Estimator 型可変階層構造学習オートマトンによる ノイズを含む観測値を用いた大域的最小点探索

最上 義夫*・馬場 則夫**・松下 正樹***

Global Minimum Point Search Under Noisy Observations Using Estimator-Type
Variable Hierarchical Structure Learning Automata

Yoshio MOGAMI*, Norio BABA** and Masaki MATSUSHITA***

The purpose of this paper is to construct a global optimization algorithm of the unknown multimodal objective function under noisy observations. Our algorithm is based on the learning performance of the variable hierarchical structure learning automata, and, in order to reduce the number of iterations, the estimator-type learning algorithm which is the rapidly converging one is used for the learning algorithm of the automata. The numerical experiment is carried out to verify the efficiency of the proposed algorithm, and, from the results, the proposed global optimization algorithm is useful for finding out a global minimum of the unknown multimodal objective function under noisy observations.

Key Words: hierarchical structure learning automata, estimator-type learning algorithm, optimization algorithm, global minimum point, noisy observation

1. ま え が き

未知目的関数の最小値を与える点(最小点)を求める,いわゆる,未知目的関数の最適化問題について,現在までに種々の解法が提案されている.一般によく用いられる方法としては,最急降下法,ニュートン法,共役勾配法,疑似ニュートン法などの勾配法^{1),2)}があるが,これらの方法は,目的関数が多峰性関数である場合には局所的な最小値を与える点(局所的な最小点)に収束してしまい,大域的な最小値を与える点(大域的な最小点)を見出せない場合が存在することが知られている^{1),2)}.そして,この問題点を解決するために,ランダム探索法^{3)~6)}のような確率的探索法やランダム探索法と勾配法を組み合わせた Hybrid Algorithm⁷⁾など種々の探索法が提案されている.

しかし,上記の手法はいずれも目的関数の真値が手に入る場合を対象としており,より一般的な最適化問題を考えるとき,目的関数の真値が得られない場合,すなわち,目的関数の値を観測するときに何らかの観測ノイズが混入する場合も

考慮することが必要である.これは,観測ノイズが存在する場合の多峰性未知目的関数の最適化問題となるわけであるが,本論文では観測ノイズが加法的に加わる場合を考察する.この問題に上述のような探索手法を適用して大域的な最小点を見出すことは,目的関数の真値が得られないことから,基本的に不可能である⁸⁾.そこで,観測ノイズに対する何らかの対策が必要となるわけであるが,一つの方法としては,観測ノイズを除去することによって目的関数の真値を手に入れることが考えられる.しかし,それを行うにはかなりの困難が伴うこともあるであろうし,また,観測ノイズを完全に除去することができないかもしれない.そこで最上,馬場,古澤⁹⁾は,観測ノイズを除去するのではなく,ノイズを含む観測データのような不確実な情報のもとでも,それに基づいて繰り返し学習することによって徐々に最適解を見出していくことが出来るという学習オートマトンの特性を利用して新しい探索アルゴリズムを構築した.そして,この探索アルゴリズムによって,加法的観測ノイズが存在する場合に多峰性未知目的関数の大域的な最小点に非常に近い点を見出すことに成功している.ところで,この探索アルゴリズムにおいては, L_{R-I} 型学習アルゴリズムをもつ可変階層構造学習オートマトンが用いられているが, L_{R-I} 型学習アルゴリズムは現在時刻の出力とそれに対する入力(返答)のみに基づいて学習を行うものであることから,探索回数が多くなってしまうという問題点を有している.しかし,実用上の観点からは,出来るだけ少ない探索回数で大域的な最小点を見出すことが望まし

* 徳島大学 徳島市南常三島町 2-1

** 大阪教育大学 柏原市旭ヶ丘 4

*** 三菱電機(株) 姫路市千代田町 840

* The University of Tokushima, Tokushima

** Osaka Educational University, Kashiwara

*** Mitsubishi Electric Corporation, Himeji
(Received November 30, 1998)
(Revised May 26, 1999)

いことは明らかである。

本論文においても、文献9)と同様、大域的最小点の探索に可変階層構造学習オートマトンを用いるわけであるが、その探索回数の削減を目的として、可変階層構造学習オートマトンの学習アルゴリズムとして、Estimator型学習アルゴリズム^{10)~12)}を採用することにする。この学習アルゴリズムは、学習開始時から現在に至るまでのすべての出力と入力（返答）を学習に用いるもので、 L_{R-I} 型学習アルゴリズムよりも少ない学習回数で最適解を見出すことが知られている¹⁰⁾。

2. 問題の記述

加法的観測ノイズが存在する場合の多峰性未知目的関数の最適化問題は、つぎのように記述される。

多峰性未知目的関数を $f(x): R^n \rightarrow R$ とする。そして、目的関数 $f(x)$ の値は、加法的なノイズを含む観測値 $y(x)$ として、以下のように得られるものとする。

$$y(x) = f(x) + \xi \quad (1)$$

ただし、 $x = (x^1, x^2, \dots, x^n)' \in R^n$ であり、また、 ξ は観測ノイズである。このとき、観測値 $y(x)$ に基づいて、

$$f(x^*) = \min_{x \in D^n} f(x) \quad (2)$$

なる大域的最小点 $x^* \in D^n \subset R^n$ を見出す。ここで、 D^n は大域的最小点の探索を行う閉領域であって、つぎのように与えられる。

$$D^n = [x_L^1, x_U^1] \times [x_L^2, x_U^2] \times \dots \times [x_L^n, x_U^n] \quad (3)$$

ただし、 $[x_L^j, x_U^j]$ ($j = 1, 2, \dots, n$) は変数 x^j の探索区間である。

3. 可変階層構造学習オートマトンの大域的最小点探索への適用

n 次元空間上で定義される未知目的関数の大域的最小点の探索を行うわけであるが、文献9)と同様に、未知目的関数の1つの変数 x^j に1つの可変階層構造学習オートマトン W_j を対応させることにする ($j = 1, 2, \dots, n$)。すなわち、 n 個の可変階層構造学習オートマトン $\{W_1, W_2, \dots, W_n\}$ を用いて探索を行うことになる。

まず、各変数の探索区間を有限個の閉区間（探索小区間と呼ぶ）に分割し、その1つ1つを可変階層構造学習オートマトンのパス^(注1)に対応させる。そして、大域的最小点が含まれていそうな探索小区間をより細かく分割していくことによって、最終的に、大域的最小点を含む小さい幅の探索小区間を見出すことにする。このことを変数 x^j に関して概念的に示すと Fig.1 のようになる：

いま、2動作の学習オートマトンを用いて可変階層構造学習オートマトン W_j を構成するとする（学習オートマトンの動作数は一般には r 個）。すると、変数 x^j の探索区間 $[x_L^j, x_U^j]$

(注1) 階層構造上のレベル1の学習オートマトンから最下レベルの学習オートマトンに至る動作の列をパスと呼ぶ。

Fig.1 Search interval and variable hierarchical structure learning automata

は2個の探索小区間 $[x_L^j, x_2^j] \cup [x_2^j, x_U^j] (= [x_L^j, x_U^j])$ に分割され、Fig.1(a)のように W_j の初期構造 Σ^0 のもとでパス $\phi_1^j = \{\alpha_1^j\}$ に探索小区間 $[x_L^j, x_2^j]$ が、また、パス $\phi_2^j = \{\alpha_2^j\}$ に探索小区間 $[x_2^j, x_U^j]$ が対応する。探索の結果、大域的最小点が含まれていそうな探索小区間は $[x_L^j, x_2^j]$ であると判断したとすると、(b)のようにその区間をさらに2分割し、それに対応するパスに学習オートマトン Z_1 を追加して W_j の構造を Σ^1 に更新する。このときには、パス $\phi_{11}^j = \{\alpha_1^j, \alpha_{11}^j\}$ に探索小区間 $[x_L^j, x_2^j]$ が、パス $\phi_{12}^j = \{\alpha_1^j, \alpha_{12}^j\}$ に探索小区間 $[x_2^j, x_3^j]$ が、そして、パス $\phi_2^j = \{\alpha_2^j\}$ に探索小区間 $[x_3^j, x_U^j]$ が、それぞれ対応する。この構造での探索の結果、大域的最小点が含まれていそうな探索小区間は $[x_2^j, x_3^j]$ であると判断すると、(c)のようにその区間をさらに2分割し、それに対応するパスに学習オートマトン Z_{12} を追加して W_j の構造を Σ^2 に更新して、更なる探索を行う。

一般的には、変数 x^j の探索区間 $[x_L^j, x_U^j]$ の分割はつぎのようになる。

$$[x_L^j, x_U^j] = [x_L^j, x_2^j] \cup [x_2^j, x_3^j] \cup \dots \cup [x_{\Psi^{m_j}+1}^j, x_U^j] \quad (4)$$

ただし、 W_j の構造は Σ^{m_j} (m_j は W_j の構造変更回数) で

あり、そのパスの総数は Ψ^{m_j} とする。そして、ある時刻 $t (= 0, 1, 2, \dots)$ において、 W_j の $l_j (= 1, 2, \dots, \Psi^{m_j})$ 番目のパス $\phi_{i_1^j i_2^j \dots i_{N_{l_j}^j}}^j$ ($N_{l_j}^j$ はこのパスのレベル数) が選択されるということは、 x^j の探索小区間として $[x_{l_j}^j, x_{l_j+1}^j]$ が選ばれたことになる。(例えば、Fig.1(c) の場合、パス $\phi_{121}^j = \{\alpha_1^j, \alpha_{12}^j, \alpha_{121}^j\}$ が選択されたとする、探索小区間として $[x_2^j, x_3^j]$ が選ばれたことになる。) なお、パス $\phi_{i_1^j i_2^j \dots i_{N_{l_j}^j}}^j (= \{\alpha_{i_1^j}^j, \alpha_{i_1^j i_2^j}^j, \dots, \alpha_{i_1^j i_2^j \dots i_{N_{l_j}^j}}^j\})$ が選択されるためには、 W_j

の各レベル $s (= 1, 2, \dots, N_{l_j}^{m_j})$ において、動作確率分布 $(p_{i_1^j i_2^j \dots i_{s-1}^j}^j(t), p_{i_1^j i_2^j \dots i_{s-1}^j}^j(t), \dots, p_{i_1^j i_2^j \dots i_{s-1}^j}^j(t))'$ に従って動作 $\alpha_{i_1^j i_2^j \dots i_{s-1}^j}^j$ が選択されることが必要である^(注2)。

目的関数の観測も文献9)と同様に行う。すなわち、時刻 t において、各変数に対して探索小区間 $\{[x_{l_j}^j, x_{l_j+1}^j] : j = 1, 2, \dots, n\}$ が選択されたとする、これらの探索小区間の直積空間として観測領域 $S_{l_1 l_2 \dots l_n}^{m_1 m_2 \dots m_n}$ を与える。

$$S_{l_1 l_2 \dots l_n}^{m_1 m_2 \dots m_n} = [x_{l_1}^1, x_{l_1+1}^1] \times [x_{l_2}^2, x_{l_2+1}^2] \times \dots \times [x_{l_n}^n, x_{l_n+1}^n] \quad (5)$$

そして、目的関数の観測は、このようにして得られた観測領域の中からランダムに一点 $x = (x^1, x^2, \dots, x^n)'$ を選択しこの点で行うことにする。

ここで、探索小区間に最小幅を設定し、この最小幅を持つ探索小区間を探索の結果見出されるべき目標物とする。そして、これを出力とするパスを目標パスと呼ぶことにすると、本探索アルゴリズムでは、 n 個の可変階層構造学習オートマトンがそれぞれ最小幅の探索小区間(目標パス)に収束したときに探索が終了することになる。そして、 n 個の最小幅の探索小区間の各中点から成る点 $\bar{x} = (\bar{x}^1, \bar{x}^2, \dots, \bar{x}^n)'$ を大域的最小点と決定する(本論文では、この点 $\bar{x}$ を探索最小点と呼ぶ)。ところで、この最小幅をどれくらいに設定するかは、探索最小点 $\bar{x}$ の各要素の決定法に注意すると、探索最小点をどの精度まで求めるかによって定まるものとなる^(注3)。例えば、大域的最小点と探索最小点との誤差 $|x^j - \bar{x}^j|$ ($j = 1, 2, \dots, n$) をすべて 0.05 以下にするためには、探索小区間の最小幅は 0.1 以下にすることが必要である。もちろん、もっと高い精度を要求する場合には探索小区間の最小幅をさらに小さく設定すればよいわけであるが、そのときには階層構造のレベル数が増えて探索回数の増加を招くと考えられる。すなわち、探索精度と探索回数は相互に関連した挙動を示すことになるので、どちらを優先するかは適用する問題によることになる。また、この最小幅から可変階層構造学習オートマトンの

最大レベル数も決定される。すなわち、ある変数の探索区間が $[0, 12]$ であるとし、探索小区間の最小幅は 0.1 とする。そして、その変数に対応させる可変階層構造学習オートマトンを 2 動作を持つ学習オートマトンを用いて構成するならば、その最大レベル数は、 $12 \div 2^7 = 12 \div 128 = 0.094 < 0.1$ から 7 となる。さらにこのことから、あるパスが目標パスかどうかの判定は、そのパスのレベル数が最大レベル数に達しているか否かによって行うことができる。

4. 学習オートマトンへの返答の生成

可変階層構造学習オートマトンを大域的最小点探索に用いる場合には、観測値 $y(x)$ に基づいて学習オートマトンへの返答(成功あるいは失敗)を生成することが必要である。文献9)においては、最新 N 個の観測値の算術平均値である平均観測値 H を用いて、 $y(x) \leq H$ のときには成功の返答 $\beta(t) = 1$ を、 $y(x) > H$ のときには失敗の返答 $\beta(t) = 0$ を生成し、そして、この返答が動作した各レベルの学習オートマトンすべてに与えられる。ここで、 $y(x) \leq H$ となるか $y(x) > H$ となるかは、観測値 $y(x)$ がノイズを含んでいるために確率的な事象であることに注意しよう。すなわち、学習オートマトンへの返答はある未知の確率に基づいて生成されると考えられる。この未知確率を、学習オートマトンにおいては reward 確率と呼ぶが、文献9)の場合には、平均観測値 H がその生成法から時刻 t に関して変動するので、reward 確率が時変であると言える。

ところで、本論文において用いる Estimator 型学習アルゴリズムでは、つぎのような reward 確率のサンプル平均値を学習に用いる^(注4)：

ある動作 $\alpha_{i_1^j i_2^j \dots i_s^j}^j$ に対する reward 確率を $d_{i_1^j i_2^j \dots i_s^j}^j$ と書くことにすると、そのサンプル平均値 $\hat{d}_{i_1^j i_2^j \dots i_s^j}^j(t)$ は次式で計算される^{10)~12)}。

$$\hat{d}_{i_1^j i_2^j \dots i_s^j}^j(t) = \frac{R_{i_1^j i_2^j \dots i_s^j}^j(t)}{z_{i_1^j i_2^j \dots i_s^j}^j(t)} \quad (6)$$

ただし、 $z_{i_1^j i_2^j \dots i_s^j}^j(t)$ は動作 $\alpha_{i_1^j i_2^j \dots i_s^j}^j$ が学習開始時から現在時刻 t までに出力された回数であり、また、 $R_{i_1^j i_2^j \dots i_s^j}^j(t)$ は動作 $\alpha_{i_1^j i_2^j \dots i_s^j}^j$ が学習開始時から現在時刻 t までに成功の返答を得た回数である。

(6) 式によって与えられる reward 確率のサンプル平均値は学習開始時から現在時刻 t までのすべての出力と入力(返答)を用いて計算されることから、この値を学習に用いるときには reward 確率が時不変であることが望ましい¹⁰⁾。すなわち、返答生成の判定基準が定数であることが望ましいということになる。よって、本論文では、文献9)の平均観測値を返答生成の判定基準に用いることはせずに、何らかの定数を

(注2) 動作確率 $p_{i_1^j i_2^j \dots i_{s-1}^j}^j(t)$ は、動作 $\alpha_{i_1^j i_2^j \dots i_{s-1}^j}^j$ が学習オートマトンによって選択される確率である。

(注3) 一般的には、探索最小点は大域的最小点とは異なる点になると考えられる。

(注4) この reward 確率のサンプル平均値は reward 確率の推定値となっており^{10), 11)}、このことからこれを用いる学習アルゴリズムは Estimator 型学習アルゴリズムと呼ばれる。

Fig. 2 Threshold of reward

判定基準として導入することにする。このとき、定数の判定基準が1つしか用意されていないとすると、大域的最小点の探索を行うことが困難となる場合が考えられる。このことをFig.2に示すような1次元空間上の目的関数で見てみよう：いま、ある定数判定基準 H_k に着目する。この H_k は、探索小区間 $[x_a, x_c]$ と $[x_c, x_d]$ のいずれに大域的最小点が含まれるかを探索することができる。そして、 $[x_a, x_c]$ に大域的最小点が含まれていそうだと判断されて、 $[x_a, x_c]$ が $[x_a, x_b]$ と $[x_b, x_c]$ に2分割されたとする。すると、Fig.2からも明らかなように、 $[x_a, x_b]$ と $[x_b, x_c]$ の両方で関数値が常に H_k を下回っているために、判定基準 H_k では $[x_a, x_b]$ と $[x_b, x_c]$ のどちらに大域的最小点が含まれるかを判断することは困難となる。そこで、さらに小さい判定基準 H_m を用意すると、これに対しては $[x_a, x_b]$ と $[x_b, x_c]$ の両方で関数値が常に下回ると言うことはなくなって、 $[x_a, x_b]$ と $[x_b, x_c]$ のどちらに大域的最小点が含まれるかを判断することが可能となる。

このような考察から、本論文では、可変階層構造学習オートマトンのレベルの増加につれて単調に小さくなっていくような定数の判定基準列を設けることが必要となる。このような判定基準として、実数 H_s を導入することにする。 H_s はレベル s ($s = 1, 2, \dots, N_{max}$) の学習オートマトンへの返答生成の判定基準であって、つぎのように構成する。

H_1 = 全探索領域からランダムにとった

有限個の点での観測値の平均値 (7)

$$H_s = aH_{s-1} + (1-a)\bar{y}_s, \quad s = 2, 3, \dots, N_{max} \quad (8)$$

ただし、 $\bar{y}_s$ は探索開始時から判定基準 H_s が構成されるまで得られたすべての観測値の中の最小のものである。また、 a は $0 < a < 1$ なる実数であって、 H_{s-1} と $\bar{y}_s$ との間を $1-a : a$ に内分する値として H_s を決定するためのパラメータである。また、 N_{max} は、時刻 t における各可変階層構造学習オートマトンのパスのレベル数の中で最大のもの

をそれぞれ $\{N_1^{m_1}, N_2^{m_2}, \dots, N_n^{m_n}\}$ とするとき、次式で与えられる。

$$N_{max} = \max\{N_1^{m_1}, N_2^{m_2}, \dots, N_n^{m_n}\} \quad (9)$$

すなわち、時刻 t における全パスのレベル数の中で最大のものを表している。そして、学習オートマトンへの返答の生成は以下のように行う：

時刻 t において得られた観測値を $y(x)$ とする。このとき、各可変階層構造学習オートマトンによって選択されたパス上のレベル s の学習オートマトンに対して、 $y(x) \leq H_s$ のときには成功の返答 $\beta_s(t) = 1$ を生成し、また、 $y(x) > H_s$ のときには失敗の返答 $\beta_s(t) = 0$ を生成する。

(7) 式および (8) 式より明かなように、 H_s はレベル1で最大であり、レベル数が増えるほど小さく設定されていく。すなわち、階層構造の上位レベルほど成功の返答を受け易く、下位レベルに行くほど成功の返答を受けにくくなる。このことから、この H_s を用いるということは、階層構造の上位のレベルで大域的最小点が存在しそうな範囲を大まかに認識しておき、レベル数を増やしながらその範囲を徐々に絞り込んでいって、最終的に、最下位レベル（最大レベル）で目的とする最小幅の探索小区間を見出そうとしていることになる。

5. Estimator 型学習アルゴリズムに基づく探索アルゴリズム

Estimator 型学習アルゴリズムをもつ可変階層構造学習オートマトンを用いて、大域的最小点探索アルゴリズムを以下のように構成する。

((探索アルゴリズム))

Step 1 初期時刻を $t = 0$ とし、 n 個の可変階層構造学習オートマトンの初期構造 Σ_j^0 ($m_j = 0$; $j = 1, 2, \dots, n$)、初期動作確率および収束判定値 q_{max} (1に近い小数) を与えるとともに、判定基準 $\{H_s : s = 1, 2, \dots, N_{max}\}$ を計算する。

Step 2 時刻 t の可変階層構造学習オートマトンの出力に従って、観測領域を決定する。

Step 3 一様乱数を用いて観測領域内の1点 x を選択し、観測値 $y(x)$ を得る。

Step 4 観測値 $y(x)$ と判定基準 $\{H_s : s = 1, 2, \dots, N_{max}\}$ とから成功あるいは失敗の返答 $\{\beta_s(t) | s = 1, 2, \dots, N_{max}\}$ を生成し、可変階層構造学習オートマトンに与える。

Step 5 可変階層構造学習オートマトン W_j ($j = 1, 2, \dots, n$) が、Estimator 型学習アルゴリズムに従って動作確率を更新する。

[Estimator 型学習アルゴリズム] 時刻 t において、 W_j のパス $\phi_{i_1^{i_j} i_2^{i_j} \dots i_{N_{l_j}}^{i_j}}^j$ が選択されたとする。

(i) 選択されたパス上の各学習オートマトン $Z_{i_1^{i_j} i_2^{i_j} \dots i_{s-1}^{i_j}}^j$ ($s = 1, 2, \dots, N_{l_j}^{m_j}$) の動作確率分布を、以下のように更新する。

$$p_{i_1^{i_j} i_2^{i_j} \dots i_{s-1}^{i_j} k_s^j}^j(t+1) = p_{i_1^{i_j} i_2^{i_j} \dots i_{s-1}^{i_j} k_s^j}^j(t)$$

$$\begin{aligned}
& -\lambda_s \left[f(\hat{e}_{i_1^j i_2^j \dots i_s^j}^j(t)) - f(\hat{e}_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^j(t)) \right] \\
& \times \left[u_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^{i_1^j i_2^j \dots i_s^j} p_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^j(t) \right. \\
& \left. + u_{i_1^j i_2^j \dots i_s^j}^{i_1^j i_2^j \dots i_{s-1}^j k_s^j} \left(1 - p_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^j(t) \right) \right. \\
& \left. \times \frac{p_{i_1^j i_2^j \dots i_s^j}^j(t)}{r-1} \right] \\
& (k_s^j = 1, 2, \dots, r; k_s^j \neq i_s^j) \quad (10)
\end{aligned}$$

$$p_{i_1^j i_2^j \dots i_s^j}^j(t+1) = 1 - \sum_{\substack{k_s^j=1 \\ k_s^j \neq i_s^j}}^r p_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^j(t+1) \quad (11)$$

ただし, λ_s は $0 < \lambda_s < 1$ を満たす実数であり, f は $[0,1] \rightarrow [0,1]$ なる単調増加関数である. また,

$$\begin{aligned}
& u_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^{i_1^j i_2^j \dots i_s^j}(t) \\
& = \begin{cases} 1 & \hat{e}_{i_1^j i_2^j \dots i_s^j}^j(t) > \hat{e}_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^j(t) \\ 0 & \hat{e}_{i_1^j i_2^j \dots i_s^j}^j(t) \leq \hat{e}_{i_1^j i_2^j \dots i_{s-1}^j k_s^j}^j(t) \end{cases} \quad (12)
\end{aligned}$$

である.

(ii) 次式に従ってパラメータ $\hat{e}$ を計算する. ただし, $\hat{d}_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j(t+1)$ の値は (6) 式に従って求める.

(a) $\alpha_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j$ が最下レベルの動作である場合:

$$\hat{e}_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j(t+1) = \hat{d}_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j(t+1) \quad (13)$$

(b) $\alpha_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j$ が最下レベルの動作でない場合:

$$\begin{aligned}
& \hat{e}_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j(t+1) = \hat{d}_{i_1^j i_2^j \dots i_{s-1}^j l_s^j}^j(t+1) \\
& \times \max_{l_{s+1}^j} \{ \hat{e}_{i_1^j i_2^j \dots i_{s-1}^j l_{s+1}^j}^j(t+1) \} \quad (14)
\end{aligned}$$

なお, レベル s において, $Z_{i_1^j i_2^j \dots i_{s-1}^j}^j$ 以外の学習オートマトンの動作確率分布およびパラメータ $\hat{e}$ の値は全く変更されない(注5).

Step 6 最大パス(注6) $\tilde{\phi}_{i_1^j i_2^j \dots i_{N_{l_j}^{m_j}}}^j$ 上の動作確率が

$p_{i_1^j i_2^j \dots i_s^j}^j \geq q_{\max}$ for $\forall s$ を満たす可変階層構造学習オートマトンが存在するとき, この可変階層構造学習オートマトンがパス $\tilde{\phi}_{i_1^j i_2^j \dots i_{N_{l_j}^{m_j}}}^j$ に収束したとして, Step 7 へ. そうでなければ $t = t+1$ として Step 2 へ.

(注5) Step 5 において行われる計算は, 1つの W_j に関して, 動作確率の計算が $N_{l_j}^{m_j} \times r$ 個, $\hat{d}$ の計算が $N_{l_j}^{m_j}$ 個および $\hat{e}$ の計算が $N_{l_j}^{m_j}$ 個であるので, 計算の総数は $N_{l_j}^{m_j} \times (r+2) \times n$ となる.

(注6) パス上の動作の動作確率の積をそのパスのパス確率と呼び, また, 階層構造上で最大のパス確率を持つパスを最大パスという.

Step 7 収束したパスのレベル数が最大レベル数であるとき Step 8 へ. そうでなければ, 収束したパスに対応する探索小区間を分割し, この可変階層構造学習オートマトンの構造を更新する. そして, $N_{\max}$ を計算し, その値が増加した場合には判定基準 $H_{N_{\max}}$ を求める. $t = t+1$ として Step 2 へ.

Step 8 すべての可変階層構造学習オートマトンが最大レベル数のパスに収束しているとき, それらのパスが示している各探索小区間の中点からなる点を探索最小点として本アルゴリズムを終了する. そうでなければ $t = t+1$ として Step 2 へ.

6. 数値シミュレーション

本探索アルゴリズムの有用性を数値シミュレーションによって検討する. 目的関数としてつぎの2次元関数を用いる (Fig. 3 参照).

$$\begin{aligned}
f(x^1, x^2) &= x^1 + x^2 \\
&+ 10 \sin\left(\frac{\pi}{20} x^1 x^2\right) + 10 \sin\left(\frac{\pi}{2} x^2\right) \quad (15)
\end{aligned}$$

ただし, $-10 \leq x^1 \leq 10$, $-10 \leq x^2 \leq 10$ である. このとき, この関数は大域的最小点 $(x^1, x^2) = (-7.82, -9.00)$ で大域的最小値 -36.80 をもつ. また, シミュレーションの条件をつぎのように設定する. すなわち, 観測値に加わる観測ノイズは $[-3, 3]$ の一様乱数とする. そして, 動作数2の学習オートマトンから成る可変階層構造学習オートマトンを用いることとし, 各変数の探索小区間の最小幅を0.1以下とすることにする. すると, $20 \div 2^8 = 20 \div 256 \cong 0.078 < 0.1$ より, 可変階層構造学習オートマトンの最大レベル数は8となる. そして, 可変階層構造学習オートマトンの初期構造 Σ^0 として1個の学習オートマトンからなる構造を採用することにする. このとき, 本探索アルゴリズムによって50回の探索実験を行い, その探索結果をTable 1に示す. このとき, SMPは探索最小点 (Searched Minimum Point) を表してお

Fig. 3 Objective function for simulation

Table 1 Simulation results

	Average Gap between SMP and GMP	Iteration Number
Our Algorithm	0.142	2618.3
L_{R-I} Type Algorithm	0.076	7504.2

Table 2 Values of a and results of search

Value of a	Average Gap between SMP and GMP	Iteration Number
0.40	0.059	27575.7
0.45	0.075	14255.9
0.50	0.110	3490.1
0.55	0.142	2618.3
0.60	0.148	6688.6

り、また、GMPは大域的最小点(Global Minimum Point)を表している。そして、比較のために、文献9)の探索アルゴリズム(L_{R-I} 型学習アルゴリズムに基づく)を適用した結果をあわせて示す。Table 1において、本探索アルゴリズムによる探索最小点(SMP)と大域的最小点(GMP)とのずれ(gap)の平均は0.142となっている。 x^1 あるいは x^2 方向の探索区間幅20に対するこの値の割合を求めてみると $0.142/20 \times 100 = 0.71\%$ であるので、本探索アルゴリズムは十分実用的な精度の探索最小点を見出しているといえる。ところで、本探索アルゴリズムの探索最小点と大域的最小点とのずれ(gap)の平均と文献9)の探索アルゴリズムのそれとを比較すると、本探索アルゴリズムの方が0.066大きくなっている。しかし、この差の x^1 あるいは x^2 方向の探索区間幅20に対する割合は $0.066/20 \times 100 = 0.33\%$ であるので、探索精度におけるこの差はほとんど無視できると考えられる。この一方で、探索回数に着目すると、本探索アルゴリズムの場合には文献9)の探索アルゴリズムの34.9%の探索回数となっており、このことから本探索アルゴリズムによって探索回数が大幅に削減されていることが分かる。

つぎに、返答の判定基準(8)式に含まれるパラメータ a の値が、本探索アルゴリズムの性能にどのような影響を与えるかを見てみることにする。これは、返答の生成の仕方が学習オートマトンの性能に影響を与えと考えられるからである。パラメータ a の値を種々変更して探索(それぞれ50回づつ)を実行した場合の結果をTable 2に示す。この表から明らかなように、 a の値が小さくなるにつれて探索精度は高くなり、 a の値が大きくなるにつれて探索精度は下がることになる。ここで、(8)式に注意すると、 a の値が小さい場合には、各レベルでの判定基準 H_s が $\bar{y}_s$ に近い値となり、その結果、下位のレベルにおいては、成功の返答を得る探索小区間が急速に絞り込まれていくことになる。また逆に、 a の値が大きい場合には、各レベルでの判定基準 H_s が $\bar{y}_s$ から遠い値となるので、成功の返答を得る探索小区間の絞り込みが不十分になると思われる。このことから、 a の値が小さいほど探索精度が高くなると考えられる。これに対して、探索回数の方

Table 3 Noise magnitudes and results of search

Noise Magnitude	Average Gap between SMP and GMP	Iteration Number
$[-3,3]$	0.142	2618.3
$[-10,10]$	0.158	5715.3
$[-20,20]$	0.161	9085.6
$[-30,30]$	0.188	10987.0

を見てみると、 $a = 0.55$ (Table 1に採用)の時に最小となり、この値から遠ざかるほどそれが増加することが分かる。すなわち、探索回数に着目する限りは、最適な a の値が存在することが示されているわけであるが、現在のところ、その正確な値を決定するには至っていない。

最後に、観測ノイズの大きさと本探索アルゴリズムの性能との関係について見てみよう。観測ノイズの大きさが $[-3,3]$ 、 $[-10,10]$ 、 $[-20,20]$ および $[-30,30]$ の場合について、本探索アルゴリズムを適用(それぞれ50回)した結果をTable 3に示す($a = 0.55$)。この表から明かなように、観測ノイズの大きさが大きくなっていくと、探索最小点と大域的最小点とのずれが徐々に拡大し、同時に探索回数が増加するが、特に探索回数の増加が著しい。このことから、観測ノイズの大きさは探索回数に大きな影響を与えるということが分かる。これは観測ノイズが大きくなると、学習オートマトンに与えられる返答の不確かさがより増大することによって考えられる。それでも、ある程度の探索回数をかければ大域的最小点に近い点を見出しており、これは、不確実な情報のもとでも、それに基づいて繰り返し学習を行うことによって徐々に最適解を見出していくことが出来るという学習オートマトンの特性によるものと思われる。

7. む す び

本論文においては、観測ノイズが存在する場合の多峰性未知目的関数の最適化問題について考察し、Estimator型学習アルゴリズムをもつ可変階層構造学習オートマトンを用いることによって、多峰性未知目的関数の大域的最小点を見出すための探索アルゴリズムを構築した。そして、観測ノイズが存在する場合の2次元多峰性未知目的関数の大域的最小点探索に本探索アルゴリズムを適用した数値シミュレーションを行い、本探索アルゴリズムが、文献9)の探索アルゴリズムと同程度の精度の探索最小点を、約1/3の探索回数で見出すことを示した。すなわち、本探索アルゴリズムによって探索回数の大幅な削減が実現されていることが分かった。これは、Estimator型学習アルゴリズムが、文献9)で用いられた L_{R-I} 型学習アルゴリズムよりも少ない学習回数で最適解を見出すことができることに依っている。

また、観測ノイズの大きさと本探索アルゴリズムの性能との関係においては、観測ノイズの大きさが探索回数に大きな影響を与えることが分かった。すなわち、観測ノイズの増大に伴って探索回数が大幅に増加していくことになる。しかしそのときでも、探索精度はやや下がるものの、大域的最小点

に近い点を見出しており、本探索アルゴリズムが観測ノイズが存在する場合の大域的最小点探索に有効であることが示された。

ところで、数値シミュレーションで明らかになったように、返答の判定基準を与えるパラメータ α に最適値が存在するわけであるが、この決定法を確立することが今後の課題である。さらに、本論文においては、探索アルゴリズムの有用性を数値シミュレーションによって確かめているが、その収束特性を理論的に明らかにすることは当然必要である。これに関しては現在考察中であり、つぎの機会に報告する予定である。

参考文献

- 1) 馬場, 坂和: 数理計画法入門, 共立出版 (1989)
- 2) 茨木, 福島: 最適化の手法, 共立出版 (1993)
- 3) J. Matyas: Random Optimization, Automation and Remote Control, **26-2**, 244/251 (1965)
- 4) F.J. Solis and R.J-B. Wets: Minimization by Random Search Techniques, Mathematics of Operations Research, **6-1**, 19/30 (1981)
- 5) N. Baba and T. Shoman: A Modified Convergence Theorem for a Random Optimization Method, Information Sciences, **13**, 159/166 (1977)
- 6) N. Baba: Convergence of a Random Optimization Method for Constrained Optimization Problems, Journal of Optimization Theory and Applications, **33-4**, 451/461 (1981)
- 7) N. Baba: A Hybrid Algorithm for Finding a Global Minimum, Int. J. Control, **37-5**, 929/942 (1983)
- 8) ネベルソン, ハスミンスキー共著 北川, 田嶋共訳: 確率近似法 確率近似と繰返し推定, 共立出版 (1983)
- 9) 最上, 馬場, 古澤: 観測ノイズを考慮した多峰性未知関数最適化問題への可変階層構造学習オートマトンの適用, 計測自動制御学会論文集, **34-8**, 1097/1103 (1998)
- 10) M.A.L. Thathachar and P.S. Sastry: A Class of Rapidly Converging Algorithms for Learning Automata, IEEE International Conference on Cybernetics and Society, 602/606 (1984)
- 11) M.A.L. Thathachar and P.S. Sastry: A Hierarchical System of Learning Automata That Can Learn the Globally Optimal Path, Information Sciences, **42**, 143/166 (1987)
- 12) 最上, 馬場, 田上: 各レベルに定常一人教師環境をもつ可変階層構造学習オートマトン, 計測自動制御学会論文集, **28-5**, 626/631 (1992)

[著者紹介]

最上 義夫 (正会員)

1975年京都工芸繊維大学工学研究科修士課程修了。同年徳島大学工学部情報工学科助手, 97年知能情報工学科講師, 現在に至る。学習オートマトン, 強化学習, 確率分布定数システムの観測器・制御器の最適配置問題の研究に従事。京都大学博士(工学)。計測自動制御学会, システム制御情報学会, 電子情報通信学会会員。

馬場 則夫 (正会員)

1974年京都大学大学院工学研究科博士課程数理工学専攻修了。同年徳島大学工学部情報工学科助手, 講師, 助教授を経て, 88年同大学知能情報工学科教授, 90年大阪教育大学教育学部教養学科情報科学専攻教授, 現在に至る。ニューラルネット, ゲーミング, 強化学習, GAs, 数理計画法, 学習オートマトン等の研究に従事。工学博士。IEEE, INNS, 計測自動制御学会, システム制御情報学会, 日本シミュレーションゲーミング学会, 日本機械学会, オペレーションズ・リサーチ学会等の会員。

松下 正樹

1996年徳島大学工学研究科博士前期課程知能情報工学専攻修了。可変階層構造学習オートマトンの研究に従事。同年三菱電機(株)入社, 現在に至る。