論　文

連続力学システムの自動制御のためのオンライン EM 強化学習法*

吉本潤一郎 †§・石井　信 †§・佐藤　雅昭 ‡§

On-Line EM Reinforcement Learning for Automatic Control of Continuous Dynamical Systems*

Junichiro Yoshimoto†‡§, Shin Ishii†§ and Masa-aki Sato‡§

In this paper, we propose a new reinforcement learning (RL) method for dynamical systems that have continuous state and action spaces. Our RL method has an architecture like the actor-critic model. The critic tries to approximate the Q-function, and the actor tries to approximate a stochastic soft-max policy dependent on the Q-function. An on-line EM algorithm is used to train the critic and the actor. We apply this method to two control problems. Computer simulations in two tasks show that our method is able to acquire good control after a few learning trials.

1. はじめに

強化学習は実際の経験と報酬に基づいて自律的に最適行動を獲得する機械学習の手法である．強化学習はゲームの戦略獲得のように有限個の状態と有限個の行動を持つ様々なマルコフ決定問題に応用され，成功を収めてきた [1,2]．一方で，実世界では人体やロボットの運動制御のように状態および行動の空間がともに連続であるようなシステムに関する問題が多く存在する．これらの問題は以下で述べる理由のために前者の問題に比べてはるかに難しい．第一に，システムの状態数および行動数が少ない時には評価関数をテーブルを用いて表現できるが，状態および行動の空間が連続である場合には評価関数をテーブルで表現することはできない．この場合，評価関数を表現するためには近似精度と汎化能力に優れた関数近似器が必要である．また，ロボットなどの高次元システムに応用するためには，高速な学習アルゴリズムが必須である．第二に，たとえ評価関数を正確に近似できたとしても，行動の空間が連続であるために，評価関数を最大化する最適行動を求めることが困難である．本論文では，このような連続システムに対する統計的学習法を用いた強化学習法を提案する．

提案する強化学習法は actor-critic モデルに基づくアーキテクチャ [3] を用いる．critic は現在の状態と行動に対して将来的に得られる報酬の累積 (Q 関数) を近似予測する予測器である．ただし，Q 学習 [4] のように最適 Q 関数を近似するのではなく，SARSA[5] のように現在の actor に依存した Q 関数を近似する．actor は制御器であり，critic が近似した Q 関数値を大きくする行動ほど高い確率で選択するような確率的行動関数を近似する．actor と critic はいずれも正規化ガウス関数ネットワーク (Normalized Gaussian network, NGnet)[6] を用いて表現される．NGnet は正規化されたガウス関数により入力空間を滑らかに領域分割し，局所的に線形近似を行うモデルである．また，NGnet は確率モデルとして定式化できるため actor の確率的関数の近似に用いることができる．actor と critic はともに NGnet の確率モデルに基づいてオンライン EM アルゴリズムによって学習が行われる．

我々の学習法は元の actor-critic モデルのものと以下のように異なる．元の actor-critic モデルにおける学習スキームでは，期待報酬の予測値に対する時間的な誤差

* 原稿受付　2002 年 6 月 19 日

† 奈良先端科学技術大学院大学　情報科学研究科　Graduate School of Information Science, Nara Institute of Science and Technology; 8916-5 Takayama-cho, Ikoma-shi, Nara 630-0101, JAPAN

‡ ATR 人間情報科学研究所　ATR Human Information Science Laboratories; 2-2-2 Seika-cho, Soraku-gun, Kyoto 619-0237, JAPAN

§ 科学技術振興事業団　CREST 銅谷プロジェクト　CREST Doya Project, Japan Science and Technology Corporation

Key Words: reinforcement learning, normalized Gaussian network, stochastic model, EM algorithm, actor-critic model.

(TD 誤差) を用いて，critic は現在の状態に対する期待報酬 (評価関数) を，actor は可能な行動に対する選択確率を近似していた [3]．この学習法は，各状態において選択可能な行動数が少数であることを想定し，テーブルを用いて各状態に対する評価関数や行動選択確率を表現するために，連続的な行動空間を持つ課題への適用が困難である．この問題点に対する一つの解決策として，関数近似器を導入し，TD 誤差に関する確率勾配法に基づいて学習を行う手法が提案されている [7]．Doya らはこの枠組で NGnet を用いた連続システムのための actor-critic モデルを提案している [8,9]．これに対して，我々の手法ではオンライン EM アルゴリズム [6] に基づく学習法を提案する．オンライン EM アルゴリズムは確率勾配法よりも高速な学習アルゴリズムであり，時間とともに入出力分布が変化するような動的な環境においても有効であることが示されている [6]．actor や critic は相互依存であるため，それらに対する関数近似も動的な環境における問題であると考えられ，オンライン EM アルゴリズムが有効に働くと期待できる．

さらに，我々の以前の研究 [10,11] とも以下のように異なる．以前の手法では，actor のための学習信号を critic の勾配信号に基づいて生成することにより，現在の行動よりも良い行動を学習するというヒューリスティクス [12] が用いられていた．これは本質的には勾配法と同じであり，オンライン EM アルゴリズムの真に有効な適用になっていなかった．一方で，新しく提案する手法では actor の近似対象を Q 関数に依存する確率分布とし，これを推定するための新たな手法を導入する．これはオンライン EM アルゴリズムを修正することによって実現することができる．

新しく提案する強化学習法の性能を調べるために，本手法を二つの最適制御問題に応用した．計算機シミュレーションの結果として，いずれの課題においても本手法が少ない試行回数から良い制御を獲得できることが示された．

2. NGnet とオンライン EM アルゴリズム

NGnet[6] は N 次元入力ベクトル x を D 次元出力ベクトル y へ変換するモデルで以下で定義される．

$$y=\sum_{i=1}^{M}\left(\frac{G_i(x)}{\sum_{j=1}^{M}G_j(x)}\right)(W_ix+b_i) \tag{1a}$$

$$G_i(x)\equiv(2\pi)^{-N/2}|\Sigma_i|^{-1/2}\times\exp\left[-\frac{1}{2}(x-\mu_i)'\Sigma_i^{-1}(x-\mu_i)\right] \tag{1b}$$

ここで，M は NGnet を構成するユニットの数，$i\in\{1,...,M\}$ はユニット番号，プライム記号 ($'$) は転置を表している．$G_i(x)$ は，N 次元中心ベクトル μ_i と $(N\times N)$ 次共分散行列 Σ_i を持つ N 次元ガウス関数である．W_i と b_i はそれぞれ $(D\times N)$ 次線形回帰行列と D 次元バイアスベクトルである．以下では，$\tilde{W}_i\equiv(W_i,b_i)$ と $\tilde{x}'\equiv(x',1)$ の表記法を用いる．

NGnet は入力と出力の組 (x,y) を確率事象とする確率モデルの出力期待値として定式化できる [13]．各事象に対して一つのユニットが選ばれるものと仮定し，選ばれたユニットの番号 i を隠れ変数とみなす．このとき，確率モデルは (x,y,i) の組に対する確率分布 $P(x,y,i)$ を以下のように与えることによって定義される．

$$P(x,y,i|\theta)=M^{-1}G_i(x)(2\pi)^{-D/2}\sigma_i^{-D}\times\exp\left[-\frac{1}{2\sigma_i^2}(y-\tilde{W}_i\tilde{x})^2\right] \tag{2}$$

ここで，$\theta\equiv\{\mu_i,\Sigma_i,\sigma_i,\tilde{W}_i|i=1,...,M\}$ はモデルパラメータである．この確率分布から入力 x が与えられた時の出力 y の期待値 $E_\theta[y|x]\equiv\int yP(y|x,\theta)dy$ が求められ，これは NGnet の出力と一致する．すなわち，確率分布 (2) は NGnet の確率モデルを定義している．以下では，(1) 式に基づいて決定論的な出力を与えるモデルを決定論的 NGnet とよび，確率分布 (2) 式に基づいて確率的な出力を与えるモデルを確率的 NGnet とよぶ．

T 個の観測データ $(X,Y)\equiv\{(x(t),y(t))\mid t=1,...,T\}$ が与えられると，モデルパラメータ θ は EM アルゴリズム [14] を用いた最尤推定法によって決定することができる．EM アルゴリズムでは，以下の E ステップと M ステップを繰り返すことによってモデルパラメータの最尤推定量が漸近的に求まる．

E(Expectation) ステップでは，現在のモデルパラメータ $\bar{\theta}$ を用いて，各観測データ $(x(t),y(t))$ に対して i 番目のユニットが選ばれる事後確率を以下で求める．

$$P(i|x(t),y(t),\bar{\theta})=\frac{P(x(t),y(t),i|\bar{\theta})}{\sum_{j=1}^{M}P(x(t),y(t),j|\bar{\theta})} \tag{3}$$

事後確率 (3) を用いて，完全事象に対する期待対数尤度 $L(\theta|\bar{\theta},X,Y)$ は以下で定義される．

$$L(\theta|\bar{\theta},X,Y)=\sum_{t=1}^{T}\sum_{i=1}^{M}P(i|x(t),y(t),\bar{\theta})\times\log P(x(t),y(t),i|\theta) \tag{4}$$

M(Maximization) ステップでは，$L(\theta|\bar{\theta},X,Y)$ を θ について最大化する．最大化の必要条件である $\partial L/\partial\theta=0$ の解は事後確率に関する重み付き平均 $\langle1\rangle_i(T)$，$\langle x\rangle_i(T)$，$\langle xx'\rangle_i(T)$，$\langle y\tilde{x}'\rangle_i(T)$，$\langle|y|^2\rangle_i(T)$ を用いて求めることができる [13]．ここで，$\langle\cdot\rangle_i(T)$ は事後確率 (3) に関する重み付き平均であり，以下で定義される．

$$\langle f(x,y)\rangle_i(T)\equiv\frac{1}{T}\sum_{t=1}^{T}f(x(t),y(t))P(i|x(t),y(t),\bar{\theta}) \tag{5}$$

上記の EM アルゴリズムはバッチ学習であり，モデル

パラメータはすべての観測データが与えられた後で更新される．以下では，逐次的にデータが与えられ，その度ごとにモデルパラメータを更新できるオンライン EM アルゴリズム [6] を示す．

t 番目の観測データが与えられた後のモデルパラメータの推定値を $\theta(t)$ とする．E ステップでは，直前までのデータに基づくモデルパラメータ $\theta(t-1)$ を用いて，t 番目のデータに対する事後確率 $P_i(t) \equiv P(i|x(t),y(t),\theta(t-1))$ を (3) 式にしたがって求める．M ステップでは，事後確率に関する重み付き平均 (5) が以下で置き換えられる．

$$\langle\langle f(x,y)\rangle\rangle_i(t) \equiv \eta(t)\sum_{\tau=1}^{t}\left(\prod_{s=\tau+1}^{t}\lambda(s)\right) \times f(x(\tau),y(\tau))P_i(\tau) \tag{6}$$

パラメータ $\lambda(s)$ $(0 \le \lambda(s) \le 1)$ は過去の良くない推定値による効果を徐々に忘却するための忘却係数である．$\eta(t) \equiv \left[\sum_{\tau=1}^{t}\left(\prod_{s=\tau+1}^{t}\lambda(s)\right)\right]^{-1}$ は正規化係数であり，学習係数のような役割を果たしている．重み付き平均 (6) 式は，以下の逐次計算式を用いて求めることができる．

$$\langle\langle f(x,y)\rangle\rangle_i(t) = (1-\eta(t))\langle\langle f(x,y)\rangle\rangle_i(t-1) + \eta(t)f(x(t),y(t))P_i(t) \tag{7a}$$

$$\eta(t) = (1+\lambda(t)/\eta(t-1))^{-1} \tag{7b}$$

(7) 式で求められた新しい重み付き平均を用いて，モデルパラメータは以下の逐次計算式によって更新される．

$$\mu_i(t) = \langle\langle x\rangle\rangle_i(t)/\langle\langle 1\rangle\rangle_i(t) \tag{8a}$$

$$\tilde{\Lambda}_i(t) = \frac{1}{1-\eta(t)}\left[\tilde{\Lambda}_i(t-1) - \Phi_i(t)\right] \tag{8b}$$

$$\Phi_i(t) \equiv \frac{P_i(t)\tilde{\Lambda}_i(t-1)\tilde{x}(t)\tilde{x}'(t)\tilde{\Lambda}_i(t-1)}{(1/\eta(t)-1)+P_i(t)\tilde{x}'(t)\tilde{\Lambda}_i(t-1)\tilde{x}(t)}$$

$$\tilde{W}_i(t) = \tilde{W}_i(t-1) + \eta(t)P_i(t) \times \left(y(t) - \tilde{W}_i(t-1)\tilde{x}(t)\right)\tilde{x}'(t)\tilde{\Lambda}_i(t) \tag{8c}$$

$$\sigma_i^2(t) = \frac{\langle\langle |y|^2\rangle\rangle_i(t) - \mathrm{Tr}\left(\tilde{W}_i(t)\langle\langle \tilde{x}y'\rangle\rangle_i(t)\right)}{D\langle\langle 1\rangle\rangle_i(t)} \tag{8d}$$

ここで，$\mathrm{Tr}(\cdot)$ は対角和を表す．$\tilde{\Lambda}_i(t) \equiv [\langle\langle \tilde{x}\tilde{x}'\rangle\rangle_i(t)]^{-1}$ は $\Sigma_i^{-1}(t)$ を求めるための補助変数である．$\Sigma_i^{-1}(t)$ は以下の $\tilde{\Lambda}_i(t)$ との関係式から求められる．

$$\tilde{\Lambda}_i(t)\langle\langle 1\rangle\rangle_i(t) = \begin{pmatrix} \Sigma_i^{-1}(t) & -\Sigma_i^{-1}(t)\mu_i(t) \\ -\mu_i'(t)\Sigma_i^{-1}(t) & 1+\mu_i'(t)\Sigma_i^{-1}(t)\mu_i(t) \end{pmatrix} \tag{9}$$

忘却係数 $\lambda(t)$ を $\lambda(t) \overset{t\to\infty}{\longrightarrow} 1$ となるようなあるスケジューリングを行うとオンライン EM アルゴリズムは最尤推定量を求めるための確率近似法になっていることを示すことができる [6]．

また，データの入出力分布に応じてユニットを効率良く配置するために，ユニットの動的操作の機構を導入する．$P(x(t),y(t)|\theta(t-1))$ がある閾値より小さい時，現在の確率モデルを使ってそのデータを説明するのは困難であるので，新しいユニットを生成する．重み付き平均 $\langle\langle 1\rangle\rangle_i(t)$ がある閾値より小さい時，ユニット i はほとんど使われていないことを意味するので削除する．この動的操作機構は強化学習のように近似の対象となる関数が時間とともに変化する環境において有効である [6]．また，データが高次元空間の一部に局在しているような場合に効率の良い関数近似を可能にすると考えられる．

3. オンライン EM 強化学習法

本節では，オンライン EM アルゴリズムを用いた新しい強化学習法を提案する．我々の強化学習法は状態および行動の空間が連続である決定論的力学系の最適制御問題を想定している．学習システムは各時刻において対象システムの状態を観測でき，各状態と各行動の組に対して直接報酬とよばれるスカラー値が与えられるものと仮定する．また，制御対象のダイナミクスに関する事前知識はないものとする．

3.1 問題設定と actor-critic アーキテクチャ

学習システムは現在の状態 $x_c(t)$ を観測すると policy とよばれる行動関数 $\Omega(\cdot)$ にしたがって行動 $u(t)$ を決定する．すなわち，$u(t) = \Omega(x_c(t))$ である．その後，現在の状態 $x_c(t)$ は行動 $u(t)$ と制御対象のダイナミクスにしたがって決定論的に $x_c(t+1)$ へと変化するものとする．この時，学習システムには直接報酬 $r(x_c(t),u(t))$ が与えられる．強化学習の目的は以下で定義される期待報酬を最大化する policy $\Omega(\cdot)$ を求めることである．

$$V^{\Omega}(x_c) \equiv \sum_{t=0}^{\infty}\gamma^t r(x_c(t),\Omega(x_c(t)))\Big|_{x_c(0)=x_c} \tag{10}$$

ここで，$\gamma(0<\gamma<1)$ は減衰係数である．$V^{\Omega}(x_c)$ は policy $\Omega(\cdot)$ に対する評価関数とよばれている．

我々が提案する学習法は actor-critic モデル [3] に基づくアーキテクチャを用いる．actor は現在の状態に対して制御信号を出力する制御器である．critic は将来にわたって得られる直接報酬の累積値 (期待報酬) を予測する予測器である．しかしながら，我々の学習スキームは以下で説明するように元の actor-critic モデルとは大きく異なる．

3.2 critic の学習

critic は以下で定義される Q 関数を学習近似する．

$$Q^{\Omega}(x_c,u) = r(x_c,u) + \gamma V^{\Omega}(x_c^+) \tag{11}$$

ここで，x_c^+ は $x_c = x_c(t)$ と $u = u(t)$ を仮定した時の次の状態 $x_c(t+1)$ を表している．$Q^{\Omega}(x_c,u)$ は，現在の状態 x_c に対しては行動 u を選択し，かつ，以降の状態に対しては常に policy $\Omega(\cdot)$ にしたがって行動を選択した場合

の期待報酬を表している．(10) 式と (11) 式より，policy $\Omega(\cdot)$ が決定論的である場合には評価関数 $V^{\Omega}(\cdot)$ と Q 関数の間に以下の関係が成り立つ．

$$V^{\Omega}(x_c) = Q^{\Omega}(x_c, \Omega(x_c)) \tag{12}$$

(10)，(11)，(12) 式より，Q 関数は任意の $x(t)$ と $u(t)$ に対して，$\Omega(\cdot)$ に依存した Bellman 方程式とよばれる以下の条件式を満たさなければならない [15]．

$$\begin{aligned} Q^{\Omega}(x_c(t),u(t)) = r(x_c(t),u(t)) \\ +\gamma Q^{\Omega}(x_c(t+1),\Omega(x_c(t+1))) \end{aligned} \tag{13}$$

critic は決定論的 NGnet を用いて表現され，(13) 式の Bellman 方程式を満たすようにオンライン EM アルゴリズムを用いて学習が行われる．

3.3 actor の学習

actor は観測された状態 x_c に対して行動 u を決定するための policy を学習近似する．もし，critic NGnet によって表現されている $Q(x_c,u)$ が現在の policy Ω を用いた時の $Q^{\Omega}(x_c,u)$ を正確に表現していると仮定すると，policy Ω より良い policy Ω_{new} は以下で与えられる．

$$\Omega_{new}(x_c) = \underset{u}{\operatorname{argmax}} Q(x_c,u), \quad \text{for any } x_c \tag{14}$$

しかしながら，critic NGnet は非線形モデルであるために，(14) 式の右辺を求めることは困難である．また，学習初期のように critic NGnet によって表現されている関数が真の Q^{Ω} とかなり異なることが予測される場合には，様々な状態・行動空間を探索することによって，critic の学習を促す必要がある．前者の問題点については，Q 関数の勾配情報や TD 誤差情報を用いて逐次的に policy を改善する手法が提案されている [12,8] が，本質的には山登り法となるため Q 関数が単峰でない場合には悪い局所解に陥る危険性がある．さらに，この手法を用いて空間探索の問題点を解決するためには，行動選択の際に適当なノイズを付加するといったヒューリスティクスを導入する必要がある．一方で，以下で提案する手法では，Q 関数に依存した soft-max policy とよばれる確率的 policy を定義し，この確率分布を近似する確率的 NGnet を actor として用いることによって，上記の二つの問題点の解決を試みる．

soft-max policy π は与えられた状態 x_c に対して制御信号 u が選択される条件付き確率を以下のように与えることによって定義される．

$$\pi(u|x_c) = \frac{\exp[\beta Q(x_c,u)]}{\int du \exp[\beta Q(x_c,u)]} \tag{15}$$

ここで，β $(0<\beta<\infty)$ は逆温度とよばれるパラメータである．policy π は，$\beta=0$ の時にはすべての行動が等確率で選択される探索的な policy となり，β が大きい値を取るにつれて $Q(x_c,u)$ が大きな値を持つ行動 u ほど高い確率で選択するグリーディな policy となる．特に，逆温度が $\beta\to\infty$ の極限では (14) 式の決定論的 policy と等価になる．したがって，β を適切に調節することによって前述した二つの問題を同時に解決することができる．一般に (15) 式の分母の積分計算は困難であるが，確率的 NGnet $P(x_c,u|\theta)$ [1] を用いて以下のように近似することができる．

条件付き確率 (15) は状態と行動の組 (x_c,u) に対する以下の同時確率分布から導き出すことができる (**付録 1.**)．

$$P_{Q,\rho}(x_c,u) = \frac{\exp[\beta Q(x_c,u)]\rho(x_c)}{Z_{Q,\rho}} \tag{16}$$

ここで，$\rho(x_c)$ は状態 x_c に関する未知の確率分布である．$Z_{Q,\rho} \equiv \int dx_c du \exp[\beta Q(x_c,u)]\rho(x_c)$ は正規化係数である．このことから，actor を表現する確率的 NGnet は (16) 式で定義される同時確率分布の近似を行う．

確率的 actor NGnet $P(x_c,u|\theta)$ と確率分布 $P_{Q,\rho}(x_c,u)$ の間の距離を以下の KL-divergence で与える．

$$\begin{aligned} KL(\theta) &\equiv \int dx_c du P_{Q,\rho}(x_c,u) \log\left[\frac{P_{Q,\rho}(x_c,u)}{P(x_c,u|\theta)}\right] \\ &= -\int dx_c du P_{Q,\rho}(x_c,u)\log P(x_c,u|\theta) \\ &\quad +(\theta\text{-}independent\ term) \end{aligned} \tag{17}$$

$KL(\theta)$ は，$P=P_{Q,\rho}$ のとき，すなわち，確率的 actor NGnet が soft-max policy (15) を表しているときに最小となる．したがって，actor NGnet の学習の目的は $KL(\theta)$ の最小化，すなわち，以下の目的関数を actor のモデルパラメータ θ に関して最大化することである．

$$J(\theta) \equiv \int dx_c du P_{Q,\rho}(x_c,u)\log P(x_c,u|\theta) \tag{18}$$

確率的 actor NGnet には隠れ変数 i が存在する．このとき，隠れ変数に関する事後分布を用いた以下の期待目的関数を定義する．

$$\begin{aligned} JC(\theta|\bar{\theta}) \equiv \int dx_c du P_{Q,\rho}(x_c,u) \sum_{i=1}^{M} P(i|x_c,u,\bar{\theta}) \\ \times \log P(x_c,u,i|\theta) \end{aligned} \tag{19}$$

$JC(\theta|\bar{\theta})$ の積分は以下の方法を用いて近似することができる．状態と行動の軌道 $\mathcal{X} \equiv \{(x_c(t),u(t))|t=1,...,T\}$ が分布 $\rho(x_c)$ とパラメータ θ_0 を持つ固定された確率的 actor NGnet から独立に得られたものと仮定する．すなわち，$(x_c,u) \sim \rho(x_c)P(u|x_c,\theta_0)$ である．このとき，任意の関数 $f(x_c,u)$ の期待値は以下のデータ平均を用いて近似することができる．

$$E[f(x_c,u)] \equiv \int dx_c du \rho(x_c) P(u|x_c,\theta_0) f(x_c,u)$$

[1] ここで，(2) 式における x と y は，それぞれ x_c と u に置き換えられている．

$$\approx \frac{1}{T}\sum_{t=1}^{T} f(x_c(t),u(t)) \tag{20}$$

(20) 式は $T\to\infty$ の極限では等式となる．(16) 式，(19) 式，(20) 式より，$JC(\theta|\bar{\theta})$ は観測された $\mathcal{X}$ と現在の Q 関数を用いて以下で近似できる．

$$JC(\theta|\bar{\theta},\mathcal{X}) \approx \frac{1}{T}\frac{1}{Z_{Q,\rho}}\sum_{t=1}^{T}\sum_{i=1}^{M}\bar{h}(i|x_c(t),u(t),\bar{\theta}) \times\log P(x_c(t),u(t),i|\theta) \tag{21a}$$

$$\bar{h}(i|x_c,u,\bar{\theta}) \equiv \frac{P(i|x_c,u,\bar{\theta})}{P(u|x_c,\theta_0)}\exp[\beta Q(x_c,u)] \tag{21b}$$

ここで，(21a) 式は (4) 式と類似していることに注意する．(4) 式の事後確率に対応する係数 $\bar{h}(i|x_c(t),u(t),\bar{\theta})$ は現在のモデルパラメータ $\bar{\theta}$，現在の critic の出力値，および，観測軌道 $\mathcal{X}$ を生成したモデルパラメータ θ_0 を持つ actor NGnet を用いて求めることができる．また，$JC(\theta|\bar{\theta})$ の増加が $J(\theta)$ の増加の十分条件となっていること，すなわち，$JC(\theta|\bar{\theta})\geq JC(\bar{\theta}|\bar{\theta})\Rightarrow J(\theta)\geq J(\bar{\theta})$ を示すことができる (**付録 2.**)．したがって，元の EM アルゴリズムと類似したスキームで $J(\theta)$ の最大化を行うことができる．

以上の議論より，確率的 actor NGnet のための EM アルゴリズムを以下のように定義する．E ステップでは，(21b) 式によって $\bar{h}(i|x_c(t),u(t),\bar{\theta})$ が求められ，それを用いて (21a) 式で $JC(\theta|\bar{\theta},\mathcal{X})$ を計算する．ここで，$JC(\theta|\bar{\theta},\mathcal{X})$ の θ に関する最大化には $Z_{Q,\rho}$ の実際の値は不必要である点に注意する．M ステップでは，$JC(\theta|\bar{\theta},\mathcal{X})$ を θ に関して最大化する．最大化の必要条件 $\partial JC(\theta|\bar{\theta},\mathcal{X})/\partial\theta=0$ の解は，重み付き平均 (5) 式を以下で置き換えることによって元の EM アルゴリズムと同様に得られる．

$$\langle f(x_c,u)\rangle_i(T) \equiv \frac{1}{T}\sum_{t=1}^{T} f(x_c(t),u(t)) \times\bar{h}(i|x_c(t),u(t),\bar{\theta}) \tag{22}$$

$J(\theta)$ は上記の E ステップと M ステップを繰り返すことによって最大化することができる．

また，このアルゴリズムは元の EM アルゴリズムと同様に以下のようにオンライン学習へ拡張することができる．状態と行動の組 $(x_c(t),u(t))$ が観測されると E ステップでは $\bar{h}(i|x_c(t),u(t),\theta(t-1))$ が求められる．M ステップでは，逐次計算式 (7a) が以下で置き換えられる．

$$\langle\langle f(x_c,u)\rangle\rangle_i(t) = (1-\eta(t))\langle\langle f(x_c,u)\rangle\rangle_i(t-1) +\eta(t)f(x_c(t),u(t))\bar{h}_i(t) \tag{23}$$

ここで，$\bar{h}_i(t)\equiv\bar{h}(i|x_c(t),u(t),\theta(t-1))$ である．この重み付き平均を用いて，新しいモデルパラメータ $\theta(t)$ が (8) 式と (9) 式を用いて更新される．

オンライン学習の場合に注意しなければならないのは，軌道 $\mathcal{X}$ をサンプリングするためのモデルパラメータ θ_0 も時間とともに変化することである．このため，θ_0 の時間変化が早い場合は (21) 式の近似が悪くなる．この問題点を回避する一つの方法は，学習係数 $\eta(t)$ を小さな値に設定することによって θ_0 の変化を遅くすることであるが，これは actor の学習の進行を遅くすることと同義である．一方，一回の学習エピソードでサンプリングされる軌道データ数がそれほど多くない場合には，エピソードの軌道生成の際には固定した θ_0 を用いて制御し，エピソード後にその軌道データを用いて学習しても，それほど計算資源を必要としない．このセミオンライン的な学習方法によれば，エピソード内では，critic は固定した policy に依存する Q 関数を学習することができ，一方で actor はエピソード後に固定された Q 関数を用いて学習を行うことができるため，安定した学習となる．そこで，後述の実験では後者のセミオンライン的な学習法を用いている．

また，逆温度パラメータ β は適当なスケジューリングを行う必要がある．これは，小さな β を用いると空間探索はより実現できるが，一方でランダム性の大きな policy となるため学習の安定性が悪くなり，逆に大きな β を用いると policy はグリーディであるため学習は安定するが，critic の正しい推定が行えなくなるからである．学習エピソードの進行に伴って critic の推定が良くなっていくことが期待されるので，学習エピソード k における逆温度パラメータ $\beta(k)$ が徐々に増加するように $\beta(k)=ak+b$ とスケジューリングする．ここで，a と b は適当な正数である．このスケジューリングはヒューリスティクスであるが，後述の実験で示すようにうまく働く．

3.4 actor-critic 学習

以上で提案した actor および critic の学習アルゴリズムは以下のようにまとめられる．

1. 制御と critic の学習

1-1. 現在の状態 $x_c(t)$ に対して，actor NGnet は確率分布 (2) 式に基づいて確率的行動 $u(t)$ を選択する．具体的には，現在の actor NGnet のモデルパラメータを θ_0 とすると，ユニット i が条件付き確率 $P(i|x_c,\theta_0)$ で選択される．その後，選択された i について行動 u が条件付き確率 $P(u|x_c,i,\theta_0)$ で生成される．

1-2. 行動 $u(t)$ と制御対象のダイナミクスにしたがって状態は $x_c(t+1)$ に変化する．この時，学習システムは報酬 $r(x_c(t),u(t))$ を観測する．

1-3. オンライン EM アルゴリズムを用いて critic NGnet の学習が行われる．critic NGnet への入力は状態と行動の組 $(x_c(t),u(t))$ である．出力のターゲットは (13) 式の右辺である．この計

算で必要な $\Omega(x_c(t+1))$ は現在の actor NGnet $P(u(t)|x_c(t+1),\theta_0)$ の条件付き期待値，すなわち，決定論的 NGnet の出力 (1) 式によって与えられる．決定論的 NGnet の出力を用いるのは，学習後の性能評価時には空間探索が必要ないためである．決定論的 actor NGnet の出力と現在の決定論的 critic NGnet を用いて $Q(x_c(x+1),\Omega(x_c(t)))$ を求めることができる．

固定された actor NGnet $P(u|x_c,\theta_0)$ を用いて，$t=1,\cdots,T$ の間，この過程を繰り返す．これを 1 回のエピソードと定義する．1 エピソードの間に観測された状態と行動の軌道 $\mathcal{X}\equiv\{(x_c(t),u(t))|t=1,...,T\}$ は保存される．

2. actor の学習

actor NGnet は，保存された軌道 $\mathcal{X}$ から 3.3 節で説明したオンライン EM アルゴリズムを用いてそのモデルパラメータを更新する．

以上が，1 エピソードの力学システムの動作に基づき行われる処理である．この学習エピソードを繰り返すことによって強化学習が行われる．

4. 実験

提案手法の性能を調べるために，本手法を二つの最適制御問題に応用した．第 1 の問題は，単振子を限られた制御トルクを用いて振り上げ，倒立位置で安定化させるものである [8]．システムの状態は $x_c\equiv(q,\dot{q})'$ で表される．ここで，q は頂点位置から計った振子の角度であり，$\dot{q}$ は振子の角速度である．制御トルク u は $|u|\le u_{max}$ に制限されており，u_{max} は振子を一度に倒立位置まで持ち上げるには不十分な量であるものとする．

1 回の学習エピソードの間に以下の過程が行われる．振子の状態を目標状態を中心とする正規分布にしたがって初期化した後，学習システムは時間間隔 0.01 秒でシステムの状態を観測し，確率的 actor にしたがって制御信号を出力する．この制御過程は 7 秒間行われ，この間に 3.4 節で述べられた学習アルゴリズムが適用される．すなわち，1 エピソードで観測される軌道データは 700 点である．

学習の目的は目標状態 $q=\dot{q}=0$ で安定化させることであるので，最短時間で安定化するための直接報酬は目標状態で 1，それ以外の状態 0 となる関数が望ましい．しかしながら，試行錯誤中に連続空間の一点である目標状態に到達することは極めてまれであり，その状態に到達できなければ，その間の情報は無駄になってしまう．そこで，本実験では，直接報酬として目標状態で最大となるような連続関数を用いる．具体的には，$r(x_c(t),u(t))$ は $\tilde{r}(x_c(t+1))$ で与えられるものとし，これを以下で定義する．

$$\tilde{r}(x_c)=\exp\left(-q^2/\nu_1-\dot{q}^2/\nu_2\right) \tag{24}$$

ここで，ν_1,ν_2 は小さな正定数である．この直接報酬は，$\nu_1=\nu_2=0$ の極限において，目標状態 $q=\dot{q}=0$ の時のみ 1，それ以外で 0 となるものである．

以上の条件でシミュレーション実験を行ったところ，最短の場合には 17 回の学習エピソード後にシステムはほぼすべての初期状態から振子を倒立位置で静止させることができた．Table 1 は，各初期状態集合から学習後の actor ネットワークを用いて制御した場合の成功率を表している．ここで，課題の成功は最後に得られた報酬が 0.99 以上になることで定義している．この報酬を得るためにはシステムは倒立位置付近で静止していなければならない．また，初期状態集合は各範囲内からランダムに 1000 個を生成したものを用いている．(1) や (2) のように初期状態が比較的簡単な場合にはすべて成功することができる．(3) のように比較的難しい初期状態が与えられた場合でも高い割合で成功する．例えば，垂れ下がりの位置で静止している状態からでも成功することができる．Fig. 1 はこの時の振子の動きをストロボ的に表している．

第 2 の問題は，acrobot を倒立位置付近で安定化させるものである [5,11]．acrobot は Fig. 2 で示されるような 2 リンク 2 関節からなるアクチュエータロボットであり，鉄棒運動のダイナミクスと類似している．腰の部分に対応する第 2 関節にトルクをかけることができるが，鉄棒を持つ手の部分に対応する第 1 関節にトルクをかけることはできない．acrobot は，システムが非線形性が強く不安定な力学系であるため，倒立位置近傍での安定化を行うだけでも非常に難しい．

しかしながら，倒立位置近傍を初期状態として強化学習を行った場合には，最短で 16 回の学習エピソード後に目標状態で安定化できる．Fig. 3 と Fig. 4 は学習後の典型的な制御過程を示している．Fig. 3 は acrobot の状態をストロボ的に表したものであり，Fig. 4 はこのときの状態と制御トルクの時系列を示したものである．システムは第 2 リンクの振動を徐々に減衰させ，最後には頂点位置で静止させる．また，性能評価時には Fig. 5 に示すように目標状態から大きく離れた初期状態を与えられた場合でも，反動をつけて倒立位置付近まで導き，最後は目標状態で安定化することができる．これは critic NGnet による Q 関数の推定の伝播が正しく行われ，かつ，NGnet の汎化性能の良いことを示すものであると考えられる．

我々は以前の研究 [10,11] で同じ問題を用いた実験を行っている．ここでは，critic の勾配に基づいて actor の学習が行われていた．倒立振子および acrobot の問題における以前の手法との比較は，Table 2 と Table 3 にそれぞれまとめられる．Table 2 および Table 3 の各項目は，良い制御を獲得するまでの平均学習エピソード数，学習後の actor NGnet と critic NGnet が必要とする平均ユニット数，および，1 回の学習エピソードに要する平均

	Range of initial states	Success rate
(1)	$0 \le \|q\| \le \pi/3,\ \|\dot{q}\| \le \pi/3$	1.000
(2)	$\pi/3 \le \|q\| \le 2\pi/3,\ \|\dot{q}\| \le 2\pi/3$	1.000
(3)	$2\pi/3 \le \|q\| \le \pi,\ \|\dot{q}\| \le \pi$	0.940

Table 1　Success rate from each initial state setting

	New	Previous
No. of episodes	96.4	127.8
No. of actor units	57.0	37.4
No. of critic units	130.6	77.8
CPU time [s/episode]	1.96	2.07

Table 2　Comparison of the two methods for inverted pendulum

	New	Previous
No. of episodes	43.4	75.4
No. of actor units	33.4	50.8
No. of critic units	35.0	76.8
CPU time [s/episode]	2.74	2.92

Table 3　Comparison of the two methods for the acrobot

CPU 時間を，それぞれ表している．ここで，各数値は課題に成功した 20 回の平均を示している．また，CPU 時間はワークステーション DEC Alpha 600 au を用いて計測している．新しい手法は良い制御を獲得するまでの学習エピソード数，すなわち，学習に要した総軌道データ数は，以前の手法に比べて大きく減少しており，効率良い学習法であることが分かる．actor や critic に要するユニット数は問題によって傾向が異なるが，1 回の学習エピソードに要する CPU 時間にはそれほど影響を及ぼさない．また，1 回の学習エピソードに要する CPU 時間は物理システムに比べて十分に短いものであり，実問題への適用も可能と考えられる．

提案手法は以前の手法と同様に制御対象のモデルを明に必要とせず，問題の性質に応じて NGnet のユニット数を自動決定できるため，汎用性が高いものである．従来の手法では，勾配法に基づいて actor の出力ターゲットを求めていたため，その更新幅を決定するための係数を問題に応じて慎重に決定する必要があった．しかしながら，提案手法では実際に経験した軌道データをそのまま学習データとして用いることができ，また，逆温度に関するスケジューリングも容易である．この点が改善されているため，提案手法は，以前の手法よりも扱いやすく，優れたものと考えられる．

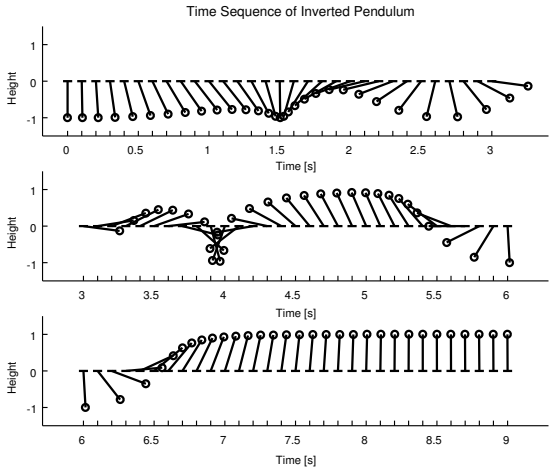


Fig. 1　Control process for the inverted pendulum

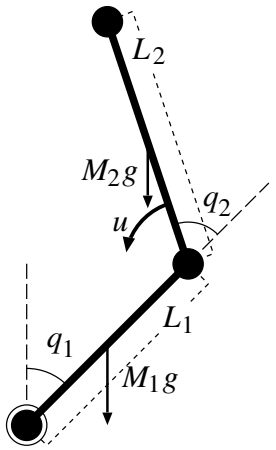


Fig. 2　The acrobot

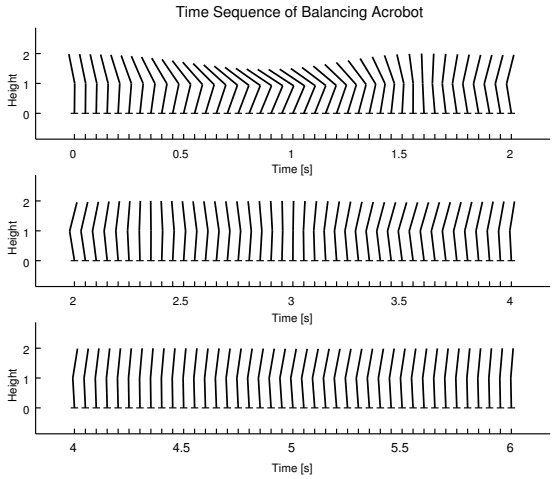


Fig. 3　Control process for the acrobot

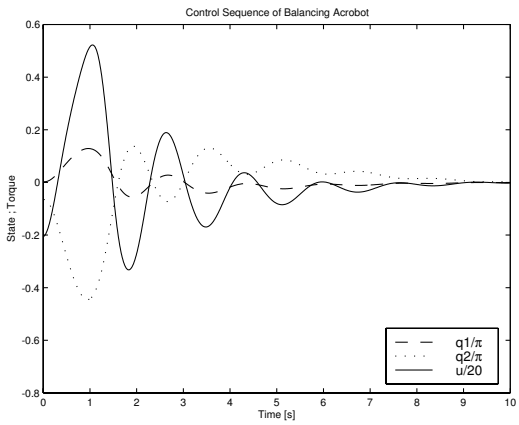


Fig. 4 State/control time-series corresponding to Figure 3

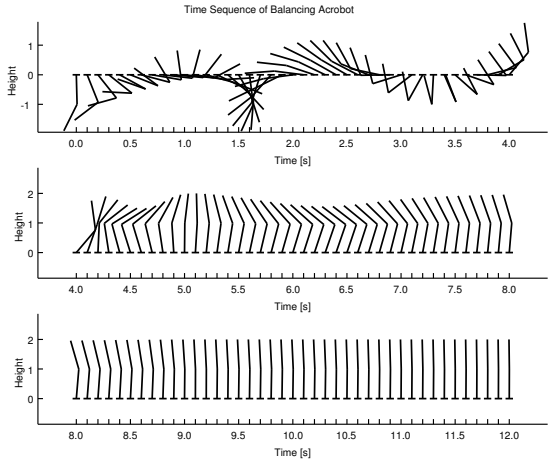


Fig. 5 A successful control from a difficult initial state

5. おわりに

本論文では，actor-critic アーキテクチャを用いた新しい強化学習法を提案した．critic は現在の policy に対する Q 関数を近似した．critic の学習にはオンライン EM アルゴリズムが用いられ，学習データは Bellman 方程式に基づいて生成された．actor は現在の Q 関数に依存する soft-max policy を近似した．actor の学習は経験した状態と行動の軌道および現在の Q 関数を用いてオンライン EM アルゴリズムを用いて行われた．提案された手法は二つの自動制御問題に応用され，いずれの場合にも少ない試行からの学習で良い制御が獲得できた．状態および行動の空間が連続である課題に強化学習を応用するためには，近似精度と汎化能力に優れた関数近似器および高速な学習アルゴリズムが必要である．実験結果から提案された手法はこれらの特徴を有しているものであると考えられる．

謝　辞

本研究は、財団法人テレコム先端技術研究支援センターおよび通信・放送機構の研究委託により実施された．また人工知能研究振興財団の研究支援を受けた．

参 考 文 献

[1] G. Tesauro: Practical issues in temporal difference learning; *Machine Learning*, Vol. 8, pp. 257–277 (1992)

[2] T. Yoshioka, S. Ishii & M. Ito: Strategy acquisition for game othello based on min-max reinforcement learning; *IEICE Transactions on Information and Systems*, Vol. E82-D, No. 12, pp. 1618–1626 (1999)

[3] G. A. Barto, R. S. Sutton, R. S. & C. W. Anderson: Neuronlike adaptive elements that can solve difficult learning control problems; *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, pp. 834–846 (1983)

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

[5] R. S. Sutton: Generalization in reinforcement learning: Successful examples using sparse coarse coding; *Advances in Neural Information Processing Systems 8* (D. S. Touretzky, M. C. Mozer & M. E. Hasselmo, eds.), pp. 1038–1044, MIT Press (1996)

[6] M. Sato & S. Ishii: On-line EM algorithm for the normalized Gaussian network; *Neural Computation*, Vol. 12, No. 2, pp. 407–432 (2000)

[7] D. P. Bertsekas & J. N. Tsitsiklis: *Neuro-Dynamic Programming*, Athena Scientific (1996)

[8] K. Doya: Reinforcement learning in continuous time and space; *Neural Computation*, Vol. 12, pp. 219–245 (2000)

[9] 森本，銅谷：強化学習を用いた高次元連続状態における系列運動学習－起き上がり運動の獲得－；電子情報通信学会論文誌（D-II），Vol. J82-D-II, pp. 2118–2131 (1999)

[10] M. Sato & S. Ishii: Reinforcement learning based on on-line EM algorithm; *Advances in Neural Information Processing Systems 11* (M. S. Kearns, S. A. Solla & D. A. Cohn, eds.), pp. 1052–1058, MIT Press (1999)

[11] 吉本，石井，佐藤：オンライン EM アルゴリズムによる強化学習法の acrobot 制御への応用；電子情報通信学会論文誌（D-II），Vol. J83-D-II, No. 3, pp. 1024–1033 (2000)

[12] D. A. Sofge & D. A. White: Applied learning - optimal control for manufacturing; *Handbook of Intelligent Control* (D. A. White & D. A. Sofge, eds.), pp. 259–282, Van Nostrand Reinhold, New York (1992)

[13] L. Xu, M. I. Jordan & G. E. Hinton: An alternative model for mixtures of experts; *Advances in Neural Information Processing Systems 7* (G. Tesauro, D. S. Touretzky & T. K. Leen, eds.), pp. 633–640, MIT Press (1995)

[14] A. P. Dempster, N. M. Laird & D. B. Rubin: Maximum likelihood from incomplete data via the EM

algorithm; *Journal of Royal Statistical Society B*, Vol. 39, pp. 1–22 (1977)
[15] R. E. Bellman: *Dynamic Programming*, Princeton University Press (1957)

付　　録

付録 1. soft-max policy π の導出

同時確率分布 (16) より，状態 x_c が与えられた時の行動 u が選ばれる条件付き確率は以下で得られる．

$$P_{Q,\rho}(u|x_c) = \frac{P_{Q,\rho}(x_c,u)}{\int du P_{Q,\rho}(x_c,u)} = \frac{\exp[\beta Q(x_c,u)]}{\int du \exp[\beta Q(x_c,u)]} \quad \text{(A1)}$$

これは，soft-max policy π の確率分布と一致する．すなわち，(A1) 式によって soft-max policy π を導出することができる．

付録 2. $JC(\theta|\bar{\theta}) \geq JC(\bar{\theta}|\bar{\theta}) \Rightarrow J(\theta) \geq J(\bar{\theta})$ の略証

モデルパラメータ θ を持つ確率的 NGnet のユニットインデクス i に関する事後確率を $P(i|x_c,u,\theta)$ とし，$\bar{\theta}$ を現在のモデルパラメータとする．このとき，Q 関数と x_c の出現確率 $\rho(x_c)$ に依存する確率分布 $P_{Q,\rho}(x_c,u)$ で重み付けした $P(i|x_c,u,\bar{\theta})$ と $P(i|x_c,u,\theta)$ の KL-divergence を用いて以下のような関係式を導き出すことができる．

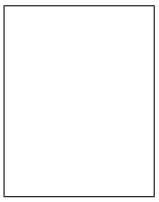
$$\int dx_c du P_{Q,\rho}(x_c,u) \sum_{i=1}^{M} P(i|x_c,u,\bar{\theta}) \times \log\left(\frac{P(i|x_c,u,\bar{\theta})}{P(i|x_c,u,\theta)}\right) = \big[JC(\bar{\theta}|\bar{\theta}) - JC(\theta|\bar{\theta})\big] + \big[J(\theta) - J(\bar{\theta})\big] \geq 0 \quad \text{(A2)}$$

この関係式は，(A2) 式において第 1 項が負 (または 0) のときには第 2 項が正 (または 0) にならなければならないことを意味する．すなわち，$JC(\theta|\bar{\theta}) \geq JC(\bar{\theta}|\bar{\theta}) \Rightarrow J(\theta) \geq J(\bar{\theta})$ である．

(証明終)

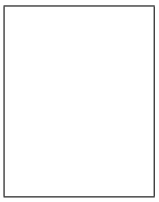
著 者 略 歴

吉本　潤一郎



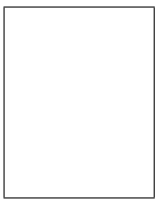
1975 年 9 月 26 日生．2002 年 9 月奈良先端科学技術大学院大学情報科学研究科博士後期課程修了．2002 年 10 月より科学技術振興事業団 CREST 銅谷プロジェクトの研究員となり現在に至る．ニューラルネットワーク，統計的学習理論，強化学習の研究に従事．神経回路学会，計測自動制御学会の会員．

石井　信 (正会員)



1988 年 3 月東京大学大学院工学系研究科修士課程修了．(株) リコー中央研究所研究員，ATR 人間情報通信研究所研究員，奈良先端大情報科学研究科助教授を経て，2001 年 4 月より奈良先端科学技術大学院大学情報科学研究科教授となり現在に至る．非線形力学系，ニューラルネットワーク，強化学習，統計的学習理論，バイオ情報学の研究に従事．電子情報通信学会，神経回路学会などの会員．

佐藤　雅昭



1980 年 3 月大阪大学大学院理学研究科物理学専攻博士課程修了．ニューヨーク大学助手，フロリダ大学助手，ATR 視聴覚機構研究所主任研究員，ATR 人間情報通信研究所主任研究員を経て，現在，ATR 人間情報科学研究所主任研究員．非線形力学系，カオス，ニューラルネットワーク，強化学習，ロボット制御，統計的学習理論の研究に従事．電子情報通信学会，神経回路学会などの会員．