知能と複雑系 119−16
(2000. 1. 13)

強化学習における環境変化認識法

山本 真也，山口 文彦，斎藤 博昭，中西 正和

慶應義塾大学大学院 理工学研究科
〒 223-8522 神奈川県横浜市港北区日吉 3-14-1

TEL: 045-563-1141

Email: {yamamo, yamagu, hxs, czl}@nak.ics.keio.ac.jp

あらまし　非マルコフ決定過程 (non-MDP) の環境における強化学習の問題点の解決法として，環境変化時に何らかの処理を行う方法が提案されている．これらの研究において，環境変化の認識法は確立されていない．本論文では，non-MDP における有力な学習エンジンである確率的傾斜法において，学習中に環境変化を認識する方法を提案する．確率的傾斜法の内部変数 W の変化量を調べることにより環境変化を認識する．提案手法は確率的傾斜法が適用できる問題であれば簡単に内部に組み込むことのでき，環境変化の認識を行うことができる．シミュレーション実験により従来の手法の半分程度のステップで環境変化を認識できることを示す．

キーワード　　強化学習，確率的傾斜法，環境変化

A recognization method of environmental change on reinforcement learning

Shinya Yamamoto, Fumihiko Yamaguchi, Hiroaki Saito, Masakazu Nakanishi

Keio University 3-14-1, Hiyoshi, Kouhoku-ku, Yokomama, Kanagawa 223-8522 Japan

TEL: 045-563-1141

Email: {yamamo, yamagu, hxs, czl}@nak.ics.keio.ac.jp

Abstract　There are some methods that resolve problems of reinforcement learning in non Marokov Decision Process (non-MDP) environment on environment changes. The efficient method of recognizing environmental change has not yet been proposed. This paper proposes a method for recognizing environmental changes on Stochastic Gradient Ascent (SGA) which is a major learning engine in non-MDP environment. It uses the change of an internal variable W of SGA. Our method can be easily put in SGA and it is available for all SGA-applicable problems. We had a simulation to show the efficiency of our method and succeeded to reduce the recognition time to almost half of the conventional method.

keywords　　reinforcement learning, Stochastic Gradient Ascent, environmental change

1 はじめに

従来はマルコフ決定過程 (MDPs) の環境における自律的ロボットのための学習エンジンとして提案された強化学習法を，マルコフ決定過程でない (non-MDPs) 環境や部分観測マルコフ決定過程 (POMDPs) の環境へ適用しようという試みが近年行われている．MDPs の環境においては，Q-learning[Watkins92] などの最適解を保証する手法も存在している．しかし，non-MDPs の環境においては大きく分けて三つの問題点が挙げられている．第一に，全ての状態と行動の組合せに対して Q 値や強化値のような評価値を保持していると非常に大きなメモリを必要とすること．第二に，エージェントが新しい環境に出会うと，それらについての知識の不足から，ゼロから学習しなくてはならないので，学習に非常に長い時間がかかること．第三に，様々な環境において与えられたタスクを完成させるために，エージェントは個々の環境ごとに政策を持たなくてはならないので，大きなメモリを必要とすること．

第一の問題点を解決するために，確率的傾斜法が提案されている．これについては，第 2 章で述べている．第二の問題点を解決するために，あらかじめ変化しにくい知識をしまっておくといった手法が提案されている [Tanaka97] が，環境の数に応じたメモリが必要となり，第三の問題点が解決されていない．第三の問題点を解決するために，環境変化時に前の環境で学習した政策を新しい環境に変換する手法などが提案されている [Minato98]．この手法では，環境の変化の認識の重要性に触れているにもかかわらず，そのための手段についてはタスクの成功率の減少といった単純な方法を用いている．

そこで，本論文ではエージェントが学習中に環境の変化を認識する手法を提案する．エージェントが環境の変化を素早く認識することができれば，環境変化の際により早く新しい環境に適応することができるようになると考える．

以下，2 章では確率的傾斜法について述べる．3 章では環境変化の認識法について述べる．4 章ではシミュレーション環境およびその結果を述べ，5 章で結果に関する考察などを行う．

2 確率的傾斜法

強化学習には様々な学習法が存在し，MDPs の環境においては最適政策を学習することができるものも存在するが，non-MDPs の環境においては学習が収束しない，非常に大きなメモリを必要とするなどの問題がある．確率的傾斜法 [Kimura95][木村 96] はこのような問題を解決するために提案された強化学習アルゴリズムであり，non-MDPs の環境における有力な学習エンジンとなっている．そこで，本研究では確率的傾斜法を強化学習の学習アルゴリズムとして採用する．以下に，そのアルゴリズムと特徴を挙げる．

2.1 アルゴリズム

図 1 に確率的傾斜法のアルゴリズムを示す．

1. 環境の観測 X_t を受け取る．
2. $\pi(a, W, X)$ の確率で行動 a_t を実行する．
3. 環境から報酬 r_t を受け取る．
4. 内部変数 W の全ての要素 w_i について以下の $e_i(t)$ と $\overline{D_i}(t)$ を求める．ただし，γ は割引率 $(0 \leq \gamma \leq 1)$ である．

$$e_i(t) = \frac{\partial}{\partial w_i} \ln\{\pi(a, W, X)\} \qquad (1)$$

$$\overline{D_i}(t) = e_i(t) + \gamma \overline{D_i}(t-1) \qquad (2)$$

5. 以下の式を用いて $\Delta w_i(t)$ を求める．

$$\Delta w_i(t) = (r_t - b)\overline{D_i}(t) \qquad (3)$$

但し b は定数である．

6. 政策の改善: 以下の式で W を更新する．

$$\Delta W(t) = (\Delta w_1(t), \Delta w_2(t), \cdots, \Delta w_i(t), \cdots)$$

$$W \leftarrow W + \alpha \Delta W(t) \qquad (4)$$

但し α は非負の学習定数である．

7. 時間ステップ t を $t+1$ へ進めて，1 へ戻る．

図 1: 確率的傾斜法のアルゴリズム

2.2 特徴

以下に確率的傾斜法のアルゴリズムの特徴を述べる．

- 各時間ステップでの状態や割引報酬の期待値などを明示的に推定するような，計算コストのか

かる処理が不要である．

- 確率的政策 $\pi(a, W, X)$ がエージェントの内部変数 W を用いて関数表示されているため，任意の関数近似システムを用いることが可能である．これらを用いれば連続値の観測も扱える．
- 政策 $\pi(a, W, X)$ を行動 a を出力する確率密度関数とすれば，連続値の行動を扱うことができる．

3 環境変化認識法

本研究では，前述のアルゴリズムの内部変数 W の変化を調べることにより環境変化の認識を行う．タスクの成功率の変化から環境変化の認識を行う場合には，成功率を算出するための一定数の試行の後でしか認識を行えない．これと比較した場合 W は毎ステップ更新されるため，一定ステップごとに W の変化を調べればより早く環境の変化を認識できることが期待できる．以下に環境変化認識法を提案し，どのようにして評価を行うかを述べる．

ここで，ステップとはエージェントが環境の観測を受け取ってから行動を選択，実行し，政策を更新するまでのことを指す．試行とは，エージェントがあるタスクを開始してから目的を達成 (報酬を獲得) するか，失敗するまでのステップの集合を指す．

3.1 提案手法

ここでは，W の変化を調べる方法として手法 1 から手法 3 の 3 つの手法を提案する．いずれの手法も一定区間ごとの変化量を調べているが，これは以下の理由による．

W は学習の初期には大きく変化するが，学習が安定してくると変化量が小さくなる．変化量が小さくなっても，行動選択時のランダムネスなどから，多少振動しているので，毎ステップごとの変化量を調べるだけでは，このようなランダムネスに対応することができない．よって，一定区間の変化量をみるべきであると言える．

3.1.1 手法 1

W の各要素は毎ステップ更新されるが，報酬が得られた場合にはその他の場合よりも大きく更新される．そのため，一定ステップごとの変化量を調べた場合には報酬の獲得がその一定区間に含まれているかどうかで大きく変化量が異なる．ある区間では報酬が獲得されず，次の区間では数回報酬が獲得された場合などは，最初の区間の W と次の区間とで大きな差があり，実際には環境が変化していないのに変化したとしてしまう，誤認識が増えると考えられる．そこで，W の各要素の一定試行分の分散を算出し，その平均の 99% 信頼区間を構成し，次の一定試行の分散の平均と比較する．

3.1.2 手法 2

W の各要素を 2 次元のベクトルとして扱い，ある時点から一定試行前までのベクトルとその一つ前までの区間のベクトルの角度を算出し，全ての要素の平均を算出する．これを一定数保持しておき，99% 信頼区間を構成し，次の一定区間と比較する．ベクトルの角度は内積を用いて算出する．

3.1.3 手法 3

手法 2 では，一定試行ごとのベクトルの角度の比較を行うが，手法 3 では，一定区間ごとのベクトルの角度の比較を行う．環境が大きく変化した直後は一試行あたりのステップ数が大きくなるので，その間は環境変化直前と比較すると W の変化量が大きく異なると考えられる．また，環境変化直後は一試行あたりのステップ数が大きくなるので，一試行終了前に変化を認識することが期待でき，手法 2 よりも早く認識できると考えられる．

3.2 評価方法

上記の 3 つの方法とタスクの成功率を用いる方法のそれぞれにおいて，環境が変化してから変化を認識するまでのステップ数を比較することで評価を行う．タスクの成功率を用いる手法 (以後，手法 0 とする) と手法 1,2 は結果が試行数で得られるので，比較のためにステップ数に変換する．タスクの成功率を求めるために，直前の (1) n 試行のステップ数と，(2) 更にその n 試行前までのステップ数を用いる．(2) の 99% 信頼区間を構成し，(1) の平均と比較したときに，信頼区間に含まれない場合に環境変化を認識するものとする．$n = 5, 10, 20$ の 3 通りについて実験を行った結果，$n = 10$ の場合が最も良い結果となったので，比較対象としては，$n = 10$ の場合のタスクの成功率の変化を用いる．また、それぞれの手法で環

境変化の認識率及び誤認識率についても比較し，評価を行う．

4 実験

実験環境としては，8×8 の迷路問題を用いる．エージェントは決められたスタート地点からゴール地点への道のりを学習する．迷路にはいくつかの障害物が存在し，障害物のある場所へは移動することができない．スタート地点からゴール地点へ到達する道筋が必ず存在するものとする．一定試行数学習の後に迷路を変化させ，これを環境変化とする．以下，エージェントの政策の構成，実験環境，実験結果を述べる．

4.1 政策の構成

学習率 α は，予備実験の結果 0.05 を用いることとした．また，[Kimura97] に従い，$\gamma = 0.9, b = 0.01$ とした．エージェントの視界は，自分の周囲 8 マスとし，毎ステップごとに各マスに対して 0 (移動可能) または 1 (壁または，障害物) が入力として与えられる．出力は 4 種類とし，これを sigmoid 関数を通した後に，正規化してそれぞれを，上下左右の 4 方向への行動選択確率として用いる．よって，政策 $\pi(a, W, X)$ における W は 32 次元のベクトルとなる．図 4.1 にエージェントの内部構造を示す．

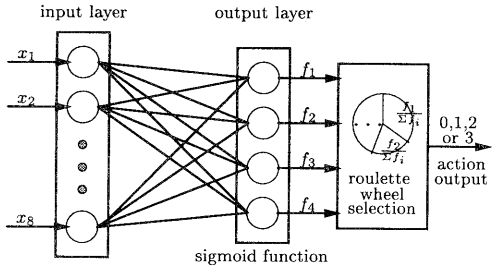

図 2: エージェントの内部構造

$x_1, x_2, \cdots, x_8$ は環境からの観測を受け取る入力層であり、その入力に対してエージェントが実行できる個々の行動の評価値を sigmoid 関数でフィルタリングして出力する層が $f_1, f_2, \cdots, f_4$ である．任意の i 番目の入力ユニットと，その入力に対する任意の j 番目の sigmoid 関数ユニットを結ぶ内部変数 (重み) を w_{ij} と表している．評価値 f_i は入力ユニット x_i と内部変数 w_{ij} の加重和をとり，以下のように計算される．

$$f_j = \frac{1}{1 + \exp\left(-\sum_i x_i w_{ij}\right)} \tag{5}$$

それぞれの評価値 f_j に比例した確率で，f_j に割り当てられた行動 a_j を一つ選択する．よって，政策 π は以下のように表される．

$$\pi(a = j, W, X) = \frac{f_j}{F} \tag{6}$$
$$\text{ただし，} \quad F = \sum_k f_k$$

行動 a_s を選択した場合の e_{ij} は式 (6) と，式 (1) より以下のように表される．

$$\begin{aligned} e_{ij} &= \frac{\partial}{\partial w_{ij}} \ln\{\pi(a_s, W, X)\} \\ &= \left(\frac{y_j}{f_s} - \frac{1}{F}\right) \frac{x_i \cdot \exp\left(-\sum_i x_i w_{ij}\right)}{\left(1 + \exp\left(-\sum_i x_i w_{ij}\right)\right)^2} \end{aligned} \tag{7}$$
$$\text{ただし，} \quad y_j = \begin{cases} 1 & \text{if } j = s \\ 0 & \text{otherwise} \end{cases}$$

4.2 実験環境

実験には図 3 と図 4 の 2 種類の迷路 (これを順に迷路 1，迷路 2 と呼ぶ) を用る．8 × 8 の迷路の外側は壁になっており，各迷路において，1500 試行学習した後に環境を変化させる．各試行において，2500 ステップ学習してもゴールにたどり着けない場合には次の試行に移ることにする．

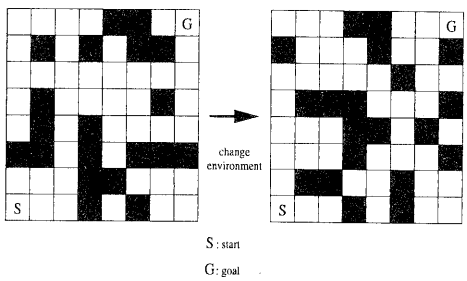

図 3: 迷路 1

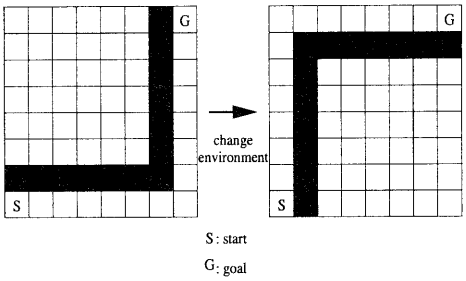

図 4: 迷路 2

4.3 実験結果

迷路 1 環境における，手法 0 から手法 3 までのそれぞれの実験の結果を表 1 に示す．また、迷路 2 における結果を表 2 に示す．各 100 回の実験を行った平均である．実験は 1500 試行学習後に環境を変化させた．誤認識率は誤認識をした回数を認識回数で割ることで算出した．

表 1: 迷路 1 における実験結果

	認識率	試行数	ステップ数	誤認識率
手法 0	100 %	1.36	485.20	2.09 %
手法 1	100 %	1.23	410.60	3.39 %
手法 2	100 %	1.59	600.29	2.97 %
手法 3	100 %	—	217.89	1.69 %

表 2: 迷路 2 における実験結果

	認識率	試行数	ステップ数	誤認識率
手法 0	83 %	2.62	2015.8	2.25 %
手法 1	95 %	1.72	1112.4	5.37 %
手法 2	44 %	6.40	6000.6	4.54 %
手法 3	83 %	—	910.0	2.43 %

これらの表より，手法 1 では，環境変化の認識までの時間を短縮することができているが，誤認識率が上がっていることがわかる．手法 2 では，良い結果を得られなかったことがわかる．手法 3 は，迷路 1 では認識までの時間，誤認識率共に改善されていることがわかる．迷路 2 では，誤認識率は上がっているが，認識までの時間は大幅に短縮されている．認識までの時間が短縮されているのは，一試行ごとではなく，ステップ単位で認識を行ったからであると考えられる．

5 考察

環境の変化によるタスク達成までのステップ数の変化やばらつきを考慮した結果の考察などを行う．

5.1 環境の変化によるステップ数の変化

環境の変化によって，タスク達成までのステップ数がどのように変化するかを述べる．

迷路 1 において，1500 試行学習後に環境を変化させたときのステップ数を示したものが 図 5 である．

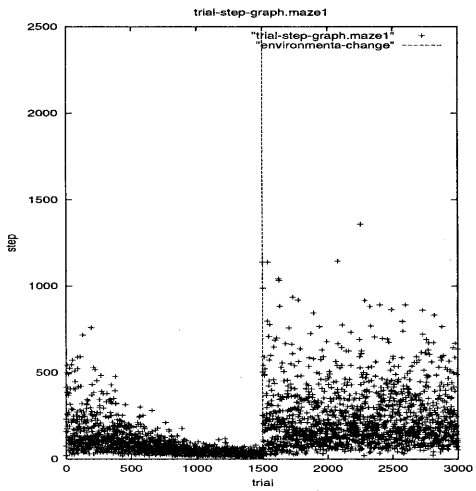

図 5: 迷路 1 におけるステップ数の変化

図 5 より環境変化直後にステップ数が大きく変化していることがわかる．これより，このような大きな環境の変化があった場合にはステップ数の変化を調べれば数試行以内に変化を認識することができると言える．しかし，エージェントの行動選択はランダムネスを含んでいるため学習が安定してきている状態においても，ステップ数は多少のばらつきはある．このばらつきが，誤認識の主な原因となっている．W の各要素よりも，ステップ数のほうが学習時のばらつきが大きいため，タスクの成功率を用いた手法よりも，手法 1 の方が誤認識率が上がっていると考えられる．

次に，迷路 2 において，1500 試行学習後に環境を変化させたときのステップ数を示したものが 図 6 である．

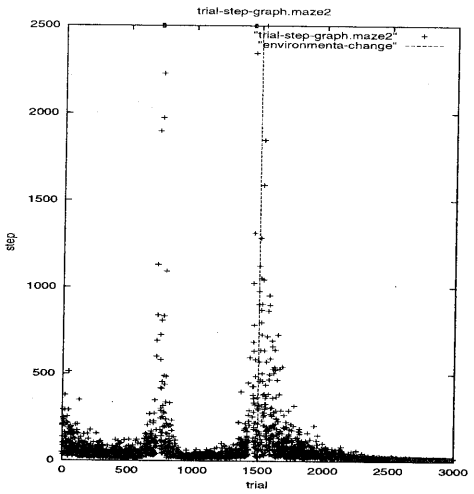

図 6: 迷路 2 におけるステップ数の変化

図より，およそ 700 試行及び 1400 試行学習後に局所解に陥っていることがわかる．迷路 2 で学習した場合，このように局所解に陥ることが多々あった．これにより，迷路 1 において認識を行った場合よりも誤認識率が高くなっていると言える．

5.2 確率的傾斜法の重みの特性

まず，学習中に W の各要素がどのように変化しているのかを図 7 示す．

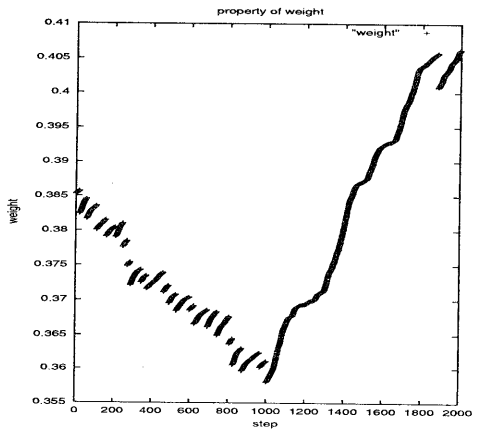

図 7: 重みの特性

この図において x 軸はステップ数，y 軸は W のある要素の値である．1000 ステップの時点で環境が変化している．つまり，環境が変化する直前の 1000 ステップと変化後の 1000 ステップのグラフである．この図より環境が変化すると，W の要素の変化のしかたが大きく変わるということがわかる．一試行あたりのステップ数が大きくなると，単位時間あたりの報酬獲得回数が少なくなる．これにより，一定区間の値をベクトルとして扱い，ベクトル同士の角度を比較する場合，環境変化により角度が大きく変化するということがわかる．手法 2,3 で，認識可能なのはこのような理由による．

6 結論及び今後の課題

確率的傾斜法において，内部変数 W を用いて環境変化を認識する手法を提案した．環境変化後にタスクの達成に時間がかかる場合には，タスクの成功率を用いた環境変化の認識は時間がかかるが，提案手法では，このような場合にも短時間で環境変化を認識することができるということを確認した．提案手法は設計者が確率的傾斜法を実装する際に，内部に組み込むことで使用でき，確率的傾斜法が適用できる問題であれば，簡単に使用できる．

今後の課題としては，他のシミュレーションに適用し，更に汎用性を確認することや，局所解に陥った場合の誤認識を抑えることなどが挙げられる．

参考文献

[Minato98] T. Minato and M. Asada: Environmental change adaptation for mobile robot navigation. *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 1859–1864, 1998.

[Kimura97] Kimura, H., Yamamura, M. and Kobayshi, S.: Reinforcement Learning in POMDPs with Function Approximation, *Proc. of the 14th Int. Conf. on Machine Learning,* 1997.

[Tanaka97] Tanaka, F., Yamamura, M.: An approach to lifelong reinforcement learning through multiple environments. *6th European Workshop on Learning Robots,* pp. 93–99, 1997.

[木村 96] 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. 6, pp. 761–768, 1996.

[Kimura95] Kimura, H., Yamamura, M. and Kobayshi, S.: Reinforcement Learning by Stochastic Hill Climbing on Discounted Reward, *Proc. of the 12th Int. Conf. on Machine Learning,* pp. 295–303, 1995.

[Watkins92] Watkins, C. J. C. H and Dayan, P. : Technical Note:Q-Learning, *Machine Learning, Vol.8,* pp. 55–68, 1992.