2004－MPS－52　(16)
2004／12／21

連続な状態行動空間において近傍状態の報酬予測を用いた強化学習

櫻井 義尚† 本多 中二†

従来の強化学習は一般的に状態行動空間が非連続であった．しかし，実問題においては連続値の状態入力と連続値の行動出力を求められることも多い．状態・行動空間を離散化するのが普通だが，あまり粗く離散化すると細やかな制御ができないという問題が生じる．かといって離散化が細かすぎると探索空間が増大し，通常の離散 MDP における Q-learning とその行動選択方法では，なかなか学習が進まなくなり非実用的となる． 本論文で提案する手法では，連続な行動空間に対応しながらも離散化が細かく探索空間が大きい場合でも近傍状態の報酬を予測し価値関数を更新することにより効率的に学習できる学習則を提案する．

Reinforcement Learning using Prediction of Neighborhood Value in Continuous State-Action Space

Yoshitaka Sakurai† Nakaji Honda†

Generally the conventional Reinforcement Learning had discontinuous state action space. However, in a real problem, it can ask for the state input of a continuation value, and the action output of a continuation value in many cases. It is ordinary to have discrete state-action space. But, warm control cannot be performed if dispersed not much coarsely. But if dispersion is too fine, search space will increase. And , study will not progress easily and it becomes un-practical by usual Q-learning in discrete MDP and its usual action selection method. In this paper, we propose a method to learn efficiently. the method updates Value function by predict neighborhoods Value. This method updates Value function efficiently, even when search space is large.

1. はじめに

試行錯誤を通じて環境に適応する学習制御の枠組として強化学習[1]がある．従来の強化学習は一般的に状態行動空間が非連続であった．しかし，実問題においては連続値の状態入力と連続値の行動出力を求められることも多い．

連続な状態空間を持つ強化学習問題では，Q-learning[2]における Q 値や状態価値の表現に関数近似を用いることが多い．代表的な関数近似法として，tile coding(CMAC)，多層ニューラルネットワーク，ファジィ，動径基底関数（radial-basis-function），多変量回帰などが提案されている[1]．しかし，多層ニューラルネットワークは静的な訓練集合と同じデータでの複数回の試行を前提としているため，強化学習のように非定常関数への対応が求められるものには不向きと考えられ，その他の線形手法がよく用いられる．

線形手法による近似の場合，状態・行動空間を離散化するのが普通だが，あまり粗く離散化すると細やかな制御ができないという問題が生じる．かといって離散化が細かすぎると探索空間が増大し，通常の離散 MDP における Q-learning とその行動選択方法では，なかなか学習が進まなくなり非実用的となる．

本論文で提案する手法では，線形手法により近似において離散化が細かく探索空間が大きい場合でも効率よく価値関数を更新し，収束速度の向上が望める手法を提案する．

われわれはこれまでパターン情報による能動的学習法 (Pattern Information Based Active Learning Method, PBALM)と呼ばれるソフトコンピューティング的なモデリング手法を提案し，これを学習制御へと適用してきた[3]．PBALM は，繁雑な数式や複雑なアルゴリズムは一切使用せず，単純な操作のみによっているのが特色である（低コスト性，扱いやすさ）．また，この手法の特徴として「ぼかし」処理による特徴の抽出がある．本論文では，この「ぼかし」処理を強化学習に用いることにより，学習速度の向上を目指し，その効果を検証する．

本論文ではまず，「ぼかし」処理を用いた強化学習を定義する．そして，これが学習に及ぼす効果を検証するため，Acrobot の学習制御をとおして Q-learning と提案手法との比較を行う．

†電気通信大学
The University of Electro-Communications

2. ぼかしを用いた強化学習

2.1. IDS

われわれはソフトコンピューティング的なモデリング手法 PBALM を提案しさまざまな応用を試みてきた[4]．PBALM の特徴の一つにぼかしによる傾向の抽出があり，われわれはこれをインクドロップスプレッド(Ink Drop Spread, 以後 IDS と呼ぶ)と呼んでいる．これは，図 1 のようなデータがあったとき，各データ点において「ぼかし」処理（図 2）を行い．その干渉パターンより図 3 のように入出力特性をモデリングする手法である．

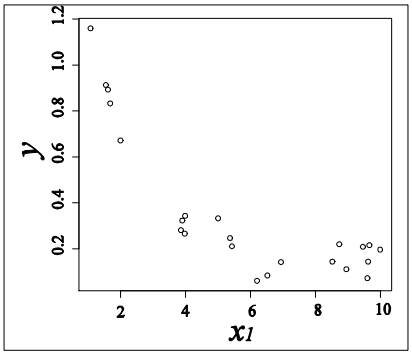

図 1. データの平面へのプロット（射影）

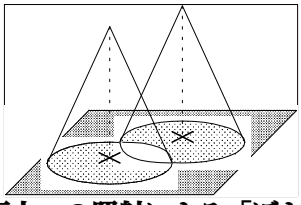

図 2. 平面上への照射による「ぼかし」操作

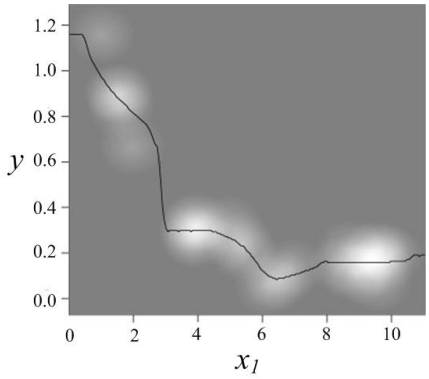

図 3.干渉パターンからの x_1 - y **入出力特性の抽出**

「ぼかし」は頑健性を実現するための手段として用いられており，IDS でも「ぼかし」という単純な処理により頑健なモデリングを実現している．この IDS により状態行動価値関数の推定を行う強化学習を定義する．

2.2. 学習アルゴリズム

強化学習を代表する手法の一つとして Q-learning があるこれは，以下のようなアルゴリズムで状態行動価値関数 $Q(s,a)$を更新するものである．

$Q(s,a)$を任意に初期化
各エピソードに対して繰り返し：
　s を初期化
　エピソードの各ステップに対して繰り返し：
　　Q から導かれる方策(ε グリーディ方策)
　　を使って，s での行動 a を選択する．
　　行動 a を取り，r, s'を観測する
　　$Q(s,a) \leftarrow Q(s,a)+\alpha[r+\gamma max_{a'} Q(s',a')- Q(s,a)]$
　　学習係数 $0 \leqq \alpha \leqq 1$ ，割引率 $0 \leqq \gamma \leqq 1$
　　$s \leftarrow s'$;
　s が終端状態ならば繰り返しを終了

Q-learning のアルゴリズム

これを IDS を適用し，「ぼかし」処理をおこなうことにより更新する状態行動対の近傍の状態行動対にも影響を与えることにより，広い範囲の状態行動価値の推定を行い，学習を早くする．これを IDS 更新則と呼び, これを Q-learning に適用すると以下のような更新則になる．

$Q(s,a) \leftarrow Q(s,a)+\alpha[r+\gamma max_{a'} Q(s',a')- Q(s,a)]$
状態行動対　から距離 I_s 以内の状態行動対すべてに対して
　$Q(s+u,a+v) \leftarrow Q(s+u,a+v) +$
　　$b\ \alpha\ [r+\gamma max_{a'} Q(s',a')- Q(s+u,a+v)]$
　$b = I_0 \exp(-I_{bias} (\|u+v\|_2))$
　$-I_s \leqq u,v \leqq I_s$; $0 < I_0, I_{bias}$;

u,v は状態 s，行動 a からのそれぞれの軸での距離を表し，状態 s，行動 a が多次元の場合，ベクトルとなり距離は l_2 ノルムにより計算される．影響の度合いを表すbはこの距離とパラメータ I_0, I_{bias} によって決まる．

IDS 更新則は主に初期の学習において未知の状態行動価値が多い場合に，一つの学習の効果を広げる働きをもち，近傍の状態行動価値の推定法として機能する．しかし，学習が進むにつれて詳細な学習が必要になると，この近傍への影響が収束への妨げになることが考えられる．そこで，時間減衰パラメータをもちいて，学習が進むにつれて影響の度合いbが小さくなるように(1)式(2)式のようにパラメータ I_0, I_{bias} を更新する．

$$I_0 = I_{02} \exp(-I_{0bias}\ \mathrm{time}^2) \quad (1)$$
$$I_{bias} = I_{bias0} (1 - \exp(-I_{bias2}\ \mathrm{time}^2)) \quad (2)$$

time にはステップ数などを設定する．このように「ぼかし」効果を時間ともに減衰させる IDS 更新則を減衰 IDS 更新則と呼ぶ.

3. 従来の強化学習との比較

3.1. Acrobot

Acrobot は鉄棒体操選手に似た 2 リンクのアクチュエーターロボットである．リンク 1 は鉄棒をつかむ体操選手の手に相当し，トルクを出すことはできない．リンク 2 は体操選手の腰に相当し，トルクを加えることができる．このシステムは各リンクの角度q と角速度$\dot{q}$ の計 4 の連続値状態変数を持つ．モデルを図 4 に示す．Acrobot は非線形が強いノンホロノミックシステムであり，さらに状態・行動空間が連続であるため，強化学習によってこの制御ルールを獲得することは難しい課題の一つであることが知られている．

また，一般的に Acrobot はリンクの駆動角度に制限を設けていないが，今回はより鉄棒体操選手のモデルに近づけるためにリンク 2 の駆動角度に制限を設けた．モデルの条件は以下のようになっている．シミュレーションの時間ステップ幅は 0.05 [sec]に設定．トルク$\tau \in [-2, 2]$ が与えられる．リンクの角度$q_1 \in [-\pi, \pi]$，$q_2 \in [-0.75\pi, 0.75\pi]$，リンクの角速度$\dot{q}_1 \in [-2\pi, 2\pi]$，$\dot{q}_2 \in [-2\pi, 2\pi]$ のように制限される．

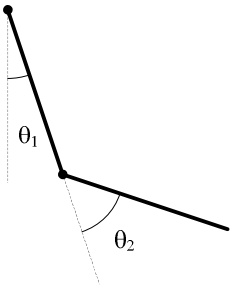

図 4. Acrobot

3.2. 実験結果

制御対象として 3.1 で説明した Acrobot を用いて学習制御を行う．条件は以下のとおりである．

制御対象：Acrobot（駆動制限あり）
目標状態：リンク 1 を 180 度に回転，180 度を過ぎたところでエピソードを終了する．
初期状態：$q_1 = q_2 = \dot{q}_1 = \dot{q}_2 = 0.0$
行動選択則：ε グリーディ方策，$\varepsilon = 0.0001$
学習則：Q-learning，$\alpha = 0.1$，$\gamma = 0.9$
報酬：$r = (q_1 / \pi)^2$

状態行動関数を状態行動空間を離散したテーブルの線形近似で表現し，Q-learning（以後 QL と呼ぶ），IDS 学習則による Q-learning（以後 IDSQ と呼ぶ），時間減衰 IDS 学習則による Q-learning（以後減衰 IDSQ と呼ぶ）による学習制御を行う．

まず，行動τを 10 分割，状態q_1，q_2，$\dot{q}_1$，$\dot{q}_2$ をそれぞれ 10 分割した場合，つまり状態数 10000 行動数 10 の場合の結果を示す．QL の結果を図 5 に，IDSQ（$I_s = 1, I_0 = 1, I_{bias} = 0.3$）の結果を図 6 に，減衰 IDSQ（$I_s = 1, I_{bias} = 0.3, I_{02} = 1, I_{0bias} = 0.0001$）の結果を図 7 に示す．

QL は最初に桁違いに長いエピソードを試行し，それから急激に学習して，収束している．IDSQ，減衰 IDSQ はその 20 分の 1 以下のステップ数で最初のエピソードを終了して，その後すぐに収束している．しかし，どの学習法もほぼ収束した状態でときおり長いステップ数をようするエピソードを試行している，これはεグリーディ方策による探索の効果であると考えられる．収束するまでのステップ数は違うが最終的な収束値はおよそ 300 ステップ程度で最短のステップは 100 程度と学習結果には差は見られなかった．

収束速度を比較しやすくするためにこの 3 つのグラフの 100 エピソードまでを並べたグラフを図に示す．これを見ると IDSQ と減衰 IDSQ はほぼ同じだが，QL が劣っているのが見て取れる．

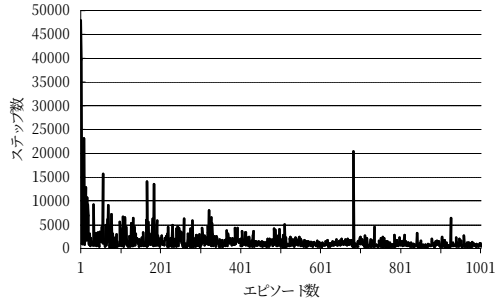

図 5. QL(状態数 10000，行動数 10)の結果

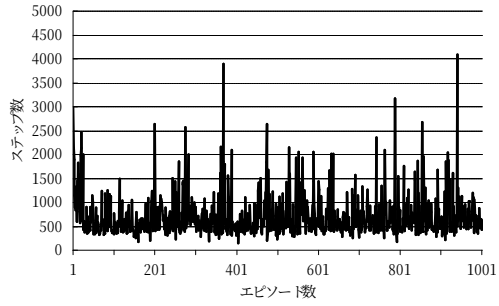

図 6. IDSQ(状態数 10000，行動数 10)の結果
（$I_s = 1, I_0 = 1, I_{bias} = 0.3$）

τを 20 分割，q_1，q_2，$\dot{q}_1$，$\dot{q}_2$ をそれぞれ 20 分割した場合，つまり状態数 160000 行動数 20 の場合の結果を示す．QL の結果を図 9 に，IDSQ（$I_s = 1, I_0 = 1, I_{bias} = 0.7$）の結果を図 10 に，減衰 IDSQ（$I_s = 1, I_{bias} = 0.7, I_{02} = 1, I_{0bias} = 0.01$）の結果を図 11

に示す.

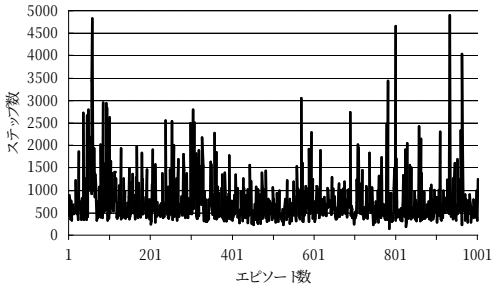

図 7. **減衰** IDSQ(**状態数** 10000，**行動数** 10)**の結果**
（I_s = 1, I_{bias} = 0.3, I_{02} = 1, I_{0bias} = 0.0001）

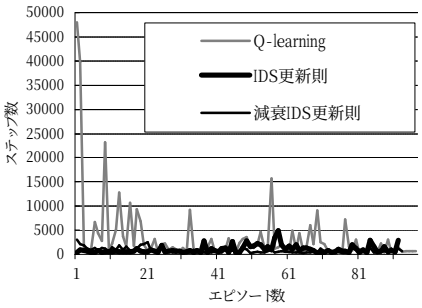

図 8. **状態数** 10000，**行動数** 10 **の時の比較**

QL の結果をみると状態数が多すぎるせいで，値が収束していない．しかし，IDSQ，減衰 IDSQ の結果をみるときちんと収束しているのが見て取れる．また，その収束速度も状態数 10000 行動数 10 の場合と比べてもさほど違いがなく，「ぼかし」による学習促進効果がはっきりと認識できる. また, IDSQ と減衰 IDSQ をくらべた場合, IDSQ の方が早く収束している．これは状態数が多ければ多いほど「ぼかし」が有効であることを意味する.

4. おわりに

本論文では「ぼかし」による近傍の状態行動価値の推定法としてはたらく IDS 学習則を提案し，これを Q-learning に適用した．これは全般的によくはたらき，一般的な Q-learning より速い収束をみせた．特に状態数が多いときの効果は顕著であり，複雑で重たい処理を必要とせずに高い効果が得られるため，有用な手法であるといえる.

<参考文献>

[1] R.S.Sutton,& A.G.Barto: "Reinforcement Learning: An Introduction", Cambridge MA, MIT Press, (1998)
[2] Watkins, C. J. C. H. "Learning from Delayed Rewards", PhD thesis, Cambridge University, Cambridge, England. (1989)
[3] Y. Sakurai, N. Honda, J. Nishino: "Acquisition of Knowledge for Gymnastic Bar Action by Active Learning Method", Journal of Advanced Computational Intelligence & Intelligent Informatics (JACIII), Vol.7, No.1, pp.10-18 (2003)
[4] M. Murakami, N. Honda : "Hardware for a New Fuzzy-based Modeling System and its Redundancy", Proceedings of NAFIPS-04, Canada , pp.599-604, (2004.06)

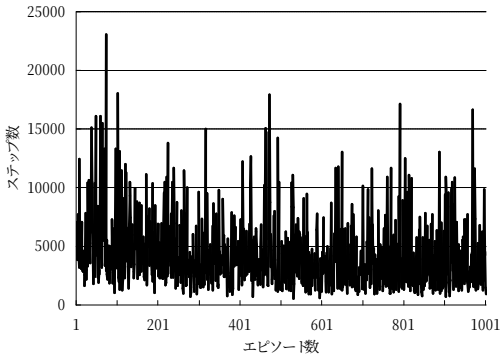

図 9. QL(**状態数** 160000，**行動数** 20)**の結果**

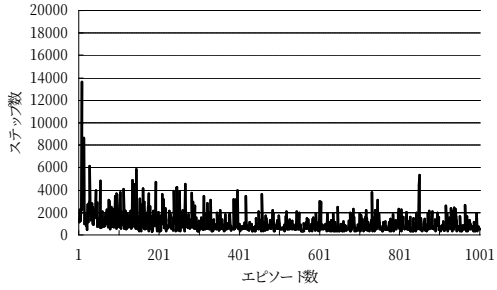

図 10. IDSQ(**状態数** 160000，**行動数** 20)**の結果**
（I_s = 1, I_0 =1, I_{bias} = 0.7）

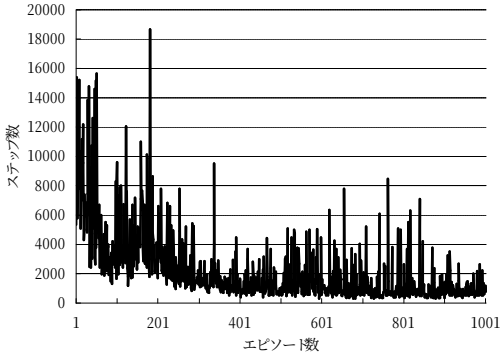

図 11. **減衰** IDSQ(**状態数** 160000，**行動数** 20)**の結果**
（I_s = 1, I_{bias} = 0.7, I_{02} = 1, I_{0bias} = 0.01）