3F4-03

POMDPsへの行動優先度学習型強化学習アルゴリズムの適用

Applying Action-Preference Learning Algorithms to POMDPs

松井 藤五郎[*1] 犬塚 信博[*2] 世木 博久[*2]
Tohgoroh Matsui Nobuhiro Inuzuka Hirohisa Seki

[*1]東京理科大学 理工学部 経営工学科
Department of Industrial Administration, Faculty of Science and Technology, Tokyo University of Science

[*2]名古屋工業大学 大学院工学研究科 情報工学専攻
Department of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology

In the real-world problems, handling *uncertainty* is a key topic and *partially observable Markov decision processes* (POMDPs) provide a framework to autonomous agents with unreliable or incomplete sensors. Therefore, learning in POMDPs is very important for applying agents to practical engineering problems. This paper shows the experimental results with action-preference learning algorithms and Sarsa(λ) in three partially observable benchmark tasks. We confirmed that OnPS is the best among the methods for POMDPs.

1. はじめに

実世界環境においては,「不確実性」の扱いが重要である．エージェントの観測に不確実性を含む問題は，**部分観測マルコフ決定過程**（*Partially observable Markov decision processes*, POMDPs）の枠組みで扱うことができる．それゆえ，POMDPsで学習することは，実用的な工学問題へエージェントを応用するにあたって非常に重要である．

強化学習アルゴリズムの多くは,「状態行動対に対してその後に得られる割引収益の期待値として定義された行動価値を推定する」という行動価値推定型のアルゴリズムである [Sutton 98]. Q 学習 [Watkins 92] や Sarsa(λ) [Rummery 94] のような，行動価値学習型の手法は，状態行動対に対する真の行動価値を推定し，そこから最適な決定的政策を導く．

Loch と Singh [Loch 98] は，Sarsa(λ) を観測と行動の対に対して行動価値を推定するよう拡張し，いくつかの POMDPs においてその性能を調査した．そして，Sarsa(λ) が POMDPs においても有効に働くことを実験的に確認した．

一方，profit sharing [Grefenstette 88, Holland 86] のような，**行動優先度学習型アルゴリズム** [松井 03] も開発されている．行動優先度学習型手法は，行動価値ではなく，それぞれの行動の優先度を状態ごとに独立に保持する．

Arai らは，マルチエージェント環境においては，profit sharing が Q 学習よりも優れていることを示している [Arai 00]．また，Arai と Sycara は，POMDPs での学習のために，状態行動対への初回訪問だけを記憶する profit sharing である first-visit profit sharing（FVPS）を提案した [Arai 01]．**オンライン型** *profit sharing* (OnPS) と *last-visit profit sharing* (LVPS) も，同じ profit sharing の仲間であり，いくつかの環境でその有効性が確認されている [松井 03].

本論文の目的は，行動優先度学習型強化学習アルゴリズムの OnPS, LVPS, FVPS と行動価値推定型の Sarsa(λ) を，[Parr 95] や [Loch 98] でも使われた PODMPs の上で実験した結果を示し，比較することである．

連絡先: 松井藤五郎，東京理科大学 理工学部 経営工学科，〒 278-8510 千葉県野田市山崎 2641，Phone: 04-7124-1501（内線 3830），Fax: 04-7122-4566，E-mail: matsui@ia.noda.tus.ac.jp

2. POMDPs

有限 POMDPs の枠組みにおいては，エージェントが置かれる環境は $\langle \mathcal{S}, \mathcal{X}, \mathcal{A}, \mathcal{P}^a_{ss'}, \mathcal{O}^{ax}_s, \mathcal{R}^{ax}_{ss'} \rangle$ の組で表される．ここで，

- $\mathcal{S}$ は有限な状態の集合，
- $\mathcal{X}$ は有限な観測の集合，
- $\mathcal{A}$ は有限な行動の集合，
- $\mathcal{P}^a_{ss'} = \Pr(s_{t+1} = s' | s_t = s, a_t = a)$ は状態遷移確率（状態 $s \in \mathcal{S}$ において $a \in \mathcal{A}$ を取ったときに次の状態が $s' \in \mathcal{S}$ になる確率），
- $\mathcal{O}^{ax}_s = \Pr(x_{t+1} = x | s_{t+1} = s, a_t = a)$ は観測確率（行動 $a \in \mathcal{A}$ を取った後で状態 $s \in \mathcal{S}$ において観測 $x \in \mathcal{X}$ を受け取る確率），
- $\mathcal{R}^{ax}_{ss'} = \mathrm{E}(r_{t+1} | s_t = s, a_t = a, s_{t+1} = s', x_{t+1} = x)$ は報酬期待値（状態 $s \in \mathcal{S}$ において行動 $a \in \mathcal{A}$ を取った後で状態 $s' \in \mathcal{S}$ において観測 $x \in \mathcal{X}$ を受け取ったときの報酬の期待値）

を表す．強化学習においては，エージェントは，状態遷移確率 $\mathcal{P}^a_{ss'}$，観測確率 $\mathcal{O}^{ax}_s$，および報酬期待値 $\mathcal{R}^{ax}_{ss'}$ に関する知識を持たない．

POMDPs における学習手法の一つは，現在の状態を識別するために，観測情報を記憶するというものである．これらの手法は，メモリベース手法と呼ばれる．現在得ている観測が信頼できるものでない場合にも，観測のシーケンスを用いると，現在の状態を高い精度で推測することができる．それゆえ，たとえば [Loch 98] のように，メモリベース手法は，メモリなしの手法よりも優れた性能を示す．

しかしながら，実世界の問題においては，起こりうる観測シーケンスの数が非常に大きく，それらすべての観測シーケンスに対する状態へのマッピングを学習するためには，非常に多くの時間を費やさねばならない．したがって，メモリベース手法よりもメモリレス手法の開発に取り組むべきである．

一方，多くの POMDPs において，いかなる決定的政策（状態が与えられると，行動を一意に決定する政策）よりも優れた確率的政策（行動を確率的に選択する政策）が存在することが知られている [Kaelbling 96, Perkins 02].

すべての $s \in \mathcal{S}, a \in \mathcal{A}(s)$ に対して：
  $P(s,a) = C$ （C は任意の小さな正の定数）
各エピソードに対して繰り返し：
  s を初期化
  すべての s, a に対して：
    $c(s,a) = 0$
  エピソード中の各ステップに対して繰り返し：
    P から導かれた重み付きルーレット選択を用いて，
      s での行動 a を選択する
    $c(s,a) \leftarrow c(s,a) + 1$
    行動 a を取り，報酬 r と次状態 s' を観測する
    すべての s, a に対して：
      $P(s,a) \leftarrow P(s,a) + rc(s,a)$
      $c(s,a) \leftarrow \gamma c(s,a)$
    $s \leftarrow s'$
  s が終端状態ならば繰り返しを終了

図 1: オンライン型 profit sharing アルゴリズム．γ は割引率パラメータ．

3. 行動優先度学習型アルゴリズム

行動優先度学習型アルゴリズムとは，行動価値推定型でない強化学習アルゴリズムのことであり，行動価値の替わりにそれぞれの行動の優先度を状態ごとに独立に保持する [松井 03]．

ここでは，行動優先度学習型手法である OnPS と LVPS について述べる．

3.1 オンライン型 Profit Sharing

オンライン型 profit sharing (OnPS) [松井 03] では，優先度 P の要素数に等しい要素数の**信用トレース**（credit traces） c が，各状行動対に対する信用割当の度合いを表す．信用トレースは，適格度トレース（eligibility traces）と同様の働きをする．ただし，適格度トレースとは異なり，状態行動対に対する信用割当の度合いだけを記憶する．これにより，エピソードに現れた状態行動対をそのまま記憶する必要がなくなる．

各ステップにおいて，すべての状態行動対の信用トレースは γ だけ減り，そのステップで訪問された状態 s と選択された行動 a の対の信用トレース $c(s,a)$ は 1 増える．

$$c_t(s,a) = \begin{cases} \gamma c_{t-1}(s,a) + 1 & s = s_t \text{ かつ } a = a_t \text{ のとき} \\ \gamma c_{t-1}(s,a) & \text{そうでないとき} \end{cases} \quad (1)$$

オンライン型 profit sharing は，各ステップにおける優先度の増分を次式のように計算する．

$$\Delta P_t(s,a) = r_{t+1} c_t(s,a) \quad \text{for all } s, a \quad (2)$$

中間報酬がないとき，この更新式によるエピソードあたりの更新量は，オフライン更新型のものと等しい．このアルゴリズムを図 1 に示す．

オンライン型 profit sharing の特徴は，

- 中間報酬を扱える（従来のオフライン型 profit sharing では扱えない）
- 有限メモリで実装できる（従来のオフライン型 profit sharing では制限がない）

ことである．

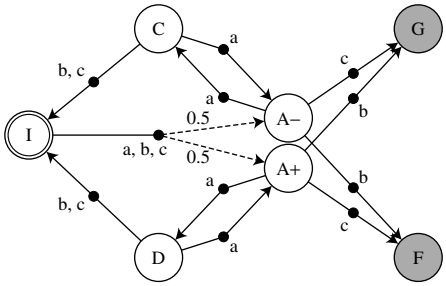

図 2: Parr と Russell の部分観測環境．状態 A+ および A− が，いずれも同じ A として観測される．実線で示された矢印の状態遷移確率は 1 である．

3.2 Last-Visit Profit Sharing

式 (1) は**累積トレース**（accumulating traces）と呼ばれる種類の適格度トレースを基にした式である．この累積トレースの代わりに，**入れ替え更新トレース**（replacing traces）と呼ばれる，次のようなトレースを用いることによって，オンライン型 profit sharing の性能向上を試みたものが *last-visit profit sharing*（LVPS）[松井 03] である．

$$c_t(s,a) = \begin{cases} 1 & s = s_t \text{ かつ } a = a_t \text{ のとき} \\ \gamma c_{t-1}(s,a) & \text{そうでないとき} \end{cases} \quad (3)$$

このトレースを用いると，ある状態行動対が選択されたとき，そのトレースの値がどのようなものであっても，その値を 1 にしてしまう．

これは，過去に選択された状態行動対と同じ対を再び選択した場合に，その対に関するそれまでの情報を捨てて最後に選択したときの情報だけを保持していることに相当する．つまり，冗長な経験をエピソードの記憶から取り除いている．この方法は，状態行動対への最後の訪問（last visit）だけを記憶する方法であることから，last-visit profit sharing と呼ばれる．

4. 実験結果

比較のため，三つの POMDPs を用いて実験を行った．OnPS, LVPS，初回の訪問だけを記憶するオフライン型 profit sharing である FVPS [Arai 01]，行動価値学習型の Sarsa(λ) [Rummery 94] を用いて実験を行った．Sarsa(λ) のパラメータは，同じ環境が用いられた [Loch 98] で採用された値を利用して $\lambda = 0.9$, $\alpha = 0.01$ とした．

学習中，profit sharing により学習するエージェントは重みつきルーレット選択を，Sarsa(λ) により学習するエージェントはボルツマンソフトマックス選択（$\tau = 0.2$）を用いた．これらの確率的政策と，決定的政策であるグリーディ選択を用いたときの性能を測定した．すべての実験において，30 回の実験の平均を取ることによって，学習曲線を滑らかにした．

4.1 Parr と Russell の POMDP

まず，図 2 に示された，Parr と Russell の部分観測環境 [Parr 95] を用いた．二つの終端状態 G と F を含む七つの状態があり， 初期状態は I で示されている．

エージェントは，状態 A+ と A− を識別することができない．エージェントの行動は，a, b, c の三種類である．初期状態

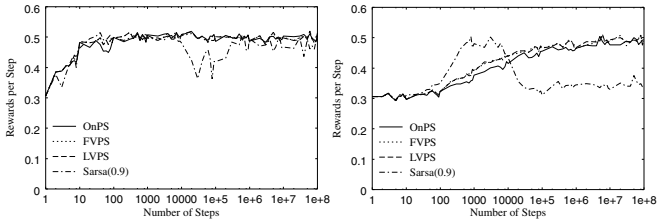

図 3: Parr と Russell の POMDP における結果．左：決定的政策の性能．右：確率的政策の性能．

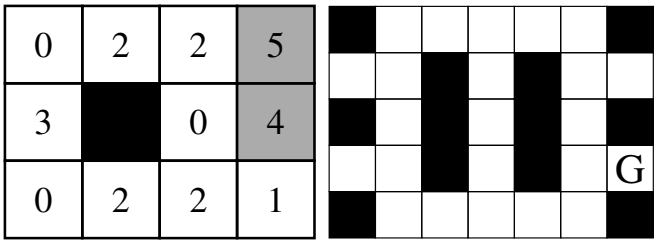

図 4: 左：Parr と Russell の 4×3 格子世界. 同じ番号のマスは同じ観測となる．灰色のマスは，終端状態である．右：Littman らの 89 状態オフィス．G のマスが目標状態である．

からの状態遷移だけが確率的で，そのほかの遷移は決定的である．

エージェントは，状態 G にたどり着くと，報酬 2 を得る．状態 F は，G と同じ終端状態である．ただし，F にたどり着いても，報酬は得られない．この問題における割引率は 0.95 である．

結果を，図 3 に示す．ここでは，[Loch 98] と同じ方法で，各時点において 101 回の試行を行い，性能を測定した．一回の試行が 101 ステップを超える場合には，そこで打ち切った．左のパネルは決定的政策の性能を，右のパネルは確率的政策，すなわち，重みつきルーレット選択あるいはソフトマックス選択の性能を示している．学習中，エージェントは確率的政策を用いて行動するため，右のパネルは，学習中の性能を表している．

Profit sharing の三つの手法は，どれも同じような性能を示している．しかし，Sarsa(0.9) だけは，確率的政策の性能，すなわち，学習中の性能が悪い．これは，観測 A において，行動 b と c の価値が $(2+0)/2 = 1$ であるのに対し，回り道になっている行動 a の価値は $0.95^2 = 0.9025$ と，行動の価値が大きく変わらないことが原因である．これらの行動価値から導かれたソフトマックス選択の行動選択確率は，どの行動も大きく変わらない．したがって，Sarsa(0.9) の性能が悪くなっている．

一方，profit sharing においては，行動優先度の差は，学習が進むにつれて広がっていく．したがって，profit sharing は，行動価値に差がない環境においても，それに苦しめられることはない．

4.2 Parr と Russell の 4×3 格子世界の問題

続いて，図 4 の左のパネルに示された，Parr と Russell の 4×3 格子世界 [Parr 95] を用いて実験した．ここには，灰色で示された二つの終端状態を含む 11 の状態がある．初期状態は，

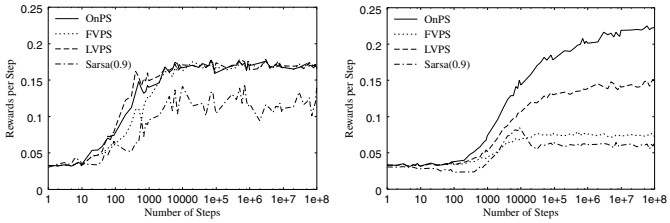

図 5: Parr と Russell の 4×3 格子世界の結果．

非終端状態から一様なランダムに選択される．

エージェントは，左右の壁があるかどうかだけを観測することができる．したがって，左右の壁の有無の組み合わせに対応した四種類の観測と，終端状態を識別するための二つの観測が存在する．観測は決定的である．すなわち，エージェントが誤った観測をすることはない．

エージェントの行動は，上下左右のいずれかへ移動する四種類である．状態遷移は決定的でなく，エージェントが望んだ方向に 0.8 の確率で移動し，その方向に対して 90 度の方向にそれぞれ 0.1 の確率で遷移する．

エージェントは，右上のマス（5 番）に到達したときに 2 の報酬を得る．割引率は $\gamma = 0.95$ である．

図 5 に結果を示す．先の実験とまったく同じ方法で性能を評価した．Sarsa(0.9) の確率的政策の学習曲線は，先の実験と同様の形をしている．Profit sharing の決定的政策の性能は，OnPS, LVPS, FVPS のいずれも大差ないが，確率的政策の性能は大きく異なる．確率的政策では OnPS が最も優れた性能を示し，続いて LVPS が良い性能を示した．FVPS は Sarsa(0.9) をわずかに上回ったが，Sarsa(0.9) の決定的政策よりも悪い性能となった．

これらの結果は，決定的政策よりも優れた確率的政策が存在することを示している．OnPS の重みつきルーレット選択の性能は，これら手法が獲得したどの決定的政策よりも優れている．

4.3 Littman らの 89 状態オフィスの問題

最後に，図 4 の右のパネルに示した，Littman らの 89 状態オフィス [Littman 95] を用いて実験を行った．この問題では，位置に加えて，エージェントの向きも考慮される．位置は 22 ある．各位置につき四つの方向があるので，可能な状態の数は，$22 \times 4 = 88$ に G で示された目標状態一つを加えた 89 となる．初期状態は，目標状態をのぞくすべての状態から，一様なランダムに選択される．

エージェントには，壁の有無を調べるセンサが，四方に独立についており，16 通りの観測が得られる．ただし，これらのセンサは，壁がある場合には 0.9 の確率で，そうでない場合には（間違って）0.05 の確率で，壁を観測する．

エージェントは，前進，右方向転換，左方向転換，逆方向転換，停留の五つの行動を取ることができる．状態遷移は決定的でない．たとえば，前進するとき，エージェントが望んだマスには 0.8 の確率で移動する．報酬は，目標状態に到達したときに 1，それ以外は 0 である．

結果を図 6 に示す．この実験では，[Littman 95] および [Loch 98] の実験方法に合わせ，一試行を最大 251 ステップとして 251 回の試行を行い，目標状態へ到達した割合と目標状態へ到達するのに要したステップ数を計測した．

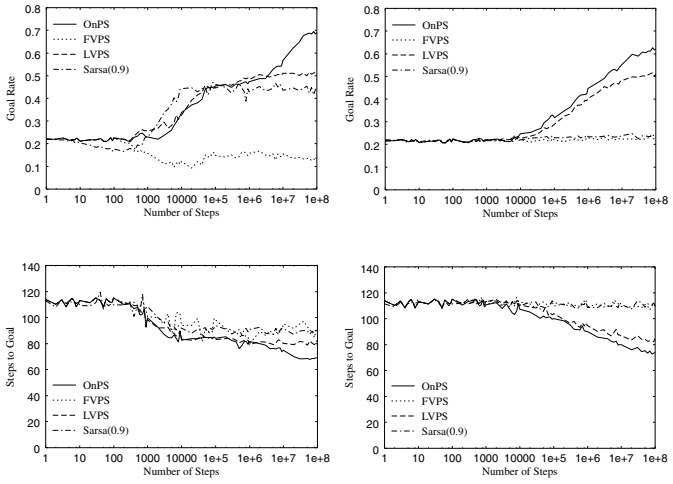

図 6: Littman らの 89 状態オフィスの結果．上：目標状態へ到達した割合．下：目標状態へ到達するのに要したステップ数．

OnPS は，いずれの評価尺度においても，他の手法より優れた性能を示した．LVPS は，OnPS の次に良い性能を示した．FVPS の決定的政策の性能は，一様なランダムに行動を選択したときよりも悪くなった．

5. まとめ

本論文では，行動優先度学習型アルゴリズムの OnPS, LVPS, FVPS と行動価値推定型アルゴリズムの Sarsa(λ) をいくつかの POMDPs において比較した．行動価値推定型アルゴリズムは，正しい価値を推定したとしても，そこから適切な確率的政策を導くことができない．それに対し，行動優先度学習型アルゴリズムは，報酬の獲得につながった行動に対する優先度を増加させていくため，優先度が同じような値に収束することがなく，優れた確率的政策を導くことに成功している．

行動優先度学習型の中でも，OnPS は，最も優れた性能を示した．LVPS は，決定的環境では OnPS よりも有効に働くが，確率的環境では OnPS に劣る．FVPS は，「目標へ到達するために通らなければならない異なる状態が同じ観測となり，かつ，それらの状態においては異なる行動を取らなければならない」ような場合において，正しく学習することができない．

以上のことから，POMDPs においては，OnPS がこれらの手法の中で最も優れているといえる．

しかしながら，OnPS や LVPS は，いくつかの欠点を抱えている．一つは，行動選択確率が負にならないようにするため，負の報酬を扱えないことである．これに対して，筆者らは，[鈴木 03] においてその改善を試みている．

もう一つは，学習が進むと，行動優先度が無限大へと発散することである．理論的には，優先度をエージェントが経験したステップの総数で割る，すなわち，ステップあたりに獲得した信用割当を優先度とすることによって，この問題を回避することができる．ただし，行動優先度は変化し続けるため，最適解へと収束することは保証されない．

今後は，OnPS と LVPS の理論的な解析を行う必要がある．

参考文献

[Arai 00] Arai, S., Sycara, K., and Payne, T. R.: Experience-Based Reinforcement Learning to Acquire Effective Behavior in a Multi-agent Domain, *Proc. of The 6th Pacific Rim Int'l Conf. on AI*, pp. 125–135, Springer-Verlag (2000)

[Arai 01] Arai, S. and Sycara, K.: Credit Assignment Method for Learning Effective Stochastic Policies in Uncertain Domains, in *Proc. of The Genetic Evolutionary Computation Conf. 2001*, pp. 815–822 (2001)

[Grefenstette 88] Grefenstette, J. J.: Credit assignment in rule discovery systems based on genetic algorithms, *Machine Learning*, Vol. 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 *Machine Learning: An Artificial Intelligence Approach*, Vol. 2, Morgan Kaufmann Publishers (1986)

[Kaelbling 96] Kaelbling, L. P., Littman, M. L., and Moore, A. W.: Reinforcement Learning: A Survey, *Journal of AI Research*, Vol. 4, pp. 237–285 (1996)

[Littman 95] Littman, M. L., Cassandra, A. R., and Kaelbling, L. P.: Learning policies for partially observable environments: Scaling up, Technical Report CS-9-11 (1995)

[Loch 98] Loch, J. and Singh, S.: Using Eligibility Traces to Find the Best Memoryless Policy in Partially Observable Markov Decision Processes, *Proc. of The 15th Int'l Conf. on Machine Learning*, pp. 141–150, Morgan Kaufmann Publishers (1998)

[松井 03] 松井 藤五郎：自律型エージェントの行動学習に関する研究, 名古屋工業大学学位審査論文 (2003)

[Parr 95] Parr, R. and Russell, S.: Approximating Optimal Policies for Partially Observable Stochastic Domains, in *Proc. of The 14th Int'l Joint Conf. on AI*, pp. 1088–1094, Morgan Kaufmann Publishers (1995)

[Perkins 02] Perkins, T. J. and Pendrith, M. D.: On the existence of fixed points for Q-learning and Sarsa in partially observable domains, in *Proc. of The 19th Int'l Conf. on Machine Learning*, pp. 490–497, Morgan Kaufmann Publishers (2002)

[Rummery 94] Rummery, G. and Niranjan, M.: On-Line Q-Learning Using Connectionist Systems, Technical Report CUED/F-INFENG/TR166 (1994)

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

[鈴木 03] 鈴木 淳司, 松井 藤五郎, 世木 博久：罰を考慮した Profit Sharing 強化学習法, 2003 年度人工知能学会全国大会（第 17 回）論文集, 3F4-02 (2003)

[Watkins 92] Watkins, C. J. C. H. and Dayan, P.: Technical Note: Q-Learning, *Machine Learning*, Vol. 8, No. 3/4, pp. 279–292 (1992)