G-7 Profit Sharing 強化学習法の適格度トレースに基づくオンライン実装
On-Line Implementation of Profit Sharing Based on Eligibility Traces

松井藤五郎* 犬塚信博† 世木博久‡
Tohgoroh Matsui Nobuhiro Inuzuka Hirohisa Seki

1 はじめに

強化学習は，試行錯誤を通じた学習の方法であり，自律型ロボットの行動学習法として広く用いられている．強化学習アルゴリズムのひとつである **Profit Sharing** は，状態行動対のユニークな価値が特定できない不完全知覚領域においても有効な確率的政策を獲得できることから，マルチエージェント環境などでの利用が可能なことが知られている [1].

しかし，従来の Profit Sharing は，選択した状態行動対をエピソード終了後に一括して強化するオフライン更新方式であり，選択したすべての状態行動対を記憶していくため必要とするメモリ量に上限がない——という点が問題である．そこで，本論文では，**適格度トレース** [5] の技法的な側面に着目し，Profit Sharing をオンライン更新方式で実装する方法を提案する．トレースには状態行動対の数と同数の記憶領域を使用するので，本手法で必要とされるメモリ量は一定である．本論文では，累積トレースと入れ替え更新トレースの二つの種類のトレースに基づいた実装方法を提案する．

また，よく知られている**車の山登り問題** [5] を用いた実験の結果を示し，本手法の有効性と従来の強化学習法との違いについて議論する．

2 アルゴリズム

従来の Profit Sharing は，エピソード終了後，エピソードに含まれる状態行動対 s_t, a_t に対して，次式を用いてその評価値としての重み W を更新する．

$$W(s_t, a_t) \leftarrow W(s_t, a_t) + f(t, r_T, T) \tag{1}$$

ここで，f は強化関数と呼ばれる関数であり，多くの場合，次式の等比減少関数が用いられる．

$$f(t, r_T, T) = \beta^{T-t-1} r_T \quad (0 \le \beta \le 1) \tag{2}$$

適格度トレース [5] は強化学習の基本的なメカニズムのひとつである．本手法は，トレースの値が，対応する状態行動対がその時刻においてどの程度学習上の変化を受けるべきかを表す——ということを利用し，選択した状態行動対を記憶することなく式 (2) の値を計算する．この式では，一つ前のステップの値には一つ多くの β がかけられている．そこで，本手法では，各ステップにおいて，すべての状態行動対のトレースに β をかけ，そのステップで選択した状態行動対のトレースを 1 増やす．これを形式的に表すと次のようになる．

$$e_t(s,a) = \begin{cases} \beta e_{t-1}(s,a) + 1 & s = s_t \text{ かつ } a = a_t \text{ のとき} \\ \beta e_{t-1}(s,a) & \text{そうでないとき} \end{cases} \tag{3}$$

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
  $W(s,a) \leftarrow$ a small constant
Repeat (for each episode):
  Initialize s
  $e(s,a) \leftarrow 0$, for all s, a
  Repeat (for each step of episode):
    Choose a from s using a policy derived from W
      (e.g., roulette)
    $e(s,a) \leftarrow e(s,a) + 1$ /* 累積トレース[1]*/
    Take action a; observe reward, r, and next state, s'
    For all s, a:
      $W(s,a) \leftarrow W(s,a) + re(s,a)$
      $e(s,a) \leftarrow \beta e(s,a)$
    $s \leftarrow s'$
  until s is terminal

図 1: 累積トレースを用いたオンライン更新型 Profit Sharing アルゴリズム.

次に，各ステップの重み更新量を次式のように求める．

$$\Delta W_t(s,a) = r_{t+1} e_t(s,a) \quad \text{for all } s, a \tag{4}$$

このアルゴリズムを図 1 に示す．

式 (3) は**累積トレース**と呼ばれる種類のトレースを基にした式である．別の強化学習アルゴリズムの Sarsa(λ) では，**入れ替え更新トレース**と呼ばれる，わずかに異なったトレースを用いることで性能が改善される場合がある [4]．このトレースでは，選択された状態行動対のトレースを 1 に再設定する．これは，Profit Sharing において，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することに相当する．そこで，このトレース更新の式を次のように表す．

$$e_t(s,a) = \begin{cases} 1 & s = s_t \text{ かつ } a = a_t \text{ のとき} \\ \beta e_{t-1}(s,a) & \text{そうでないとき} \end{cases} \tag{5}$$

本論文では，式 (5) のトレースを用いたオンライン更新型 Profit Sharing を Last-Visit Profit Sharing (LVPS) と呼ぶ．

3 実験結果

本手法の有効性を確認するため，**車の山登り問題** [5] を用いて実験を行った．この問題では，状態が位置と速度という二つの連続値パラメータで表現される．連続的な状態変数を扱

*名古屋工業大学 大学院 工学研究科 電気情報工学専攻
†名古屋工業大学 工学部 電気情報工学科
‡名古屋工業大学 工学部 知能情報システム学科

[1] 入れ替え更新トレースを用いるには，この行を $e(s,a) \leftarrow 1$ に置き換える．

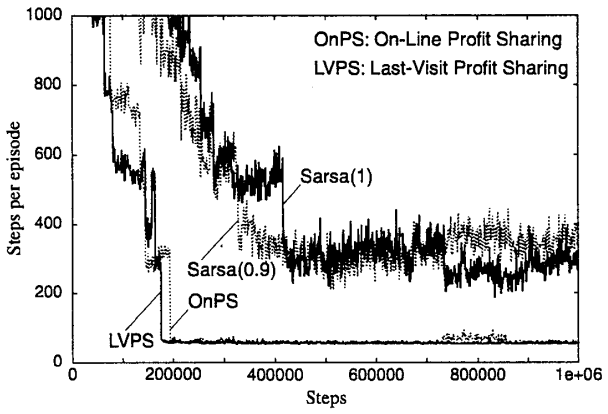

図 2: 車の山登り問題の結果（30 回の平均）.

うため，9×9 のタイリングを 10 枚重ねた線形関数近似を用いた [3]. β は，予備実験に基づいて経験的に選択し，$\beta = 0.9$ とした．重みの初期値は 0.0001 とし，報酬はゴールに到達するまでを 0，到達したときに 1 とした．学習中の政策は重み付きルーレット選択 $P(a_t|s_t) = W_t(a_t, s_t)/\sum_a W_t(a, s_t)$ を用い，グリーディ政策により学習した重みの評価を行った．エピソード長が 10,000 に達したときにエピソード，初期状態，適格度トレースの初期化を行った．評価は，タスクを 100 回実行してその平均エピソード長を求めることにより行った．

比較のため，$\lambda = 1$ および 0.9 の入れ替え更新トレース Sarsa(λ) を用いて実験した．学習中の政策は $\epsilon = 0.01$ の ϵ グリーディ政策を用いた．そのほかのパラメータは $\alpha = 0.001$, $\gamma = 0.9$ とした．これらの値は [5] を参考に，いい結果を示したものを選んだ．

これらの結果を図 2 に示す．提案手法は，いずれも Sarsa(λ) より早く，いい政策を学習できた．入れ替え更新トレースに基づいた実装である LVPS は，累積トレースを基にした OnPS よりもわずかではあるが早く学習できた．また，本手法が最終的に獲得した政策の性能は，トレースの種類が異なってもほとんど差がなかった．

4 議論

従来の Profit Sharing では，エピソード途中の報酬を 0 に限定している．このとき，式 (3), (4) によるオンライン更新方式のエピソードあたりの更新量は，式 (1), (2) に示されたオフライン更新方式のものと等しい[2]．また，従来のオフライン更新型 Profit Sharing はエピソードが終了するまで何も学習しないが，オンライン更新型 Profit Sharing はエピソードの途中に発生する報酬から学習することができる．この点において，本手法は従来手法より優れている．

オンライン更新方式では，各ステップにおける計算コストが余分にかかるが，従来の適格度トレースにおける技法 [5] を応用し，非ゼロトレースの値を持つ状態行動対だけを記憶することにより計算上の手間を大幅に減らすことが可能である．

[2]証明は [3] を参照されたい．

Profit Sharing は，ある程度いい解を学習できるが，それが最適解であることを保証しない．それに対し，Q 学習や Sarsa(1) などの手法は，学習時間を無限大としたとき MDPs（マルコフ決定過程）環境において最適解に収束することが保証されることから，優れた手法であると主張されてきた．しかしながら，多くの工学応用的環境が MDPs でないことや，たとえ環境が MDPs であったとしても Q 学習や Sarsa(1) は学習に非常に多くの時間ステップを必要とすることから，少ない時間ステップである程度いい解を学習できる本手法は有効であると考える．

Q(λ) や Sarsa(λ) など適格度トレースを用いた従来の手法は，1 ステップ後の報酬をそのとき選択した状態行動対の価値だけに反映させていた手法に対し，最近選択した状態行動対の価値にも反映させるものである．本手法は，適格度トレースにおけるトレースの値が，最近選択した状態行動対が学習上の変化を受けることの適格度を保持していることに着目し，Profit Sharing におけるエピソードの記憶を適格度トレースに基づいて実現している——という点で従来の手法とは大きく異なる．

本論文では，累積トレースと入れ替え更新トレースという二種類のトレースに基づく実装方法を示した．入れ替え更新トレースを基にした実装である LVPS は，Arai と Sycara の FVPS (First-Visit Profit Sharing) [2] と似た仕組みを持つ．FVPS は，一度選択し記憶した状態行動対は再び選択してもエピソードに加えないことによってエピソード中にループ系列が生じることを防ぎ，不完全知覚下（部分観測 MDPs, POMDPs）でも適切な政策を学習することに成功している．また，各状態行動対は多くても 1 回しか記憶されないので，必要とされるメモリ量には上限がある．LVPS も，再び同じ状態行動対を選択した場合には最後に選択したものだけが記憶されており，FVPS と同様にエピソード中のループ系列を解消している．したがって，LVPS は POMDPs 環境でも有効に働くと期待できる．FVPS はオフライン更新方式であり，LVPS はオンライン更新方式である——という点でこの二つの手法は大きく異なる．

今後は，本手法の有効性を理論的に示すことが課題である．また，POMDPs 環境での有効性やエピソード途中に学習できることの有効性を実験により確認する必要がある．

参考文献

[1] 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. 467–481, 2001.

[2] Sachiyo Arai and Katia Sycara. Credit assignment method for learning effective stochastic policies in uncertain domains. In *Proceedings of the Genetic Evolutionary Computation Conference 2001*, pp. 815–822, 2001.

[3] 松井藤五郎, 犬塚信博, 世木博久. 線形関数近似を用いた profit sharing 強化学習法. 2002 年度人工知能学会全国大会（第 16 回）論文集, 2D3-03, 2002.

[4] Satinder P. Singh and Richard S. Sutton. Reinforcement learning with replacing eligibility traces. *Machine Learning*, Vol. 22, pp. 123–158, 1996.

[5] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 1998. 三上貞芳, 皆川雅章 共訳. 強化学習. 森北出版, 2000.