カオスの縁を考慮したカオスシステムのニューラルネットワーク制御†

大 林 正 直*・梅 迫 公 輔**・中 山 大 輔**

Chaotic System Control Considering Edge of Chaos Using Neural Network†

Masanao OBAYASHI*, Kousuke UMESAKO** and Daisuke NAKAYAMA**

In this paper, an efficient robust control method for chaotic system introducing the concept, the edge of chaos (:boundary status between chaos and non-chaos), is proposed. To realize this concept, we introduce an extended performance index which consists of two parts. One is for achievement of the system's objects, another is for keeping the system edge of chaos. Parameters of the neural network controller are adjusted to minimize the value of the extended performance index and achieve the above two objects using Random Search with Intensification and Diversification (RAS-ID). Through Lorenz chaotic system control simulation, it is clarified that our proposed method is superior to the conventional neuro control method in the view point of the generalization ability.

Key Words: chaotic system, edge of chaos, neural network controller, RAS-ID

1. はじめに

現在、制御工学において制御理論を適用したカオスシステムの制御は重要な研究領域として注目されており、学習によるニューラルネットワーク制御方式を用いた OGY 法などを含め、さまざまな手法が提案されている^{1)~3)}。本論文では、カオスシステムのニューラルネットワーク制御法における一つの提案をし、その制御性能について従来法と比較検討を行なう。

一般に情報処理システムにおいては、情報を保存する安定性と情報を伝達する流動性の両方の調和が必要であるとされている。システムが非カオス状態であれば静的すぎ情報は途中で凍結してしまう。また逆に、カオス状態であれば、動的すぎ情報は途中で壊れてしまう。静的すぎず動的すぎないカオスの縁でのみ情報が適度に保持される安定性と適度に伝達される流動性とが絶妙なバランスを保てる⁴⁾。また生命においても、静的すぎても、動的すぎても生命は存在し得ない。このちょうど境界の部分でバランスをとっていると言われる⁵⁾。このようにカオスの縁は重要な研究領域であると言える。

そこで本論文ではこのカオスの縁状態を考慮しながら制御を行なう新しい方式を提案する。通常、制御器はある評価指

標を最小化するように設計する。これまではこの評価指標としてシステムの目標出力値と出力値との 2 乗誤差平均、もしくはこれにエネルギー項の拘束条件を付加したものが用いられている。本論文では、この通常的评价指標に、カオスであればそのリアプノフスペクトルにおいて、その最大が正を示しかつその最小が負を示す、リアプノフ指数を用いた評価項を新たに追加する。具体的には、コントローラ的设计時にカオスシステムにおけるリアプノフ指数の目標値と実際のリアプノフ指数値の 2 乗誤差を新たな評価項として付加する。

ところで、OGY 法は、軌道が目標軌道の近傍に遷移したときに制御入力を印加して目標軌道に収束させる。しかしながら、この制御入力を求める、すなわち、OGY 法を適用するためには、目標軌道の近傍での線形化システムを求める必要があるため、制御系設計時に指定された目標軌道が変更されるたびに設計変更が必要となる。この点、本論文の主題である制御系設計時に指定されたシステム出力の目標値が、その後、システムの稼働中にその目標値の変更を余儀なくされた場合、当初設計された制御系でどこまで対応できるかという点が比較できない。したがって、比較する従来法として、本主題において比較可能な、これまで提案されているニューラルネットワークによるカオス制御法(以後、従来法と呼ぶ)を採用する。

シミュレーションでは、提案法におけるリアプノフ指数に関する評価項を、システムのリアプノフスペクトルにおけるリアプノフ指数の最大値が正、零、負となるように各リアプノフ指数の目標値を設定した 3 種類のコントローラとリアプノフ指数に関する評価指標を付加しない(システムの出力値とその目標値との誤差のみを考慮する従来法)の場合の 4 種類のコントローラをそれぞれ構成し、それらの制御性能の違

† 第 10 回 SICE 中国支部学術講演会で発表 (2001・12)

* 山口大学工学部 宇部市常盤台 2-16-1

** 山口大学大学院理工学研究科 宇部市常盤台 2-16-1

* Faculty of Eng., Yamaguchi University, Ube-shi

** Graduate School of Sci. and Eng., Yamaguchi University, Ube-shi

(Received April 8, 2002)

(Revised August 26, 2002)

Fig. 1 Control system

Fig. 2 A structure of neural network controller

いを検討する。さらに、汎化性に注目し、特定の目標出力値を出力するように設計されたニューラルネットワークのコントローラが、変更された目標出力値に対してどれほどの追従能力があるかという性能についての検討を行なう⁶⁾。

2. 制御システムの構成

制御システムの構成を Fig. 1 に示す。Chaos system は制御対象システムとしての Chaos system を表わしている。この Chaos system の出力フィードバック信号と目標値との誤差が Neural network controller への入力となる。そして、Neural network controller からの出力が制御入力として Chaos system へ入力される。

またニューラルネットワークコントローラとして Fig. 2 のような 1 入力 1 出力で中間層が 1 つの階層型ニューラルネットワークを用いる。また中間層ニューロンには $[-1.0, 1.0]$ の範囲をとる以下のようなシグモイド関数を用いる。

$$\left. \begin{aligned} h_j &= \frac{1.0 - \exp(-\phi_j \alpha_j)}{1.0 + \exp(-\phi_j \alpha_j)} \\ \alpha_j &= \omega_{jI} h_I - \theta_j \quad (j = 1, \dots, N) \\ h_O &= \sum_{j \in N} \omega_{jO} h_j - \theta \end{aligned} \right\} \quad (1)$$

h_I : 入力, N : 中間層ニューロンの総数, h_j : 中間層の j 番目のニューロン出力, θ_j : 中間層の j 番目ニューロンのしきい値, ϕ_j : 中間層の j 番目ニューロンの傾き, ω_{jI} : 入力から中間層の j 番目ニューロンへの結合重み, ω_j : 中間層の j 番目ニューロンから出力への結合重み, θ : 出力層ニューロンのしきい値, h_O : コントローラ出力

3. リアプノフ指数

カオスか非カオスを判別するには、システムの最大リアプノフ指数が正、かつ最小リアプノフ指数が負であるか否かを判定する必要があるが、本論文では、対象システムがカオス

を含むことが既知であるとして、最大リアプノフ指数が正であれば多くの場合カオスであると判定できるとし、最大リアプノフ指数の値のみによって判別する方法をとる^{7)~9)}。

カオスでは、わずかに離れた二つの軌道が時間とともに指数分離していく。リアプノフ指数はその離れた二つの軌道が分離する、ないし近付く割合を測るものである。具体的には軌道の差の対数における長時間平均をとる。二つの軌道をわずかにずらす方向は相対空間の自由度の数だけあるので上述した分離度もその数だけ定義できる。つまり一般に自由度の数だけのリアプノフ指数が存在する。このリアプノフ指数の少なくとも一つが正であるならば状態はカオスであるといえることができる。これは、初期値を少し変えたと後の状態が大きく変わるという、鋭敏な初期値依存性を示す^{5), 7)~10)}。このようにして求められたリアプノフ指数の内最大のものを最大リアプノフ指数と言う。最大リアプノフ指数が正であればシステムはカオス状態であり、負であれば安定状態を表わす。また、カオス状態と非カオス状態の境界に位置するカオスの縁と呼ばれる状態をとるとき、リアプノフ指数は 0 付近の値を示す³⁾。以下に三次元のリアプノフ指数を例にあげ、最大リアプノフ指数の計算法を示す。

まず新たな三次元ベクトルを定義する。

$$s(t) = [x(t), y(t), z(t)]^t \quad (2)$$

t : 転置

x, y, z : 状態変数

ここでつぎのような非線形写像を与える。

$$s(t+1) = F(s(t)) \quad (3)$$

$s(t+1)$ の微小変化 $\delta s(t+1)$ を用いて線形近似する。

$$\delta s(t+1) = DFF(s(t))\delta s(t) \quad (4)$$

ここで DF はヤコビ行列といつぎのように計算する。

$$DF = \begin{pmatrix} \frac{\partial x(t+1)}{\partial x(t)} & \frac{\partial x(t+1)}{\partial y(t)} & \frac{\partial x(t+1)}{\partial z(t)} \\ \frac{\partial y(t+1)}{\partial x(t)} & \frac{\partial y(t+1)}{\partial y(t)} & \frac{\partial y(t+1)}{\partial z(t)} \\ \frac{\partial z(t+1)}{\partial x(t)} & \frac{\partial z(t+1)}{\partial y(t)} & \frac{\partial z(t+1)}{\partial z(t)} \end{pmatrix} \quad (5)$$

リアプノフスペクトルは微小変化を三次元相空間の互いに直交する単位ベクトルの組 $u_i(t_0)$ で与えられたときの各ベクトルの伸びを時間平均することにより得られる。単位ベクトル $u_i(t_0)$ は以下のとおりとする。

$$u_i(t_0) = \begin{cases} u_1(t_0) = (1, 0, 0)^t \\ u_2(t_0) = (0, 1, 0)^t \\ u_3(t_0) = (0, 0, 1)^t \end{cases} \quad (6)$$

ここで以下の式を繰り返し計算することによりベクトルの時系列 $e_i(t+1)$ が得られる。

$$e_i(t+1) = DF \cdot u_i(t) \quad (7)$$

$u_i(t)$ は、 $e_i(t)$ を求めた後にグラムシュミット直交化法により正規化したものである。グラムシュミット直交化法は以下

により与えられる.

$$e_i(t) = \begin{cases} e_i(t) & (i=1) \\ e_i(t) - \sum_{j=1}^{i-1} \langle e_i(t), u_j(t) \rangle u_j(t) & (i \geq 2) \end{cases} \quad (8)$$

$$u_i(t) = \frac{e_i(t)}{\|e_i(t)\|} \quad (9)$$

ここで $\langle \cdot, \cdot \rangle$ は内積, $\|\cdot\|$ はユークリッドノルムを表わす. 以上の繰り返し計算により, 以下のリアプノフスペクトルが得られる.

$$\lambda_i = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{t=t_0+1}^{t+N} \log \|e_i(t)\| \quad (i=1, 2, 3) \quad (10)$$

この内, 最大のものが最大リアプノフ指数である.

4. 適応的ランダム最適化手法 (RasID) ¹²⁾

RasID は, パラメータ探索の成功, 不成功情報を活用し, 探索の集中化と多様化を統一化した枠組で行なう最適化手法である. 探索は, 過去の探索の成功, 不成功情報により決定され探索成功であれば改善解が近傍に存在する可能性が高いと考え, 探索範囲を限定し, 探索が成功した方向を重点的に探索する. 一方, 不成功であれば近傍に改善解の存在する可能性が低いと考え, 探索を広範囲でランダムに行なう. 以上により評価指標の改善とローカルミニマムからの脱出が可能となる. RasID におけるパラメータ探索の基本手順を以下に示す.

$$\text{if } E(\omega + x) < E(\omega) \text{ であれば探索成功} \quad (11)$$

$$\text{then } \omega \leftarrow \omega + x$$

$$\text{if } E(\omega + x) \geq E(\omega) \text{ であれば探索不成功} \quad (12)$$

$$\text{then } \omega \leftarrow \omega$$

ここで,

E : 評価指標

$\omega = [\omega_1, \dots, \omega_m, \dots, \omega_M]$ (パラメータベクトル)

$x = [x_1, \dots, x_m, \dots, x_M]$ (探索パラメータベクトル)

x_m を探索する確率密度関数 $f(x_m)$ および, 確率分布関数 $F(x_m)$ をつぎのように与える.

$$f(x_m) = \begin{cases} p_m \beta e^{\beta x_m} & (x_m \leq 0) \\ q_m \beta e^{-\beta x_m} & (x_m > 0) \end{cases} \quad (13)$$

$$F(x_m) = \begin{cases} p_m e^{\beta x_m} & (x_m \leq 0) \\ 1 - q_m e^{-\beta x_m} & (x_m > 0) \end{cases} \quad (14)$$

$$p_m + q_m = 1$$

探索パラメーター x_m はつぎのように計算できる (Fig. 3 参照).

$$0 \leq z \leq p_m \Rightarrow x_m = \frac{1}{\beta} \ln\left(\frac{z}{p_m}\right) \quad (15)$$

$$p_m \leq z \leq 1.0 \Rightarrow x_m = -\frac{1}{\beta} \ln\left(\frac{1-z}{q_m}\right) \quad (16)$$

なお, パラメーターの探索範囲を決定する (16) 式と (17) 式の β は, 以下の式で計算することが出来る.

$$\beta = \bar{\beta} e^{-\eta n} + \underline{\beta} \quad (17)$$

$$\bar{\beta} + \underline{\beta} : \beta \text{ の上限, } \underline{\beta} : \beta \text{ の下限}$$

η : 定数

また, (13) 式と (14) 式における p_m, q_m と (17) 式の n を過去の探索成功, 不成功情報によりつぎのように更新する. なお, p_m を更新した時は $q_m = 1 - p_m$ とし q_m を更新した時は $p_m = 1 - q_m$ とする.

○ 探索成功の場合

$$\text{負方向探索のとき, } p_m \leftarrow \alpha p_m + (1 - \alpha) \quad (18)$$

$$\text{正方向探索のとき, } q_m \leftarrow \alpha q_m + (1 - \alpha) \quad (19)$$

$$n \leftarrow n - 1 (n \leq n_0 \text{ のとき}) \quad (20)$$

ただし, $n = 0$ を下限とする.

$$n \leftarrow n_0 (n > n_0 \text{ のとき}) \quad (21)$$

○ 探索不成功の場合

$$\text{負方向探索のとき, } p_m \leftarrow \alpha p_m \quad (22)$$

$$\text{正方向探索のとき, } q_m \leftarrow \alpha q_m \quad (23)$$

$$n \leftarrow n + 1 \quad (24)$$

α : 探索確率の指数平滑定数 ($0 < \alpha < 1$)

n_0 : 集中化探索と多様化探索の分岐点

(n の初期値)

Fig. 4 に n と β の関係を示しているが, 探索が成功している間は, n の値が小さくなるので β が大きくなり, したがって, 探索範囲が小さくなる. また (18), (19) 式のいずれかで過去に探索が成功してきた方向の確率が大きくなるように p_m, q_m を更新していく. この結果, 評価指標の確実な改善が期待される (Fig. 5 での A 参照).

探索が進むにつれて, 評価指標の確実な改善が見られなくなると n の値は, 徐々に大きくなっていき β が小さくなる. また, (22), (23) 式のいずれかで p_m, q_m が更新されていくが, 特定方向の探索確率が大きくなることはないで, 結果的に正負両方向を広い範囲で探索することになる (Fig. 5 での B 参照). 広範囲の探索で探索が成功し, 評価指標の改善が実現すると, (21) 式より, n が指定値 n_0 に変更されるので, 再び狭い範囲での探索になる (Fig. 5 での A 参照).

以上のアルゴリズムで探索の集中化と多様化が繰り返され効率的な最適値探索が可能となる.

5. 評価指標

従来法における評価指標は, (25) 式に示すようなカオスシステムの目標出力とシステムの出力値との 2 乗誤差平均とする通常の指標とする.

$$E_{state} = \frac{1}{2T} \sum_t (y_d - y(t))^2 \quad (25)$$

y_d : システムの目標出力値

Fig. 3 Probability density and distribution function of searching x_m

Fig. 4 Relation between function slope β and n

Fig. 5 Intensified and diversified searching

$y(t)$: t 時刻におけるシステムの出力値

T : システムの時間ステップ

ここで, (26) 式に示すようなリアプノフ指数に関する評価指標を導入する.

$$E_{Lyapunov} = \frac{1}{2} \sum_{i=1}^n (\lambda_{d_i} - \lambda_i)^2 \quad (26)$$

λ_{d_i} : システムの i 番目リアプノフ指数の目標値

λ_i : システムの i 番目リアプノフ指数値

n : システムの次数

本稿で提案する評価指標は (25) 式に (26) 式を付加した (27) 式である. またリアプノフ指数の評価指標に対する重みとして係数 X を付加する.

Fig. 6 Lorentz chaos

$$E = \frac{1}{2} \left\{ \frac{1}{T} \sum_t (y_d - y(t))^2 + X \sum_{i=1}^n (\lambda_{d_i} - \lambda_i)^2 \right\} \quad (27)$$

X : リアプノフ指数の評価指標における重み係数

6, 7 節の計算機シミュレーションにおける, 評価 E を最小化するニューラルネットワークコントローラの学習では, コントローラを構成する (1) 式のパラメータ $\phi_j, \omega_{jI}, \omega_j, \theta_j, \theta$ すべてを学習する.

6. 計算機シミュレーション 1

本稿の数値実験には制御対象システムとしてローレンツカオスを用いる. ローレンツカオスはいくつかの大胆な近似の下に対流現象から導かれた以下の 3 変数の常微分方程式で表わされる.

$$\frac{dx}{dt} = \sigma(y - x) \quad (28)$$

$$\frac{dy}{dt} = -xz + \gamma x - y \quad (29)$$

$$\frac{dz}{dt} = xy - \xi z + u \quad (30)$$

x, y, z はそれぞれの状態値, u は制御入力を表わしている. σ, γ, ξ はシステム定数でシミュレーションでは $\sigma = 10, \gamma = 28, \xi = 8/3$ を用いた. 実際の数値実験では以下のように離散化した差分方程式を用いてローレンツ方程式の時系列データを求める.

$$x(t+1) = x(t) + \Delta t \sigma(y(t) - x(t)) \quad (31)$$

$$y(t+1) = y(t) - \Delta t(x(t)z(t) - \gamma x(t) + y(t)) \quad (32)$$

$$z(t+1) = z(t) + \Delta t \sigma(x(t)y(t) - \xi z(t) + u(t)) \quad (33)$$

$x(t), y(t), z(t)$ は時刻 t におけるそれぞれの状態値, Δt は時間刻み幅, $u(t)$ は時刻 t における制御入力を表わしており, 制御入力が 0 のとき, システムはカオス状態となる (Fig. 6 参照). また, Fig. 7 にシステムの出力 $y(t)$ の時刻 t に対する振舞を示す.

6.1 実験方法

本数値実験例では, (26) 式における次数 n は 3. リアプノフ指数は 3 個存在し, 正のリアプノフ指数は 1 個である. そこで, 簡単化のため, リアプノフ指数に関する評価は, この正のリアプノフ指数についてのみ目標値を与え, ほかは評価項に含めない. すなわち, $\lambda_d = \lambda_{d_1}$ とする. また, システムの目標出力値 y_d を 1.0 とし, 評価指標を最小化するように

Fig. 7 Output of Lorentz chaos $y(t)-t$ Fig. 8 Comparison of values of criterion function E between conventional method and proposed Fig. 9 Output $y(t)-t$ for (a) Conventional method, $\lambda_d = -0.03$, $\lambda_d = 0$ (b) $\lambda_d = +0.03$

コントローラの学習を行なう。ここで、表記の簡単化のため(27)式におけるリアプノフ指数に関する評価指標を考慮しない場合、つまり重み係数 $X = 0$ の場合を従来法とし、考慮する場合、つまり重み係数 $X \neq 0$ の場合を提案法とする。さらに、リアプノフ指数の目標値を $\lambda_d = \{-0.03, 0, +0.03\}$ の3ケースを考慮し、従来法と合わせた計4種類のコントローラを学習により設計し、それぞれの制御性能の違いを検討する。システムの時間ステップは $T = 1000$ とし、 t の最

Fig. 10 Robust performance for (a) $\lambda_d = -0.03$, (b) $\lambda_d = 0$

終時刻 $t = 1000$ を t_f とする。

6.2 制御性能

Fig. 8 は学習ステップごとの評価指標の変化を示したものである。評価指標の収束は従来法が最も速い。また提案法においてはリアプノフ指数の目標値を -0.03 にした場合が最も収束が速い。これはシステムの状態とリアプノフ指数の値がともに安定状態に向かうということと一致しているため学習がスムーズに行なわれたためと考えられる。それに対しリアプノフ指数の目標値を $+0.03$ にした場合、最も収束が遅くかつ評価値(2乗誤差)が大きい。これはシステムの状態は安定状態に向かわせ、リアプノフ指数の値はカオス状態に向かわせるという矛盾した学習を行なわせているためであると考えられる。Fig. 9 は学習終了時におけるシステムの出力を表わしたものであるが、リアプノフ指数の目標値が正の場合のみシステム状態が目標値 y_d に収束せず振動しているのがわかる。また学習終了時におけるシステムの最終出力は従来法が最も目標値に近付いている。すなわち、学習終了時の制御性能は従来法が優れているといえる。

6.3 汎化性能

汎化性能を調べるため、それぞれの学習によって得られたコントローラを用いて、目標値 y_d を変化させた場合におけるシステムの出力値 y と、変化させた目標値 y_d とを比較し、汎化能力の検討を行なう。また、(27)式における係数 X が $X = 1/8, 1, 2, 4$ の場合について、リアプノフ指数重みの違いの影響も検討した (Fig. 10)。

Fig. 10 は目標値 y_d に対する実際のシステムの最終時刻出力値 $y(t_f)$ と目標出力値を示したものであり、Fig. 11 は Fig. 10(b) の左端の一部を拡大したものである。Fig. 11 より

Fig. 11 Extended robust performance for $\lambda_d = 0$ Fig. 12 Comparison of the robust performance among the number of middle layer neuron: (a) Conventional method, (b) $\lambda_d = 0$

重み X を大きくするにつれて誤差が大きくなっているのがわかる。また Fig. 10 より目標値 y_d を大きくするにつれてリアプノフ指数の評価指標を付加した提案法のいずれもが目標値の変化に追従し、従来法と比較して優れた汎化能力を示しているのがわかる。提案法同士の比較においてはリアプノフ指数の目標値を $\lambda_d = 0$ とした場合、すなわち、カオスの縁を考慮した場合が汎化性能は最も良い。また X の値が $X = 0$ (従来法) に近付くにつれて汎化能力は悪くなっている。しかし逆に X の値を大きくしすぎても若干性能は悪くなる。これは、リアプノフ指数の影響度が強すぎて目標値追従能力が弱まったためと考えられる。

また、ニューラルネットワークコントローラの問題がある。そこで、ニューラルネットワークコントローラにおいて中間層ノード数はどのくらいが適切かを調べるため、中間層ノード数を変化させて汎化性能の違いを調べた (Fig. 12)。Fig. 12を見ると、どの場合においても中間層ノード数が5個

Fig. 13 Henon chaos

Fig. 14 Output of Henon chaos $y(t)-t$

では少なすぎるといえる。また、どの場合においても15個と25個の場合においては若干25個の方が優れている。25個以上の場合においても実験を試みたが微小な改善は見られるものの、ほとんど違いを確認出来なかった。このことから、計算時間や精度などを考慮すると本ケースの場合、このシステムにおいて中間層ノード数は25個が最も適切であると言える。($\lambda_d = 0.03$ の場合は評価が悪く制御できていないので比較の対象外とした。)

7. 計算機シミュレーション 2

提案法の有効性確認のため、数値実験のもう一つの例として、以下で表わされる2次元離散時間カオスシステムのエノンカオスを制御対象として提案法の有効性を検討した。

$$x(t+1) = 1.0 - 1.4x^2(t) + 0.3y(t) + u(t) \quad (34)$$

$$y(t+1) = x(t) \quad (35)$$

$x(t), y(t)$ は時刻 t におけるそれぞれの状態値、 $u(t)$ は時刻 t における制御入力を表わしており、制御入力が0のとき、システムはカオス状態となる (Fig. 13 参照)。また、Fig. 14 にシステムからの出力 $y(t)$ の時刻 t に対する振舞を示す。

7.1 実験方法

本数値実験例では、(26) 式における次数 n は2。リアプノフ指数は2個存在し、正のリアプノフ指数は1個である。そこで、簡単化のため、リアプノフ指数に関する評価は、この正のリアプノフ指数についてのみ目標値を与え、ほかは評価項目に含めない。すなわち、 $\lambda_d = \lambda_{d_i}$ とする。また、システムの目標出力値は $y_d = 0.2$ とし、リアプノフ指数の目標値は $\lambda_d = \{-0.05, 0, +0.05\}$ の3ケースを考慮する。またシステムの時間ステップを $T = 1000$ とし、 t の最終時刻 $t = 1000$ を t_f とする。

Fig. 15 Comparison of values of criterion function E between conventional method and proposed

(a)

(b)

Fig. 16 Robust performance for (a) $\lambda_d = -0.05$, (b) $\lambda_d = 0$

7.2 制御性能

Fig. 15 は学習ステップごとの評価指標の変化を示したものである。Fig. 15 および、Fig. 16 における $y_d = 0.2$ とした場合の誤差よりローレンツカオス同様、学習においては従来法が早く学習し、制御性能もすぐれている。

7.3 汎化性能

Fig. 16 は目標値 y_d に対する実際のシステムの最終時刻出力値 $y(t_f)$ を示したものであり、目標値 y_d を大きくするにつれて従来法と提案法は汎化能力において違いが明確になる。Fig. 17 は Fig. 16(b) の左端の一部を拡大したものである。ローレンツカオス同様重み X を大きくするにつれて誤差が拡大している。また Fig. 18 は中間層ノード数を変化させての汎化性能の違いを調べたものである。エノンカオスにおいてもローレンツカオス同様提案法の方が従来法と比較して優れた汎化能力を示しているのがわかる。提案法同士の比較においてもローレンツカオスと同様に、カオスの縁を考慮し

Fig. 17 Extended robust performance for $\lambda_d = 0$

(a)

(b)

Fig. 18 Comparison of the robust performance among the number of middle layer neuron: (a) Conventional method, (b) $\lambda_d = 0$

た場合が汎化性能が最も良い。また、 X の値も 1 の場合が最も優れた汎化能力を示している。また、Fig. 18 より本提案方式では中間層ノード数はエノンカオスにおいては 15 が適切であると言える。

8. 考察

計算機シミュレーション 1, 2 共通であるが、提案法の、システムのリアプノフスペクトルにおけるリアプノフ指数の最大値が零となるようリアプノフ指数の目標値を評価項目として追加した場合は、従来法に比較し、かなり汎化能力が向上した。この理由について考察する。提案法における学習終了時のシステムの最大リアプノフ指数は、ほぼ零に近い負の値となり、一方、従来法の場合は負でかなり小さな値となった。これは、提案法は、最大リアプノフ指数が負の値であることを要求するシステム出力を目標固定点へ一致させる評価と、最大リアプノフ指数の零を要求する評価を同時に考慮した結

果であり、従来法は、システム出力を目標固定点へ一致させる評価のみを考慮した結果といえる。提案法の場合、固定点の要求を満たすものの、最大リアプノフ指数は零に近く、かなり弱い固定点への拘束と考えられ、目標固定点の変更に伴う誤差のフィードバックの効果により、その変更に対応できたものと考えられる。その点、従来法は、目標固定点への拘束が強く、誤差のフィードバックの効果が及ばず、目標固定点の変更に対応できなかったものと考えられる。

9. ま と め

本論文では、リアプノフ指数を評価指標として取り入れた学習によるニューラルネットワークコントローラを用いてカオスシステムを制御し、その性能をローレンツカオス制御およびエノンカオス制御の2システムについて検討した。その結果、ローレンツカオス、エノンカオスどちらの場合においても従来法と比べ、リアプノフ指数を評価指標として取り入れた提案法の方が、汎化性能では明らかに優れていると言える結果が得られた。特に、システムをカオスの縁に保ちながら制御を行なった場合に最も優れた汎化性能を示した。

しかしながら、学習終了時の性能は従来法に比べ提案法の方が悪くなってしまうため、本提案方式は目標値との誤差よりも汎化性能が重要視される場合に有効と考えられる。

謝辞 本研究の一部は、文部科学省科学研究費補助金基盤研究B(2)(No.13450176)の補助の下に行なわれました。また、査読者の方から多くの有用なコメントをいただきました。ここに謝意を表します。

参 考 文 献

- 1) 潮 俊光：カオス制御，朝倉書店(1996)
- 2) Po ki yuen and Haim H. Bau: Controlling chaotic convection using neural nets - theory and experiments, Neural Networks, 11, 557/569 (1998)
- 3) K. Hirasawa, X. Wan, J. Murata and J. Hu: Chaos Control of Universal Learning Network (1999)
- 4) 井庭，福原：複雑系入門，NTT出版(1998)
- 5) 金子，津田：複雑系のカオスのシナリオ，朝倉出版(1997)
- 6) 木村，藤井，森：ロバスト制御，コロナ社(1994)
- 7) カオスセミナー，第5章，海文堂(1994)
- 8) 井上政義：カオスと複雑系の科学，日本実業出版(1996)
- 9) 長島，馬場：カオス入門，培風館(1996)
- 10) GL ベイカー，J.P.ゴラブ：カオス力学入門，啓学出版(1992)
- 11) 合原一幸編：応用カオス，サイエンス社(1994)
- 12) 平澤，東郷，胡，大林，村田：ニューラルネットワークの適応的ランダム探索最適化手法，計測自動制御学会論文集，34-8, 1088/1096 (1997)

[著 者 紹 介]

大 林 正 直 (正会員)

1951年生，75年九州工業大学工学部電気工学科卒業。77年九州大学大学院工学研究科修士課程電気工学専攻修了。77年(株)日立製作所入社大みか工場勤務。89年九州大学助手。97年同助教授。2001年山口大学教授。現在に至る。電気学会，システム制御情報学会，日本神経回路学会，米国電気電子学会の各会員。博士(工学)。

梅 迫 公 輔 (学生会員)

1976年生。98年広島工業大学工学部電子工学科卒業。2000年山口大学大学院理工学研究科博士前期課程知能情報システム工学専攻修了。同年同大学院理工学研究科博士後期課程システム工学専攻入学。現在に至る。

中 山 大 輔

1976年生。2000年山口大学工学部知能情報システム工学科卒業。2002年山口大学大学院理工学研究科博士前期課程知能情報システム工学専攻修了。同年(株)日立情報システムズ入社。現在に至る。