特集論文

Technical Papers

実例に基づく強化学習法による失敗しない制御方法の学習

Learning not to Fail by an Instance-Based Reinforcement Learning Method

畝見 達夫*
Tatsuo Unemi

* 創価大学工学部情報システム学科
Dept. of Information System Sci., Faculty of Eng., Soka Univ., Tokyo 192, Japan.

1992年3月30日 受理

Keywords: instance-based learning, reinforcement learning method, learning control, cart-pole system, delay system.

Summary

We propose an instance-based learning algorithm named IBRL3 which learns how to avoid the negative reinforcement from environment. A cart-pole balancing problem and a monitoring ship navigation problem are used to certify its learing performance. In this algorithm, a tuple of input and output data of each execution cycle are stored in memory verbatim, and the action of each cycle is decided by retrieving the nearest neighbor of the current input data. The number of stored instances is reduced by replacing the nearest but less reliable instance by new one. Experimental results of computer simulation show that IBRL3 is robust for distinct settings of parameter and for noisy environments, and it is efficient enough to apply to real-time control problems.

1. まえがき

本論文では，失敗しないことを学習する実例に基づく強化学習法のアルゴリズムIBRL3を提案し，不安定系の典型例である倒立振子の安定制御と，非線形遅れ系の典型例である大型船舶の操舵制御への応用を通して，計算機シミュレーションによりその性能を評価する．

実例に基づく学習方式は，ニューラルネットや学習オートマトンあるいは帰納学習などによるアプローチとは異なり，対象システムのモデルをある種の方程式の形で近似したり，抽象化されたルール集合や決定木の形で推定するようなことはせずに，各実行サイクルで得られた入出力対のデータをそのまま記憶し，新たな状況に対しては，過去の記憶の中から類似データを検索することによって意思決定を行う．この種のアプローチはSamuelの先駆的な研究[1]以来，綴りから発音への変換[2]，自然言語の翻訳[3][4]，診断システム[5]，簡単なロボット制御[6]，ロボットマニピュレータの制御知識の獲得[7]などの応用例が報告されている．また，教師つきの分類問題を対象領域とするアルゴリズムについては，その性質が調べられている[8]．これらの手法は，暗記学習(rote learning)，記憶に基づく推論(memory-based reasoning)，実例基づく学習(instance-based learning)などの名で呼ばれてきたが，最近の文献[9]ではFig. 1に示すように，手本に基づく学習(exemplar-based learning)の一種として位置づけられている．

我々は先に，主に生体の環境適応行動をシミュレートする立場から，記憶に基づく学習の枠組みに基づいた無限長離散時系列を対象とする学習機構を提案

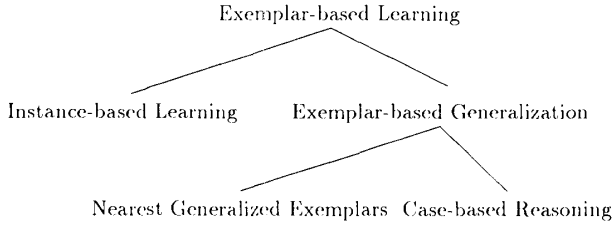


Fig. 1 Subtypes of exemplar-based learning[9].

し[10]，さらに，その学習機構を一般化する形で，学習アルゴリズムの確立を試みた[11]．ここでは，倒立振子の制御などへの応用が可能な，拡張された学習アルゴリズムを提案する．

応用領域としての倒立振子の学習制御については，Michie and Chambers の BOXES[12] をはじめ，強化学習法（reinforcement learning method）の応用問題として Barto, Sutton, Anderson, Selfridge[13]-[15] らによって取り組まれてきた．また，実例に基づく方法としては，Connel and Utgoff の CART[16] があり，これらの性能比較が Sammut によって報告されている[17]．実例に基づく方法という意味では IBRL3 は CART に似ているが，基本的なアルゴリズムに大きな相違があり，問題領域に対する汎用性とアルゴリズムの簡単さの点で IBRL3 のほうが優れている．また，倒立振子の学習制御問題は，遺伝的アルゴリズムの応用として Odetayo and McGregor[18], Whitley, Dominic and Rajarshi[19]，小池ほか[20]らによっても扱われている．

大型船舶の操舵制御については，認知科学の立場から人間の学習の特性を調べる目的で行われた安西の研究[21] をはじめ，ファジィあるいはニューラルネットの応用領域としても取り上げられている．ただし，ここで用いる制御目的は，それらの研究で取り上げられた水門の通過ではなく，目標物の監視である．

以下，学習制御の対象問題，学習アルゴリズム，シミュレーション実験とその結果，ほかのアルゴリズムとの比較について述べる．

2. 学習の課題と試行過程

本章では，学習問題の仕様と学習を行うための試行過程について述べる．

2・1 倒立振子の学習制御問題

倒立振子の制御は，典型的な不安定系制御の問題としてしばしば取り上げられる課題である．ここでは，学習メカニズムの比較研究[17] や遺伝的アルゴリズムの応用研究[18] と同様に Anderson が用いた計算機シミュレーションによる方法[15] を採用した．Fig. 2 にその概念図を示す．

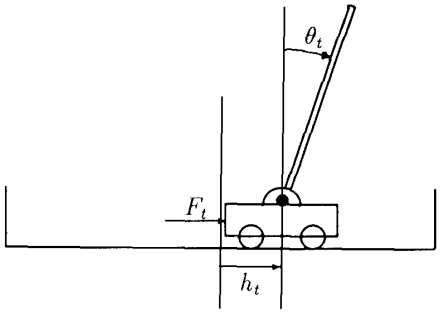


Fig. 2 The cart-pole system.

時刻 t における系の状態は以下に示す四つの状態量によって記述される．

h_t：区間内における台車の中央からの距離

$\dot{h}_t$：台車の速度

θ_t：垂直な状態からの棒の角度

$\dot{\theta}_t$：棒の角速度

すなわち，これら四つの状態量が学習システムへの入力となる．BOXES や遺伝的アルゴリズムでは各次元の数値を区間に分割しているが，ここでは，AHC や CART のように 4 次元の数値データそのものを入力として利用する．

学習システムからの出力，すなわち制御量は，台車に加えられる力 F_t である．ここでは，BOXES などと同様に，学習システムは各実行ステップにおいて，左右どちらに台車を押すかを決定するものとする．つまり，制御量の選択肢は「右」または「左」のいずれかであり，具体的な力の大きさを連続値として調節することはしない．左右それぞれの力の大きさは，あらかじめ固定されているものとする．

学習システムには，上にあげた四つの入力とともに，強化信号（reinforcement）r_t を与える．r_t の値は，台車がレールの端に到達するかあるいは棒が倒れた場合 -1 であり，その他の場合は 0 である．学習の目的は，なるべく $r_t=-1$ とならないようにすることである．

倒立振子の系は次のような運動方程式を用いてシミュレートする．

$$\ddot{\theta}_t=\frac{g\sin\theta_t+\cos\theta_t\left[\dfrac{-F_t-m_p l\dot{\theta}_t^2\sin\theta_t}{m_c+m_p}\right]}{l\left[\dfrac{4}{3}-\dfrac{m_p\cos^2\theta_t}{m_c+m_p}\right]}$$

$$\ddot{h}_t=\frac{F_t+m_p l\left[\dot{\theta}_t^2\sin\theta_t-\ddot{\theta}_t\cos\theta_t\right]}{m_c+m_p}$$

ただし，g は重力加速度 (9.8 m/s^2)，m_c は台車の質量，m_p は棒の質量，l は棒の長さである．レールと台車の間の摩擦および台車と棒をつなぐヒンジの摩擦はないものとする．各時刻の状態を一定の小さな時間間隔 τ ごとに Euler 法による数値計算により近似する．

2・2 大型監視船の学習制御問題

大型船舶の航行制御は，大きな遅れを持つシステムの制御問題としてしばしば取り上げられる課題である．ここでは，次のような問題設定を用いることにより，学習問題としての条件を倒立振子の場合と同じも

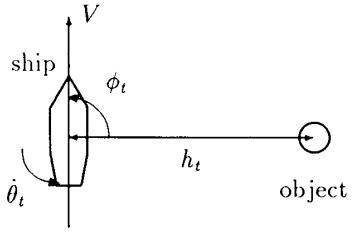


Fig. 3 The monitoring ship.

のとした．すなわち，船舶は目標物を監視しながら定速で航行する．ただし，目標物からは決めれた距離の範囲内にいなければならず，遠すぎても近すぎてもいけない．さらに，進行方向から目標物への角度も決められた範囲内に保たなければならない．Fig. 3 にその概念図を示す．

学習システムへの入力は次の三つの連続量である．

h_t：目標物と船舶の距離

ϕ_t：船舶の進行方向に対する目標物の視角度

$\dot{\theta}_t$：船舶の角速度

学習システムからの出力，すなわち制御量は，舵の切り角 U_t である．ここでは，学習システムは各実行ステップにおいて，「左」「右」「直進」の三つの選択肢から一つを選ぶものとする．

船舶と目標物の距離，および，船舶の進行方向に対する目標物の視角度が決められた範囲からはずれた場合，強化信号 r_t として -1 を与える．その他の場合は $r_t=0$ である．

大型船舶の系は次のような運動方程式を用いてシミュレートする．

$$\dot{x}=V\cos\theta$$
$$\dot{y}=V\sin\theta$$
$$\ddot{\theta}=(U-\dot{\theta})/T$$

ここで，(x, y) は船舶の座標，V は船舶の速度，θ は船舶の角度，U は舵の切り角，T は時定数である．各時刻の状態は倒立振子の場合と同様に Euler 法によって近似する．

2・3 学習の手順

学習は制御の試行を繰り返すことによって実行される．1 回の試行は次の手順に従って行われる．

(1) 理想的な制御規則に従えば失敗を起こさないような初期状態から始める．

(2) 学習システムよって対象システムを制御する．

(3) 強化信号が -1 となったところで終了．

学習が効果的であれば，試行を繰り返すうちに，1 試行当りのシミュレーションステップ数が増加する．

3. 学習アルゴリズム

実例に基づく学習の基本的な考え方は次のようなものである．

- とにかく経験によって得られた入力データと出力データの対を加工せずに記憶の中へ次々と蓄える．
- 意思決定の場面では，記憶されたデータの中から，現在の状況に類似したものを検索し，目的に沿った出力を選択する．

分類問題に適用する場合は，課題として与えられた入力データと最も類似した記憶データを検索し，それと同じカテゴリに含まれるものと予想すればよい．しかし，強化学習の対象となる制御問題では，各ステップでの意思決定の評価が後にならないと明らかにならないため，Fig. 4 のような枠組みを用いる．すなわち，学習要素は，環境からの入力 $\boldsymbol{x}$，それに対して実行要素が決定した出力 y，さらに，環境から与えられる強化信号 r を取り込み，記憶を書き換える．実行要素は，学習要素によって編集された記憶を参照しながら現在の入力から次の出力を決定する．

ここで提案するアルゴリズムを先に提案したアルゴリズム IBRL1, IBRL2[11] の拡張と考え，**IBRL 3** と呼ぶことにする．IBRL3 では，一つの試行が終了するまで，検索の対象となる記憶には変更を加えずに，試行が終了した時点で，その試行中に得られたデータをまとめて記憶する．これによって評価が明らかとなったデータのみを検索対象とすることができる．

学習アルゴリズムの主プログラムは次のように書ける．

［**アルゴリズム 1**］

```
program IBRL 3;
repeat begin
  t=0;
  repeat begin
```

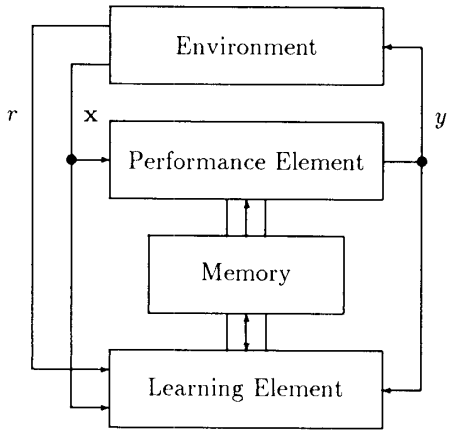


Fig. 4 Block diagram of learning system.

t：＝t＋1；
$\boldsymbol{x}$：＝入力データ；
y：＝**Policy** ($\boldsymbol{x}$)；
出力データ y を実行；
r：＝強化信号
end until $r=-1$
ModifyMem
end forever.

各ステップにおいて得られた入力と決定した出力は，関数 **Policy** の中で作業用の待ち行列に記録し，強化信号が r が -1 となった時点で，記憶へ移しかえる．

本アルゴリズムでは，記憶として，出力データの選択肢に対応する複数の集合 M_y ($y\in\mathcal{O}$) を用いる．ここで $\mathcal{O}$ は出力選択肢であり，倒立振子の場合 {右，左}，監視船の場合 {右，直進，左} である．個々の記憶集合 M_y は，入力データと評価値 P_i の組

$$I_i=((u_{i1}, u_{i2}, \cdots, u_{in}), P_i)$$

を要素とする．この要素を「実例」と呼ぶことにする．ただし，u_{ij} は入力ベクトル $\boldsymbol{x}_i$ の j 番目の要素，P_i は，その後失敗に至るまでのステップ数である．n は入力ベクトルの要素数であり，倒立振子の場合 $n=4$，監視船の場合 $n=3$ である．

以下，意思決定手続き **Policy** と，記憶変更手続き **ModifyMem** について説明する．

3・1 意思決定手続き

各実行ステップにおいて，既知の実例の中から，現在の入力データ $\boldsymbol{x}$ に最も近い入力データを持つものを，それぞれの記憶 M_y の中から見つけ出し，評価値の高いほう，つまり失敗までの残りステップ数が多いほうを現ステップの出力として選択する．ただし，M_y が空，あるいは最も近い実例までの距離がしきい値以上の場合は，該当する出力データの評価値を平均的な評価値であるものと仮定する．最後に，記憶変更手続きに備えて，入力，出力，実例，距離の 4 つ組を待ち行列 Q の t 番目の要素 Q_t として記録する．詳細は次のとおりである．

［アルゴリズム 2］
function **Policy** ($\boldsymbol{x}$)
for all v in $\mathcal{O}$ do begin
既知の実例集合 M_v の中から $\mathcal{D}(I_i, \boldsymbol{x})$ を
最小にする実例 I_i を見つけ出す；
if $M_v=\phi$ or $\mathcal{D}(I_i, \boldsymbol{x})\geq 1$
then begin p_v：＝E；N_v：＝ϕ；D_v：＝∞ end
else begin p_v：＝P_i；N_v：＝I_i；D_v：＝$\mathcal{D}(I_i, \boldsymbol{x})$ end
end
$\mathcal{O}$ の中から p_y が最大となる出力データ y を選ぶ；
ただし，y の候補が複数ある場合には
それらの中からランダムに一つを選ぶ；
Q_t：＝($\boldsymbol{x}$, y, N_y, D_y)；
return y.

$\mathcal{D}(I_i, \boldsymbol{x})$ は実例 I_i と入力データ $\boldsymbol{x}$ の距離を表す関数であり，次のように定義する．

$$\mathcal{D}(I_i, \boldsymbol{x})=\frac{1}{2n}\sum_{j=1}^{n}\frac{(u_{ij}-x_j)^2}{s_j^2}$$

ここで，s_j^2 は，記憶中に蓄えられているすべての実例についての，入力データの各要素 u_{ij} の標本分散である．すなわち，N を実例の個数とすると

$$s_j^2=\frac{1}{N-1}\sum_{k}^{N}(\bar{u}_j-u_{kj})^2$$

である．次に示すとおり，$\mathcal{D}(I_i, I_k)$ の平均値は分布に関係なく 1 となる．

$$E_{\mathcal{D}}=\frac{1}{2n}\cdot\frac{2}{N(N-1)}\cdot\sum_{i=2}^{N}\sum_{k=1}^{i-1}\sum_{j=1}^{n}\frac{(u_{ij}-u_{kj})^2}{s_j^2}$$
$$=\frac{1}{2n}\cdot\frac{2}{N(N-1)}\cdot Nn(N-1)$$
$$=1$$

この性質を考慮し，距離のしきい値を 1 とした．つまり，平均より近い実例のみを参考にする．もちろん，距離の定義にほかの尺度を用いることも可能だが，計算コストの小ささおよび標本分散を用いた標準化による上述のような利点から，このようなユークリッド距離の 2 乗を採用した．

E は評価値の期待値を表す大域変数であり，その値は 1 回の試行が終了するたびに次の式に従って更新する．

$$E：=\alpha\cdot E+(1-\alpha)\cdot\beta\cdot t.$$

α と β は区間 $[0, 1]$ 内にある定数であり，実験では $\alpha=0.8$，$\beta=0.5$ とした．

3・2 記憶変更手続き

一つの試行が終了した時点で，その試行における実行の履歴 Q をもとに，記憶に変更を加える．基本的な設計方針は，

- 状態空間を適当な間隔で区切るように実例を記憶する．
- 現在の記憶を使うと，その出力データを選択した場合に，「失敗までにあと何ステップ実行を続けられそうか」を示す値を評価値として管理する．

の二つである．これらの設計方針は次のような手続きによって実現される．

- データどうしが非常に近く，かつ新しいデータのほうが評価が高ければ，古いデータを新しいデー

タで置き換える．

- データどうしは非常に近いが，新しいデータのほうが評価が低い場合には，古いデータの評価を下げる．
- 新しいデータに近いデータがなければ，新しいデータを記憶に追加する．

詳細は次のとおりである．

［アルゴリズム 3］

procedure **ModifyMem**；
for k in [1..t] do begin
($\boldsymbol{x}, y, I_i, d$)：＝$Q_k$；
if $d<\rho$ then
if $t-k>P_i$
then M_y 中の I_i を新たな実例 ($\boldsymbol{x}, t-k$) で置き換える
else P_i を縮小する
else M_y に新たな実例 ($\boldsymbol{x}, t-k$) を追加する
end.

Q の中のすべての要素 Q_k ($k=1, 2, \cdots, t$)について，もし Q_k の第 4 要素である距離 d がしきい値 ρ より小さく，かつ Q_k の第 3 要素である実例 I_i について，その評価値 P_i が $t-k$ より小さければ，Q_k 中の入力データ $\boldsymbol{x}$ と評価値 $t-k$ から構成される新たな実例で記憶中の I_i を置き換える．もし P_i が，$t-k$ より大きければ P_i を縮小する．もし距離 d がしきい値 ρ より大きければ，評価値 P_i の大きさに関係なく新たな実例を記憶 M_y に追加する．もちろん y は Q_k の第 2 要素である．

評価値の縮小は，もとの値の重みを 1 とし，Q_k に対応する評価値 $t-k$ の重みを $1-\sqrt{d}$ として加重平均をとることによって実現する．つまり，近いものほど 1 に近い大きな重みを与える．この種の評価値の縮小を導入しないと，以前に成功した経路の近くをたどって失敗するという行動パターンを何度も繰り返す可能性が生じ，学習性能低下の原因となる．

4. シミュレーション実験とその結果

4・1 倒 立 振 子

Sammut の実験[20]と同じ設定でシミュレーション実験を行った．すなわち，各試行の初期状態を $|h_0|<0.1$ m，$|\theta_0|<6°$ なる一様乱数によって設定し，$|h_t|>2.4$ m あるいは $|\theta_t|>12°$ となったところで強化信号 $r_t=-1$ を与える．$\dot{h}_0$ および $\dot{\theta}_0$ はともに 0 とする．倒立振子の仕様は，台車の質量 $m_c=1.0$ kg，棒の質量 $m_p=0.1$ kg，棒の長さ $l=0.5$ m とする．台車に加える力 F_t については，以下の 3 種類の設定についてそれぞれ実験を行った．

設定 1：出力データ「右」の場合 10 N，「左」の場合 −10N

設定 2：「右」の場合 5N，「左」の場合 −10N とする．

設定 3：設定 1 と同様だが，$|e_t|<2$N なる一様分布のノイズ e_t を加える．すなわち $F_t=\pm 10\text{N}+e_t$ とする

設定 3 は Sammut の実験では扱われていない．シミュレーションステップの時間間隔 τ は 0.02 s とし，台車に力 F_t を加える時間間隔は τ と等しいものとした．

1 回の実験は，試行を繰り返すうちに 10 000 ステップ以上倒れなかった時点で終了とし，それまでに要した試行の回数を実験結果とする．数回の実験結果から，学習能力を判定する．もちろん，試行回数は少ないほうが望ましい．

距離のしきい値 ρ の値を 0.2 から 1.0 まで 0.1 刻みに変化させ，それぞれの値について，100 回の実験を行った．Fig. 5 に ρ の変化に伴う試行回数および実例数の平均値の変化を示す．

試行回数は，設定 1, 2, 3 の順に多くなる．ρ の変化に対しては，$\rho=0.3$ から 0.5付近で試行回数が最小となっており，ρ の値は大きすぎても小さすぎても，学習

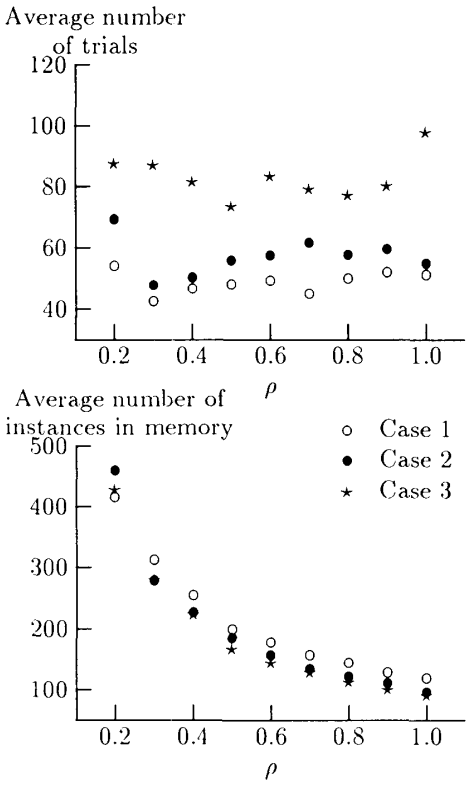


Fig. 5 Average number of trials and instances in memory over 100 runs varying the threshold value ρ —cart pole system.

性能は悪くなることがわかる.実例数のほうを見ると,いずれの実験でもρが大きくなるに従って確実に小さくなることがわかる.ρが小さいほど新たな実例を記憶に追加する機会が増すはずであるから,この結果は当然である.

4・2 大型監視船

大型監視船問題は次のような設定でシミュレーションを行った.各試行の初期状態を$125\,\mathrm{m}<r_0<175\,\mathrm{m}$, $45°<|\phi_0|<90°$となる一様乱数によって設定し,$r_t<100\,\mathrm{m}$あるいは$r_t>200\,\mathrm{m}$あるいは$|\phi_t|>120°$となったところで強化信号$r_t=-1$を与える.角速度$\dot{\theta}_0$は0とする.舵の切り角U_tは,出力データが「右」の場合$-1°/\mathrm{s}$,「直進」の場合$0°/\mathrm{s}$,「左」の場合$1°/\mathrm{s}$とする.船舶の速度Vは0 m/sした.時定数および外乱について次のような三つの異なる値を設定し,それぞれ実験を行った.

設定1:時定数$T=8$, 外乱なし.

設定2:時定数$T=16$, 外乱なし.

設定3:時定数$T=8$, 外乱として,各シミュレーションステップにおいて,$|e_t|<2\mathrm{m/s}$なる一様分布のノイズを船舶のx座標とy座標にそれぞれ加える.

シミュレーションステップの時間間隔τは0.02秒,舵の切換間隔,すなわち,1ステップの時間間隔はτの50倍の1秒間とした.

1回の実験は,試行を繰り返すうちに1回の試行が2 000ステップ以上続いた時点で終了とする.倒立振子の場合と同様に,距離のしきい値ρの値を0.2から1.0まで0.1刻みに変化させ,それぞれの値について,100回の実験を行った.Fig. 6にρの変化に伴う試行回数および実例数の平均値の変化を示す.倒立振子の場合と異なり,$\rho=0.2$あるいは0.3付近で最小となっている.すなわち,学習性能の意味でのρの最適値は問題領域に依存する.しかし,ここで行った2種類の問題に関する限り,それほど大きな差はない.

また,双方の実験とも,設定3の結果は本アルゴリズムがノイズを含む環境に対してもある程度適用可能であることを示しており,実機への適用に対する有用性を示すものである.

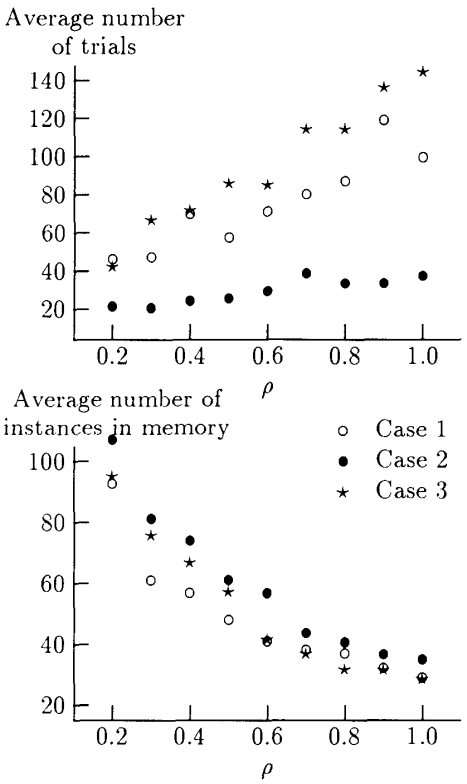


Fig. 6 Average number of trials and instances in memory over 100 runs varying the threshold value ρ—monitoring ship.

5. ほかのアルゴリズムとの比較

Sammut は Michie らの BOXES[12], Barto, Sutton らの AHC[13][14], および Connel らの CART[16] を実験によって比較した[17].その結果と比較すると,Table 1のようになる.ここで提案した学習アルゴリズムIBRL3のデータは,$\rho=0.3$の場合である.この表からわかるとおり,設定1ではCARTについで優れた学習能力を示し,設定2では,これらの中で最も優れた学習能力を示している.Sammutの論文には設定2でのCARTの成績が書かれていないが,それより条件の緩い設定で平均58回の試行を必要としたことが述べられている.BOXES, AHCおよびCARTでは,設定2における試行回数が設定1の3.7倍,28倍および4.5倍となっているのに対し,IBRL3では1.2倍以下となっており,IBRL3が異なった環境に対する適応性に優れていることがわかる.

計算量の面では,IBRL3は,各実行ステップでの意思決定のために既知の実例数Nに比例した計算時間を必要とする.しかし,ρの値を適当に設定してやることで,Nを減らすことが可能である.実験では最も性能が良くなる$\rho=0.3$の場合でもNの値は平均

Table 1 Comparison with the experimental results by Sammut. Each value except IBRL3 is the averaged number of trials of five runs. The value of IBRL3 is of 100 runs where $\rho=0.3$.

	BOXES	AHC	CART	IBRL3
Case1	225	90	13	42.43
Case2	837	2562	> 58	47.73

87.22，最大で624(設定1)となったが，実際，25 MHzのSPARCIUとFPUを持つSPARC Station IPCを用いた実験では，実例数が376の場合で1ステップ当りのCPU時間は3.03 msであった．シミュレーションステップが20 msであることを考えれば，この程度ならば実時間制御に十分適用可能であるといえよう．これに対し，SammutによればCARTは実機に適用するには計算時間がかかりすぎる．

監視船の制御問題は本研究において考案されたものであるため，ほかのアルゴリズムの適用は試みられていないが，問題固有パラメータの調整を行えば，BOXES, AHC, CARTなどでも制御可能と思われる．しかし，BOXESでは状態空間の分割の仕方，AHCでは入力変数の基準化量，CARTでは好ましい状態と判断するための基準など設計者の試行錯誤による多くの調整が必要となろう．

ここで例題として取り上げた二つの問題に共通する特徴は，各ステップにおいて取られる行為によって，失敗が起きるかどうかが決定されるという点にある．すなわち，1試行を一つの問題と考えた場合の解は，出力候補であるアトムの列，例えば倒立振子では「右」「左」2種類の文字からなる一つの文字列となり，結果として失敗を起こさない文字列が，その試行における好ましい解ということになる．二つの問題ともに，そのような解は一般に非常に多く存在するが，とり得る解の中での割合という意味では非常に少ないと考えられる．ただし，ここでの学習は，入力から出力を決める関数を同定する，つまり，入力ベクトルで張られる多次元ユークリッド空間を適切な出力アトムに対応する部分空間に分割するという形式をとっている．ニューラルネットや帰納学習の方法をこの問題に適用する場合，学習効率を良くするために，なるべく単純なネットワーク構造や単純な仮説表現方法をとることが要請され，その結果，なるべく単純な空間分割を採用する方向にバイアスがかかることになる．これは，文字列の意味での解空間を狭める効果をもたらす．倒立振子問題において，ニューラルネットによる方法より実例に基づく方法のほうが，好ましい解を早く見つけられる理由の一つは，ニューラルネットによる方法では早々に刈り取られてしまうような複雑な空間分割も，実例に基づく方法では解として採用される機会が失われず，しかも，複雑な空間分割の中にも十分な数の好ましい解が含まれているためであると考えられる．実際，シミュレーション実験においても，倒れはしないがあまり安定ではない運動パターンに収束する場合がいくつか観察されており，これらは複雑な空間分割に基づくものと思われる．ニューラルネットや帰納学習の方法は，良い解を多く含む部分解空間に探索を絞り込むための適切なバイアスが用意できるような問題には有効であるが，適切なバイアスが用意できない場合，学習はかなり困難になる．これに対して実例に基づく方法は仮説表現のためのバイアスが不要なため，バイアスによって区別することが困難な複雑な解が良い解であるような問題にも有効であろうと考えられる．

6. むすび

実例に基づく学習の枠組みを用いて，倒立振子の学習制御等に適応可能なアルゴリズムを提案し，シミュレーション実験によって，その有効性を確認した．しかし，さらに改良すべき点もいくつか残されている．

実験の設定をさまざまに変化させてみると，極端な場合には当然ながら学習が困難になる．本手法が有効に働く問題領域をさらに一般的かつ詳細に調査することは，今後の重要な課題である．

ここでは，無限の記憶容量を仮定していたが，このような仮定は現実的でない．その解決策として，IBRL1あるいはIBRL2のように記憶容量をあらかじめ制限する方法をとり入れる必要があると考えられるが，その場合，記憶容量と記憶変更時に参照される距離のしきい値ρの関係について，十分考察する必要があると思われる．

IBRL3では負の強化信号のみを用い，それを被るまでのステップ数をできるだけ延ばすことを学習目的としているが，逆に正の強化信号を用いて，それが得られるまでのステップ数をできるだけ短くすることを学習目的とするような問題，例えば，制御時間の最小化問題に適用可能な学習アルゴリズムの開発も，今後の課題である．

その他，冗長なあるいはノイズを含んだ入力データ，環境変化への適応，文脈依存性のある環境などへ適用可能な拡張が今後の課題となろう．

謝　辞

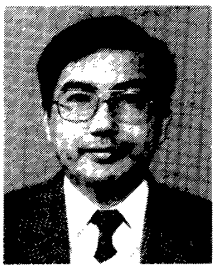
関連研究の調査およびシミュレーション実験の助けをしてくれた，長岡技術科学大学の小池昭彦君，小倉宏明君，上高康博君，正能秀昭君，また，本研究について有益な助言をいただいた計測自動制御学会第14回および第15回知能システムシンポジウム，Workshop on Learning'92，日本機械学会RC-106研究分科会のそれぞれの参加者の方々に感謝いたします．

◇参 考 文 献◇

(1) Samuel, A. L.: Some Studies in Machine Learning Using the Game of Checkers, *IBM J. on Research and Development*, Vol. 3, pp. 210-229 (1959).
(2) Stanfill, C. and Waltz, D.: Toward Memory-Based Reasoning, *Commun. of the ACM*, Vol. 29, pp. 1213-1228 (1986).
(3) 佐藤理史：MBT1：例に基づく訳語選択，人工知能学会誌, Vol. 6, No. 4, pp. 592-600 (1991).
(4) 佐藤理史：MBT2：実例に基づく翻訳における複数翻訳例の組合せ利用，人工知能学会誌, Vol. 6, No. 6, pp. 861-871 (1991).
(5) Waltz, D. L.: Applications of the Connection Machine, *IEEE Computer*, Vol. 20, pp. 85-92 (1987).
(6) Mason, M. T., Christiansen, A. D. and Mitchell, T. M.: Experiments in Robot Learning, *Proc. of the 6th Int. Workshop on Machine Learning*, pp. 141-150 (1989).
(7) Moore, A. W.: Acquisition of Dynamic Control Knowledge for a Robotic Manipulator, *Proc. of the 7th Int. Conf. on Machine Learning*, pp. 244-252 (1990).
(8) Aha, D. W., Kibler, D. and Albert, M. K.: Instance-Based Learning Algorithm, *Machine Learning*, Vol. 6, pp. 37-66 (1991).
(9) Salzberg, S.: A Nearest Hyperrectangle Learning Method, *Machine Learning*, Vol. 6, pp. 251-276 (1991).
(10) 畝見達夫：記憶に基づく離散時系列の学習と環境適応シミュレーションへの応用, *Workshop on Learning 1991*, 北海道札幌市手稲 (1991).
(11) 畝見達夫：実例に基づく強化学習法，人工知能学会誌, Vol. 7, No. 4, pp. 697-707 (1992).
(12) Michie, D. and Chambers, R. A.: Boxes: An Experiment in Adaptive Control, in Dale, E. and Michie, D. eds. *Machine Intelligence 2*, Oliver & Boyd, Edinburgh, pp. 137-152 (1968).
(13) Barto, A. G., Sutton, R. S. and Anderson, C. W.: Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems, *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 13, No. 5, pp. 834-846 (1983).
(14) Selfridge, O. G., Sutton, R. S. and Barto, A. G.: Training and Tracking in Robotics, *Proc. of the 9th Int. Joint Conf. on Artif. Intell.*, pp. 670-672 (1985).
(15) Anderson, C. W.: Strategy Learning with Multilayer Connectionist Representations, *Proc. of 4th Int. Workshop on Machine Learning*, pp. 103-114 (1987).
(16) Connell, M. E. and Utgoff, P. E.: Learning to Control a Dynamic Physical System, *Proc. of the 6th National Conf. on Artif. Insell.*, pp. 456-460 (1987).
(17) Sammut, C.: Experimental Results from an Evaluation of Algorithms that Learn to Control Dynamic Systems, *Proc. of the 5th Int. Conf. on Machine Learning*, pp. 437-443 (1988).
(18) Odetayo, M. O. and McGregor, D. R.: Genetic Algorithm for Inducing Control Rules for a Dynamic System, *Proc. of the 3rd Int. Conf. on Genetic Algorithm*, pp. 177-182 (1989).
(19) Whitley, D., Dominic, S. and Rajarshi, D.: Genetic Reinforcement Learning with Multilayer Neural Networks, *Proc. of the 4th Int. Conf. on Genetic Algorithm*, pp. 562-569 (1991).
(20) 小池昭彦, 小倉宏明, 畝見達夫, 吉谷 豊：遺伝的アルゴリズムによる不安定系の一制御法, 第17回「システムシンポジウム」第14回「知能システムシンポジウム」第1回「ニューラルネットワークシンポジウム」合同シンポジウム資料, 計測自動制御学会 (1991).
(21) Anzai, Y.: Cognitive Control of Real-Time Event-Driven Systems, *Cognitive Science*, Vol. 8, pp. 221-254 (1984).

〔担当編集委員・査読者：小林重信，堀 浩一〕

著 者 紹 介



畝見 達夫（正会員）

1978年東京工業大学制御工学科卒業．1980年同大学院システム科学専攻修士課程修了．1981年同専攻博士後期課程中退後，同専攻助手．1987年長岡技術科学大学計画経営系講師．1992年4月より創価大学情報システム学科講師と国際ファジィ工学研究所客員研究員を兼務．学習制御，帰納学習の理論，人工生命の研究を行っている．共著書「Prologとその応用2」(総研出版)，「インタフェースの科学」(共立出版)，「法律エキスパートシステムの基礎」(ぎょうせい) など．情報処理学会，電子情報通信学会，日本認知科学会，日本ソフトウェア科学会各会員．