様に計 5 セット行った．なお，温度 T および学習係数 ε の値については，予備実験の結果に基づいて T=0.2，ε=0.1 と設定した．また，ルールの重みの初期値は実験 1 同様 0.1 とした．

エピソード長に関する 5,000 エピソード毎の平均値（平均エピソード長）の推移を図 5 に示す．ただし，平均エピソード長に関する 5 セット間の平均値を折れ線で結ぶと共に，その最大値と最小値をエラーバーで表示した．図 5 に示されるように，(10,30)という制約の範囲内にエピソード長を収束させる学習ができた．

上記の各試行実験について，実験 1 同様，学習の結果得られたパラメータ（ルール重み）を固定し，さらに 1 万エピソード分の行動計画実験を温度 T=0.01 として行った．その結果得られた 1 万エピソード分の平均エピソード長に関する 5 セット間の平均値は 27.7 であった．このことから，報酬関数(4.3)を用いた方策勾配法によって得られたルールの重みが，時間窓制約を満たす望ましい方策を与えることが確認できた．

5. 考察

4.3.の実験 1 における学習初期段階での収束の速度は，Q 学習の方が方策勾配法よりも速い(図 2 および図 3 参照)．この原因は，方策勾配法では平均報酬を使用し，Q 学習では割引報酬を使用したためと考えられる．つまり，平均報酬を用いる方策勾配法による学習では，学習初期段階における各ルールの重みの更新量が一様に微少となり，様々な方策を探索しようとする傾向がある．一方，割引報酬を用いる Q 学習では，目標状態に近い状態における行動選択が早期に絞られ，探索する方策の範囲が狭くなる傾向があると考えられる．

そこで，方策勾配法において，学習初期段階における収束の速度を上げる方法としては，目標状態に近い状態を初期状態として多く提示する方法と目的関数の項にポテンシャル型の知識を加える方法とが考えられる．前者は，学習データの与え方を工夫する方法である．一方，後者は，**3.4.**で示したように，方策のためのヒューリスティクスをポテンシャルの形で目的関数に追加する方法であり，その効果は **4.4.**における実験 2 の結果に示した通りである．このように，学習アルゴリズムを変更することなく，目的や制約に応じた項を目的関数に追加するだけで，ヒューリスティクスを行動計画に簡単に利用できることは，方策勾配法の大きな利点である．

また，状態と行動の履歴に対して報酬を与えることも可能であるので，エピソード長，運動機能や総移動距離に関する制約などにも柔軟に対応できる．例えば，エピソード長に時間窓制約がある場合は **4.5.**における実験 3 に示した通りである．

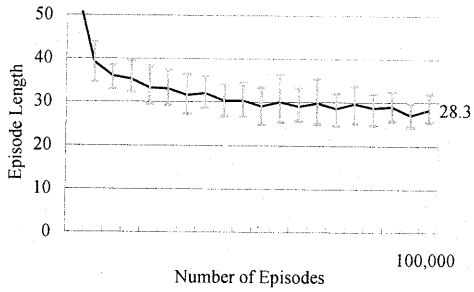

図 5 エピソード長に時間窓制約がある場合の方策勾配法による学習

Fig. 5 Learning by policy gradient method when the episode length should be longer than 10 shorter than 30.

6. まとめ

本研究では，マルチエージェント系における行動学習方式として，方策勾配法を採用し，自律分散方式における学習則を導出した．具体的には，オンライン型行動計画という仮定の下，ボルツマン分布による確率的方策を採用し，確率的勾配法に基づく方策関数中のパラメータ学習則を得た．次に，この学習則を，自律分散的なマルチエージェント系の行動学習方式に拡張した．さらに，状態の価値，状態・行動ルール，ポテンシャルという 3 種類の目的関数を方策として用いる場合の学習則を示した．

本方式の適用例として，追跡問題に対する 3 つの実験を行った．実験の結果，本方式によって良好な方策が得られることに加え，方策中への行動制約や目的の変更追加，状態と行動の履歴を必要とする制約，ヒューリスティクスの利用等に柔軟に対応できることを確認した．

文　献

[1] 三上貞芳, “強化学習のマルチエージェント系への応用,” 人工知能学会誌, vol. 12, no. 6, pp. 845-849, 1997.

[2] 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. 4, pp. 609-617, 1998.

[3] 荒井幸代, “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 - ,” 人工知能学会誌, vol. 16, no. 4, pp. 476-481, 2001.

[4] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” Machine Learning, vol. 8, pp. 229-256, 1992.

[5] 木村元, 山村雅幸, 小林重信, “部分観測マルコフ決定過程下での強化学習：確率的傾斜法による接近,” 人工知能学会誌, vol. 11, no. 5, pp. 761-768, 1996.