

# 確率一般化学習ネットワークとその非線形制御システムへの応用

金 春 植<sup>†</sup> 平澤 宏太郎<sup>†</sup> 胡 敬 焜<sup>†</sup>  
村 田 純 一<sup>†</sup> 松 岡 拓 哉<sup>†</sup>

本論文では、確率信号の処理能力を備えた確率一般化学習ネットワーク (Probabilistic Universal Learning Networks: PrULNs) を提案する。確率一般化学習ネットワークは一般化学習ネットワーク (Universal Learning Networks: ULNs) の拡張である。一般化学習ネットワークは非線形大規模複雑系のモデリングや制御系設計のための一般的な枠組みを提供しており、多くのニューラルネットワークのクラスをカバーしている。ULNs のために一般化された学習アルゴリズムが開発されているが、ほとんどの学習ネットワークがこのアルゴリズムによって学習できる。しかしながら、ULNs は確率信号が処理できない。ここで提案する PrULNs は信号の統計的特性量を計算する機構を備えており、また、ネットワークによって処理された信号が所望の確率特性を持つようにパラメータの学習をすることもできる。PrULNs は次のような問題を解決するのに役立つ：(1) 学習ネットワークの汎化能力の向上；(2) 従来の確率的制御より進んだ制御；(3) カオスのような複雑系の設計。

## Probabilistic Universal Learning Network and Its Application to a Nonlinear Control System

CHUNZHI JIN,<sup>†</sup> KOTARO HIRASAWA,<sup>†</sup> JINGLU HU,<sup>†</sup> JUNICHI MURATA<sup>†</sup>  
and TAKUYA MATSUOKA<sup>†</sup>

Probabilistic Universal Learning Networks (PrULNs) are proposed which are learning networks with capability of dealing with stochastic signals. PrULNs are extension of Universal Learning Networks (ULNs). ULNs form a superset of neural networks and were proposed to provide a universal framework for modeling and control of nonlinear large-scale complex systems. A generalized learning algorithm has been devised for ULNs which can also be used in a unified manner for almost all kinds of learning networks. However, the ULNs can not deal with stochastic variables. Specific value of a stochastic signal can be propagated through a ULN, but the ULN does not provide any stochastic characteristic of the signals propagating through it. The PrULNs proposed here are equipped with machinery to calculate stochastic properties of signals and to train network parameters so that the signals behave with the pre-specified stochastic properties. The PrULNs will contribute to the solution of the following problems: (1) improving the generalization capability of the learning networks, (2) more sophisticated stochastic control than the conventional stochastic control, (3) designing problem for the complex systems such as chaotic systems.

### 1. はじめに

本論文は、筆者らの提案してきた一般化学習ネットワークの確率バージョンへの拡張とその非線形制御システムへの応用について述べたものである。

一般化学習ネットワーク<sup>1)~5)</sup> は物理現象を表す微分方程式やそのコントローラあるいは各種の先見情報等を自然な形でモデル化し制御するための一般的な枠組みで、ロバスト制御<sup>6)</sup>、カオス制御<sup>7)</sup>等に広く活用

されている。

動的システムをモデル化し制御するために、従来から制御工学では状態方程式が広く活用されているが、状態方程式による枠組みと一般化学習ネットワークの枠組みに数学的な本質的相違があるわけではない。

しかし、非線形システムの制御にニューラルネットワークを活用する事例<sup>8)~10)</sup>が増加していることから推察されるように、次のような理由でネットワーク表現によるシステムのモデル化と制御の研究が活発になりつつある。(1) ネットワーク表現を活用すると、微分方程式で表されている物理現象からそれを制御するためのコントローラまでを統一した枠組みでモデル

<sup>†</sup> 九州大学大学院  
Graduate School, Kyushu University

化できる<sup>1)</sup>。(2) 確かな情報が不在の場合はニューラルネットワークでブラックボックス的にモデル化し、また、先見情報を活用できる場合にはファジイネットワーク<sup>11),12)</sup>でシステムを構成する等、応用に依存した適切なモデル化が可能となる。(3) パラメータの学習に関してはネットワークのノード関数や接続に依存しない一般的な順伝播・逆伝播の学習アルゴリズムが研究<sup>1),2),13)</sup>されているので、同定問題や制御問題あるいはシステムの形態に依存しない学習機構を構築できる。

したがって、本論文では、動的システムのモデル化と制御の枠組みとして、静的・動的構成、順伝播・逆伝播学習、高次微分計算、マルチブランチ、任意の非線形ノード、任意の遅れ時間等で、一般化した一般化学学習ネットワーク<sup>1)~5)</sup>を使用することにする。

さて、確率一般化学学習ネットワーク<sup>14)</sup>は一般化学学習ネットワークの確率バージョンである。通常のネットワークでは確定信号がネットワークを伝播するのであるが、確率一般化学学習ネットワークでは確率信号がネットワーク内を伝播する。

制御工学では従来から統計的自動制御理論<sup>15)</sup>、確率システム理論<sup>16)</sup>として確率信号を含むシステムの研究が活発に行われてきた。そこでは、伊藤形確率微分方程式に基づく推定理論、安定性、最適制御等の検討が行われているが、いずれも計算の複雑性から実システムの設計に関しては線形システムの議論が中心である。

そこで、確率一般化学学習ネットワークでは、一般化学学習ネットワークと同様、ネットワーク表現をベースにして、比較の見通しのきく形で非線形システムのモデル化と制御が可能な枠組みを提供することを試みている。

基本的な考え方は古く、静的な非線形写像によって確率特性を表すモーメントがどのように変換されるかを検討した Hahn ら<sup>17)</sup>までさかのぼる。

確率一般化学学習ネットワークでは、この静的な写像のアルゴリズムを動的な非線形写像、すなわち、動的な非線形ネットワークにおける写像にまで拡張しているが、そこで活用されている基本技術はテラー展開と高次微分である。

ネットワークの高次微分を計算するのは静的なネットワークでも一般に非常に困難とされている<sup>18),19)</sup>が、一般化学学習ネットワークでは動的なネットワークについても順伝播と逆伝播アルゴリズムによって高次微分を体系的に計算する手法が確立されている<sup>1),2)</sup>。したがって、一般化学学習ネットワークの高次微分計算によっ

てはじめて非線形動的システムの確率信号の伝播の解析が比較的容易に行えるようになったといえる。

また、ネットワーク表現では、統計的自動制御理論、確率システム理論と異なり、システムの最適化あるいは最適制御はネットワークの中の未知パラメータの最適値探索という形式で定式化されるのが特徴である。これは、コントローラの構造を定め、そのパラメータを最適化することによって最適なコントローラを設計することによる。

したがって、前述のような、ネットワークのノード関数や接続に依存しない順伝播・逆伝播の学習アルゴリズムがあれば、各種の形態のシステムの最適パラメータの探索が可能となる。なお、順伝播・逆伝播の学習アルゴリズムでは高次微分の計算も可能であるが、3次以上の高次微分の計算が煩雑な場合にはランダム探索等の学習も可能である。

本論文のシミュレーションでは、2次微分までを順伝播・逆伝播の学習アルゴリズムを活用し、3次以上の高次微分の計算はランダム局所探索の一種である適応的ランダム探索最適化手法 RasID (Random Search with Intensification and Diversification)<sup>20),21)</sup>を活用している。

本論文では、一般化学学習ネットワークを要約した後、一般化学学習ネットワーク上で確率信号が伝播する場合に、各時刻での各ノード出力の集合平均、共分散を計算する方式、また、平均および共分散を各々の目標値に近づけるためのネットワーク内のパラメータの学習方式を述べる。

ネットワーク内のパラメータの学習は、前述のように制御システムのコントローラの最適設計に対応するので、ノイズの混入した非線形動的システムのコントローラの設計を行うコンピュータシミュレーションによって理論の有用性を実証する。

## 2. 一般化学学習ネットワーク要約

一般化学学習ネットワークは、前述のように、微分方程式で表されている物理現象からそれを制御するコントローラまで、統一した枠組みでモデル化する。不確かな情報から先見情報までを自在にネットワークに組み込む。ネットワークのノード関数や接続に依存しない一般的な順伝播・逆伝播の学習アルゴリズムを開発する等を目指したシステムのモデル化と制御のための枠組みで、図1に示すように、微分可能な任意の非線形処理ノード  $j$  が、任意のサンプル遅れ時間  $D_{ij}(p)$  を介して任意にマルチ接続された離散時間型のネットワークである。このネットワーク上で、静的・動的構

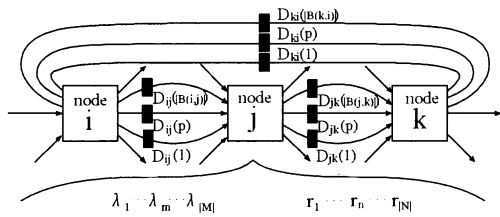


図1 一般化学学習ネットワークの構造  
Fig. 1 Structure of the ULNs.

成、階層型・リカレント型構成、多重ブランチ等の各種の構成を統一的に扱える学習アルゴリズムと高次微分を体系的に計算する手法が提案されている。応用については、2次微分を活用したロバスト制御<sup>6)</sup>およびカオス制御<sup>7)</sup>の事例が報告されている。

一般化学学習ネットワークの基本式は

$$h_j(t) = f_j(\{h_i(t - D_{ij}(p)) | i \in JF(j), p \in B(i, j)\}, \{r_n(t) | n \in N(j)\}, \{\lambda_m(t) | m \in M(j)\}). \quad (1)$$

となる。ここで

- $h_j(t)$ :  $j$  ノードの  $t$  時刻の出力値
- $r_n(t)$ :  $n$  外部入力変数の  $t$  時刻の値
- $\lambda_m(t)$ :  $m$  パラメータの  $t$  時刻の値
- $f_j$ :  $j$  ノードの非線形関数
- $D_{ij}(p)$ :  $i$  ノードより  $j$  ノードへの  $p$  番目のブランチの遅れ時間
- $J$ : ノード番号の集合
- $JF(j)$ :  $j$  ノードへ接続するノード番号の集合
- $N(j)$ :  $j$  ノードへ接続する外部入力変数番号の集合
- $M(j)$ :  $j$  ノード出力を偏微分可能なパラメータの番号の集合
- $B(i, j)$ :  $i$  ノードから  $j$  ノードへ接続するブランチの番号の集合
- $T$ : サンプル時刻の集合

である。

制御系の場合には、外部入力変数  $r_n(t)$  は目標値、外乱に対応し、また、各ノードは制御対象やコントローラを表現する要素に対応する。したがって、図1の構成により制御対象からそのコントローラまでを統一した枠組みで自在にモデル化できる。制御対象が未知の場合には、これをニューラルネットワークやファジネットワークでモデル化することも容易である。また、一般化学学習ネットワークでは、システム上で定義された一般的な評価指標  $E$  に対してコントローラ

内の未知パラメータ  $\lambda_1(t), \lambda_2(t), \dots, \lambda_m(t)$  の高次オーダー微分  $\partial^{+m} E / \partial \lambda_1(t) \partial \lambda_2(t) \dots \partial \lambda_m(t)$  を順伝播・逆伝播で体系的に計算する手法を提案している。したがって、これらを利用して評価指標が最小となるコントローラのパラメータを探索することが可能である。換言すると、制御対象とコントローラの両者をまとめて1個のネットワークと考え、このネットワーク上の信号の伝播により、任意のノード出力  $h$  について  $\lambda$  に関する高次微分が計算できる。 $E$  の中にこれら高次微分が含まれていても、さらにそのもう1次高い次数の微分が計算できるので、勾配法による  $E$  の最小化が可能である<sup>6), 7)</sup>。

以下本章では、確率一般化学学習ネットワークの高次微分の計算に必要な1次微分

$$P_1(k, t, x_\alpha(t_\ell)) = \frac{\partial^+ h_k(t)}{\partial x_\alpha(t_\ell)}$$

2次微分

$$P_2(k, t, x_\alpha(t_\ell), x_\beta(t_m)) = \frac{\partial^{+2} h_k(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)}$$

3次微分

$$P_3(k, t, x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)) = \frac{\partial^{+3} h_k(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m) \partial x_\gamma(t_n)}$$

を順伝播のアルゴリズムにより計算する方式を整理する。ここで、 $x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)$  は各々  $t_\ell, t_m, t_n$  時刻の確率変数  $x_\alpha, x_\beta, x_\gamma$  を表す記号である。なお、1次微分の場合には一般に逆伝播を多用しているが、2次以上の高次微分の計算には計算負荷が増加するので<sup>1), 2)</sup>、順伝播アルゴリズムを活用するのが好ましい。

$$P_1(k, t, x_\alpha(t_\ell)) = \sum_{j \in JF(k)} \sum_{p \in B(j, k)} \left\{ \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \times P_1(j, t - D_{jk}(p), x_\alpha(t_\ell)) \right\} + \frac{\partial h_k(t)}{\partial x_\alpha(t_\ell)} \quad (2)$$

$(k \in J, t \in T, t_\ell \leq t)$

$$P_2(k, t, x_\alpha(t_\ell), x_\beta(t_m)) = \sum_{j \in JF(k)} \sum_{p \in B(j, k)} \left\{ \partial^+ \left( \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \right) \times P_1(j, t - D_{jk}(p), x_\alpha(t_\ell)) \right\}$$

$$\begin{aligned}
& + \sum_{j \in JF(k)} \sum_{p \in B(j,k)} \left\{ \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \right. \\
& \times P_2(j, t - D_{jk}(p), x_\alpha(t_\ell), x_\beta(t_m)) \left. \right\} \\
& + \frac{\partial^+ \left( \frac{\partial h_k(t)}{\partial x_\alpha(t_\ell)} \right)}{\partial x_\beta(t_m)} \quad (3) \\
& (k \in J, t \in T, t_\ell, t_m \leq t)
\end{aligned}$$

$$\begin{aligned}
& P_3(k, t, x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)) \\
& = \sum_{j \in JF(k)} \sum_{p \in B(j,k)} \left\{ \frac{\partial^{+2} \left( \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \right)}{\partial x_\beta(t_m) \partial x_\gamma(t_n)} \right. \\
& \times P_1(j, t - D_{jk}(p), x_\alpha(t_\ell)) \left. \right\} \\
& + \sum_{j \in JF(k)} \sum_{p \in B(j,k)} \left\{ \frac{\partial^+ \left( \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \right)}{\partial x_\beta(t_m)} \right. \\
& \times P_2(j, t - D_{jk}(p), x_\alpha(t_\ell), x_\gamma(t_n)) \left. \right\} \\
& + \sum_{j \in JF(k)} \sum_{p \in B(j,k)} \left\{ \frac{\partial^+ \left( \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \right)}{\partial x_\gamma(t_n)} \right. \\
& \times P_2(j, t - D_{jk}(p), x_\alpha(t_\ell), x_\beta(t_m)) \left. \right\} \\
& + \sum_{j \in JF(k)} \sum_{p \in B(j,k)} \left\{ \frac{\partial h_k(t)}{\partial h_j(t - D_{jk}(p))} \right. \\
& \times P_3(j, t - D_{jk}(p), x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)) \left. \right\} \\
& + \frac{\partial^{+2} \left( \frac{\partial h_k(t)}{\partial x_\alpha(t_\ell)} \right)}{\partial x_\beta(t_m) \partial x_\gamma(t_n)} \quad (4) \\
& (k \in J, t \in T, t_\ell, t_m, t_n \leq t)
\end{aligned}$$

### 3. ネットワークにおけるノード出力の平均と共分散の計算

確率一般化学習ネットワークは一般化学習ネットワークの確率バージョンであり、一般化学習ネットワークの上で、確率信号の伝播と確率特性量を含む評価指標を最小化する学習アルゴリズムを取り入れたものである。確率一般化学習ネットワークの各ノード、ブランチおよびその接続のトポロジーは一般化学習ネットワークと変わりがないが、確率変数を扱い、その伝播と学習のアルゴリズムが整備されていることを明確に

する意味で、確率一般化学習ネットワークと呼んでいる。したがって、一般化学習ネットワークは確率一般化学習ネットワークのサブセットである。本章では確率一般化学習ネットワークにおける確率信号の伝播、つまり入力確率変数の平均・分散等の確率特性量からノード出力変数の平均・分散等を計算する手法について説明し、次章で確率特性量を含む評価指標を最小化する学習アルゴリズムについて説明する。

今、確率一般化学習ネットワーク上の  $p$  ノードの出力を動的ネットワーク、静的ネットワークに分けて各々式 (5), (6) で表すものとする。

$$h_p(t) = \text{Dynamic}(\{x_\alpha(t_\ell) \mid \alpha = 1, 2, \dots, \ell = 1, 2, \dots\}) \quad (5)$$

$$h_p = \text{Static}(\{x_\alpha \mid \alpha = 1, 2, \dots\}) \quad (6)$$

式 (5), (6) では  $p$  ノードの  $t$  時刻の出力  $h_p(t)$  あるいは  $p$  ノードの出力  $h_p$  が各々確率一般化学習ネットワーク上の確率変数  $x_\alpha(t_\ell)$ ,  $x_\alpha$  のみによって表現されており、便宜上その他の変数は省略されている。

また、*Dynamic* は  $x_\alpha(t_\ell)$  から  $h_p(t)$  への動的な非線形ネットワークによる写像を、*Static* は  $x_\alpha$  から  $h_p$  への静的な非線形ネットワークによる写像を表している。具体的なシステムと式 (5), (6) の対応関係はたとえば次のようになる。動的システムを表現する式 (5) では、確率変数  $x_\alpha(t_\ell)$  はたとえば、制御システムの入力値、外乱、あるいは確率変動をするシステムのパラメータ等と考えられる。これより、目標値からシステムパラメータまでの幅広い確率変数を統一的に取り扱うことを可能としている。

また、静的システムを表現する式 (6) はたとえば電力システムの負荷予測問題<sup>23)</sup>に適用可能である。 $x_\alpha$  は電力負荷を予測するための要因、 $h_p$  が電力負荷予測値であるが、予測要因は一般に確率変数と考えられるため電力負荷予測値も確率変数と考えるのが自然である。また、動的システムのフィードバック接続を削除し、ブランチの遅れ時間を零にすれば、一般に動的システムは静的システムに変換される。したがって、静的システムは動的システムの特例と考えることができるので、以下主に動的システムについて検討を行う。

本章では、まず確率変数  $x_\alpha(t_\ell)$  の平均と共分散よりノード出力  $h_p(t)$  の平均と共分散を計算する手法を求め、次にノード出力  $h_p(t)$  の平均と共分散が目標値となるように、動的な非線形ネットワークのパラメータを計算する手法、換言すると、制御システムの場合には非線形コントローラのパラメータを計算する手法を示す。

これにより、制御対象とそのコントローラをネット

ワーク表現した一般化学習ネットワークにノイズ等の確率信号が混入した確率システムの最適制御が可能となる。その結果、提案する確率一般化学習ネットワークでは、従来の線形システム、2次形式制御規範、正規形外乱等の制約を超えたより一般的な非線形確率システムの最適制御が実現できることになる。なお、線形システム、2次形式規範、正規形外乱の制約を超えられる理由は、後述の動的非線形ネットワークにおけるモーメントの写像の計算が高次微分を使用して可能になったこととコントローラの構造を決めてその最適パラメータの探索により最適コントローラを設計するネットワークの枠組みを採用しているためである。

### 3.1 平均の計算

以下の表記法を使用する。

$E[x_\alpha(t_\ell)] :$	$x_\alpha(t_\ell)$ の集合平均
$E[h_p(t)] :$	$h_p(t)$ の集合平均
$Cov[x_\alpha(t_\ell), x_\beta(t_m)] :$	$x_\alpha(t_\ell)$ と $x_\beta(t_m)$ の 共分散 = $E[(x_\alpha(t_\ell) - E[x_\alpha(t_\ell)])$ $\cdot (x_\beta(t_m) - E[x_\beta(t_m)])]$
$Cov[h_p(t), h_q(s)] :$	$h_p(t)$ と $h_q(s)$ の共分散 $= E[(h_p(t) - E[h_p(t)])$ $\cdot (h_q(s) - E[h_q(s)])]$
$Var[x_\alpha(t_\ell)] :$	$x_\alpha(t_\ell)$ の分散 = $E[(x_\alpha(t_\ell) - E[x_\alpha(t_\ell)])^2]$
$Var[h_p(t)] :$	$h_p(t)$ の分散 $= E[(h_p(t) - E[h_p(t)])^2]$
$\mu_3[x_\alpha(t_\ell), x_\beta(t_m),$ $x_\gamma(t_n)] :$	$x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)$ の 3 次中心モーメント = $E[(x_\alpha(t_\ell) - E[x_\alpha(t_\ell)])$ $\cdot (x_\beta(t_m) - E[x_\beta(t_m)])$ $\cdot (x_\gamma(t_n) - E[x_\gamma(t_n)])]$

はじめに  $h_p(t)$  の集合平均の計算を行う。

式 (5) を  $E[x_\alpha(t_\ell)]$  を中心に 2 次の項までのテーラー展開を行うと

$$\begin{aligned}
 h_p(t) &= \text{Dynamic}(\{E[x_\alpha(t_\ell)] \\
 &\quad | \alpha = 1.2, \dots, \ell = 1.2, \dots\}) \\
 &\quad + \sum_{\alpha, \ell} \frac{\partial^+ h_p(t)}{\partial x_\alpha(t_\ell)} (x_\alpha(t_\ell) - E[x_\alpha(t_\ell)]) \\
 &\quad + \frac{1}{2} \sum_{\alpha, \beta, \ell, m} \frac{\partial^{+2} h_p(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \\
 &\quad \cdot (x_\alpha(t_\ell) - E[x_\alpha(t_\ell)]) \\
 &\quad \cdot (x_\beta(t_m) - E[x_\beta(t_m)])
 \end{aligned} \tag{7}$$

となる。

2 次のテーラー展開で不十分な場合は、さらに高次の項まで使用することになるが、理論展開上両者の基本的違いはないため、以降平均については 2 次、共分散については 3 次で近似するものとする。

式 (7) の  $\partial^+ h_p(t)/\partial x_\alpha(t_\ell)$ ,  $\partial^{+2} h_p(t)/\partial x_\alpha(t_\ell) \partial x_\beta(t_m)$  は Werbos の提案したオーダード微分<sup>22)</sup>であり、他の変数を固定して  $x_\alpha(t_\ell)$ ,  $x_\beta(t_m)$  を変動した場合の  $h_p(t)$  の変動分を表している。また、オーダード微分は  $x_\alpha(t_\ell)$  の平均値  $E[x_\alpha(t_\ell)]$  で評価するものとする。

式 (7) の集合平均をとると、式 (7) 右辺第 2 項は零となるため、

$$\begin{aligned}
 E[h_p(t)] &= \text{Dynamic}(\{E[(x_\alpha(t_\ell)] \\
 &\quad | \alpha = 1.2, \dots, \ell = 1.2, \dots\}) \\
 &\quad + \frac{1}{2} \sum_{\alpha, \beta, \ell, m} \frac{\partial^{+2} h_p(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \\
 &\quad \cdot \text{Cov}[x_\alpha(t_\ell), x_\beta(t_m)]
 \end{aligned} \tag{8}$$

となる。

式 (8) が  $Cov[x_\alpha(t_\ell), x_\beta(t_m)]$  より  $h_p(t)$  の集合平均を計算するための基本式である。

### 3.2 共分散の計算

次に  $h_p(t)$  と  $h_q(s)$  の共分散の計算を行う。式 (7)、(8) より

$$\begin{aligned}
 h_p(t) - E[h_p(t)] &= \sum_{\alpha, \ell} \frac{\partial^+ h_p(t)}{\partial x_\alpha(t_\ell)} (x_\alpha(t_\ell) - E[x_\alpha(t_\ell)]) \\
 &\quad + \frac{1}{2} \sum_{\alpha, \beta, \ell, m} \frac{\partial^{+2} h_p(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \\
 &\quad \cdot (x_\alpha(t_\ell) - E[x_\alpha(t_\ell)]) \\
 &\quad \cdot (x_\beta(t_m) - E[x_\beta(t_m)]) \\
 &\quad - \frac{1}{2} \sum_{\alpha, \beta, \ell, m} \frac{\partial^{+2} h_p(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \\
 &\quad \cdot \text{Cov}[x_\alpha(t_\ell), x_\beta(t_m)]
 \end{aligned} \tag{9}$$

となる。同様にして、

$$\begin{aligned}
 h_q(s) - E[h_q(s)] &= \sum_{\alpha, \ell} \frac{\partial^+ h_q(s)}{\partial x_\alpha(t_\ell)} (x_\alpha(t_\ell) - E[x_\alpha(t_\ell)]) \\
 &\quad + \frac{1}{2} \sum_{\alpha, \beta, \ell, m} \frac{\partial^{+2} h_q(s)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \\
 &\quad \cdot (x_\alpha(t_\ell) - E[x_\alpha(t_\ell)]) \\
 &\quad \cdot (x_\beta(t_m) - E[x_\beta(t_m)])
 \end{aligned}$$

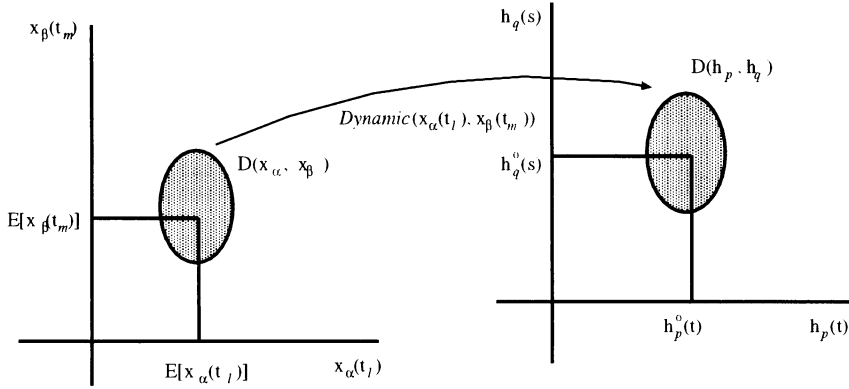


図2 写像  $Dynamic(\cdot)$  による  $D(x_\alpha, x_\beta)$  から  $D(h_p, h_q)$  への変換  
 Fig. 2 Transformation from  $D(x_\alpha, x_\beta)$  to  $D(h_p, h_q)$  by mapping  $Dynamic(\cdot)$ .

$$-\frac{1}{2} \sum_{\alpha, \beta, \ell, m} \frac{\partial^{+2} h_q(s)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \cdot Cov[x_\alpha(t_\ell), x_\beta(t_m)] \quad (10)$$

となる。3次の項まで考慮して、式(9)、(10)より  $h_p(t)$  と  $h_q(s)$  の共分散を求めると

$$\begin{aligned} & Cov[h_p(t), h_q(s)] \\ &= E[(h_p(t) - E[h_p(t)])(h_q(s) - E[h_q(s)])] \\ &= \sum_{\alpha, \beta, \ell, m} \frac{\partial^+ h_p(t)}{\partial x_\alpha(t_\ell)} \frac{\partial^+ h_q(s)}{\partial x_\beta(t_m)} \\ & \cdot Cov[x_\alpha(t_\ell), x_\beta(t_m)] \\ &+ \frac{1}{2} \sum_{\alpha, \beta, \gamma, \ell, m, n} \frac{\partial^{+2} h_q(s)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \frac{\partial^+ h_p(t)}{\partial x_\gamma(t_n)} \\ & \cdot \mu_3[x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)] \\ &+ \frac{1}{2} \sum_{\alpha, \beta, \gamma, \ell, m, n} \frac{\partial^{+2} h_p(t)}{\partial x_\alpha(t_\ell) \partial x_\beta(t_m)} \frac{\partial^+ h_q(s)}{\partial x_\gamma(t_n)} \\ & \cdot \mu_3[x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)] \quad (11) \end{aligned}$$

となる。式(11)が  $Cov[x_\alpha(t_\ell), x_\beta(t_m)]$  および  $\mu_3[x_\alpha(t_\ell), x_\beta(t_m), x_\gamma(t_n)]$  から  $h_p(t)$  と  $h_q(s)$  の共分散を計算するための基本式である。

#### 4. パラメータの学習アルゴリズム

式(5)、(6)の写像  $Dynamic$  あるいは  $Static$  に含まれるパラメータ  $\lambda_m$  の学習問題を考える。パラメータ最適化のための評価指標  $L$  として下記を考える。

$$\begin{aligned} L &= \sum_p \sum_t \mu_p(t) (E[h_p(t)] - h_p^o(t))^2 \\ &+ \sum_p \sum_q \sum_t \sum_s \mu_{pq}(t, s) \\ & \cdot (Cov[h_p(t), h_q(s)] - h_{pq}^o(t, s))^2 \quad (12) \end{aligned}$$

ここで

$h_p^o(t)$  :  $E[h_p(t)]$  の目標値  
 $h_{pq}^o(t, s)$  :  $Cov[h_p(t), h_q(s)]$  の目標値  
 $\mu_p(t), \mu_{pq}(t, s)$  : 重み係数である。

式(12)は写像  $Dynamic$  により計算される一般化学習ネットワークの  $t$  時刻  $p$  ノードの出力  $h_p(t)$  の平均値  $E[h_p(t)]$  が目標値  $h_p^o(t)$  に、また、 $h_p(t)$  と  $h_q(s)$  の共分散  $Cov[h_p(t), h_q(s)]$  が目標値  $h_{pq}^o(t, s)$  に近づくための指標である。

式(12)の評価指標により確率信号の混在したシステムの最適化が可能となり、制御システムの場合にはコントローラのパラメータの最適化を求めることができるようになる。従来の確率システム理論とは異なり、提案する評価指標では、平均2乗誤差のみでなく、共分散の項を考慮しているのが特徴である。たとえば、従来の最小分散制御では  $\sum_p \sum_t E[(h_p(t) - h_p^o(t))^2]$  を最小にするようにコントローラが設計されている。これは  $\sum_p \sum_t (E[h_p(t)] - h_p^o(t))^2 + \sum_p \sum_t Var[h_p(t)]$  となるので、提案方式と従来方式との違いは、非線形システムに対応できるという本質的な相異のほかに、評価指標の中に分散のみでなく共分散を考慮していること、共分散の目標値は一般に零でないこと、重み係数が1でないこと等となる。

図2は  $x_\alpha(t_\ell), x_\beta(t_m)$  の等確率線  $D(x_\alpha, x_\beta)$  とパラメータ  $\lambda_m$  学習後の  $h_p(t), h_q(s)$  の等確率線  $D(h_p, h_q)$  の関係を描いたものである。換言すると、学習の結果  $D(x_\alpha, x_\beta)$  内の領域が  $D(h_p, h_q)$  内の領域に写像  $Dynamic$  により変換される確率が高いことを示している。

さて、パラメータの学習は一般化学習ネットワークの学習と同様勾配法により次のように行う。

$$\lambda_m \leftarrow \lambda_m - \delta \frac{\partial^+ L}{\partial \lambda_m} \quad (13)$$

$\delta > 0$ : 学習係数

式 (13) の  $\partial^+ L / \partial \lambda_m$  は式 (12) より

$$\begin{aligned} & \frac{\partial^+ L}{\partial \lambda_m} \\ &= 2 \sum_p \sum_t \mu_p(t) (E[h_p(t)] - h_p^c(t)) \frac{\partial^+ E[h_p(t)]}{\partial \lambda_m} \\ & \quad + 2 \sum_p \sum_q \sum_t \sum_s \mu_{pq}(t, s) \\ & \quad (Cov[h_p(t), h_q(s)] - h_{pq}^c(t, s)) \\ & \quad \frac{\partial^+ Cov[h_p(t), h_q(s)]}{\partial \lambda_m} \quad (14) \end{aligned}$$

となる。また、式 (14) の  $\partial^+ E[h_p(t)] / \partial \lambda_m$ 、 $\partial^+ Cov[h_p(t), h_q(s)] / \partial \lambda_m$  は式 (8)、(11) より

$$\begin{aligned} & \frac{\partial^+ E[h_p(t)]}{\partial \lambda_m} \\ &= \frac{\partial^+ h_p(t)}{\partial \lambda_m} + \frac{1}{2} \sum_{\alpha, \beta, t, m} \frac{\partial^+ h_p(t)}{\partial x_\alpha(t) \partial x_\beta(t_m) \partial \lambda_m} \\ & \quad \cdot Cov[x_\alpha(t), x_\beta(t_m)] \quad (15) \end{aligned}$$

$$\begin{aligned} & \frac{\partial^+ Cov[h_p(t), h_q(s)]}{\partial \lambda_m} \\ &= \sum_{\alpha, \beta, t, m} \left[ \frac{\partial^+ h_p(t)}{\partial x_\alpha(t) \partial \lambda_m} \frac{\partial^+ h_q(s)}{\partial x_\beta(t_m)} \right. \\ & \quad \left. + \frac{\partial^+ h_p(t)}{\partial x_\alpha(t)} \frac{\partial^+ h_q(s)}{\partial x_\beta(t_m) \partial \lambda_m} \right] \\ & \quad Cov[x_\alpha(t), x_\beta(t_m)] \\ & \quad + \frac{1}{2} \sum_{\alpha, \beta, \gamma, t, m, n} \\ & \quad \left[ \frac{\partial^+ h_p(t)}{\partial x_\alpha(t) \partial x_\beta(t_m) \partial \lambda_m} \frac{\partial^+ h_q(s)}{\partial x_\gamma(t_n)} \right. \\ & \quad \left. + \frac{\partial^+ h_p(t)}{\partial x_\alpha(t) \partial x_\beta(t_m)} \frac{\partial^+ h_q(s)}{\partial x_\gamma(t_n) \partial \lambda_m} \right] \\ & \quad \cdot \mu_3[x_\alpha(t), x_\beta(t_m), x_\gamma(t_n)] \\ & \quad + \frac{1}{2} \sum_{\alpha, \beta, \gamma, t, m, n} \\ & \quad \left[ \frac{\partial^+ h_p(t)}{\partial x_\alpha(t) \partial x_\beta(t_m) \partial \lambda_m} \frac{\partial^+ h_q(s)}{\partial x_\gamma(t_n)} \right. \\ & \quad \left. + \frac{\partial^+ h_p(t)}{\partial x_\alpha(t) \partial x_\beta(t_m)} \frac{\partial^+ h_q(s)}{\partial x_\gamma(t_n) \partial \lambda_m} \right] \\ & \quad \cdot \mu_3[x_\alpha(t), x_\beta(t_m), x_\gamma(t_n)] \quad (16) \end{aligned}$$

となる。式 (15)、(16) の高次微分は前述の式 (2)~(4) で求めることができる。ただし、高次微分は  $x_\alpha(t)$  の平均値  $E[x_\alpha(t)]$  で評価するものとする。したがって、式 (13)~(16) と  $E[x_\alpha(t)]$  を入力とする *Dynamic*

の計算式 (5)、(8)、(11) により式 (12) を局所最小とする *Dynamic* 中のパラメータ  $\lambda_m$  を学習により求めることができる。

なお、静的システムで  $x_\alpha$ 、 $x_\beta$ 、 $x_\gamma$  が互いに独立な場合の分散を考慮した評価指標  $L$  を

$$\begin{aligned} L &= \sum_p \mu_p^1 (E[h_p] - h_p^1)^2 \\ & \quad + \sum_p \mu_p^2 (Var[h_p] - h_p^2)^2 \quad (17) \end{aligned}$$

ただし

$$\begin{aligned} h_p^1 &: E[h_p] \text{ の目標値} \\ h_p^2 &: Var[h_p] \text{ の目標値} \\ \mu_p^1, \mu_p^2 &: \text{重み係数} \end{aligned}$$

とすると、式 (14)~(16) の計算は次のように簡単になる。

$$\begin{aligned} & \frac{\partial^+ L}{\partial \lambda_m} \\ &= 2 \sum_p \mu_p^1 (E[h_p] - h_p^1) \frac{\partial^+ E[h_p]}{\partial \lambda_m} \\ & \quad + 2 \sum_p \mu_p^2 (Var[h_p] - h_p^2) \frac{\partial^+ Var[h_p]}{\partial \lambda_m} \quad (18) \end{aligned}$$

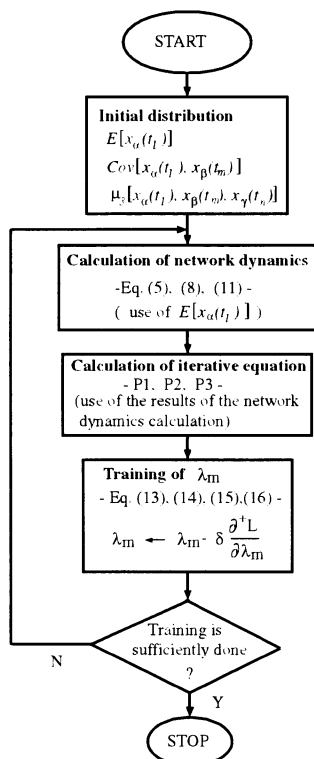
$$\begin{aligned} & \frac{\partial^+ E[h_p]}{\partial \lambda_m} \\ &= \frac{\partial^+ h_p}{\partial \lambda_m} + \frac{1}{2} \sum_\alpha \frac{\partial^+ h_p}{\partial x_\alpha^2 \partial \lambda_m} Var[x_\alpha] \quad (19) \end{aligned}$$

$$\begin{aligned} & \frac{\partial^+ Var[h_p]}{\partial \lambda_m} \\ &= 2 \sum_\alpha \left( \frac{\partial^+ h_p}{\partial x_\alpha} \right) \left( \frac{\partial^+ h_p}{\partial x_\alpha \partial \lambda_m} \right) Var[x_\alpha] \\ & \quad + \sum_\alpha \left[ \frac{\partial^+ h_p}{\partial x_\alpha^2 \partial \lambda_m} \frac{\partial^+ h_p}{\partial x_\alpha} + \frac{\partial^+ h_p}{\partial x_\alpha^2} \frac{\partial^+ h_p}{\partial x_\alpha \partial \lambda_m} \right] \\ & \quad \cdot \mu_3[x_\alpha] \quad (20) \end{aligned}$$

図 3 にパラメータ  $\lambda_m$  の学習プロセスのフローチャートを示す。なお、式 (4) の 3 次微分計算が繁雑の場合には、式 (12) の  $E[h_p(t)]$ 、 $Cov[h_p(t), h_q(s)]$  を 2 次微分を使用して式 (8)、(11) より求め、式 (12) を最小化する  $\lambda_m$  を適応的ランダム探索最適化手法 RasID<sup>20),21)</sup> (付録参照) を使用して計算することも可能である。

## 5. 非線形制御システムへの応用

本章では、ネットワーク表現された非線形クレーン制御システムのノード出力に外乱が混入する場合につ

図3 パラメータ  $\lambda_m$  の学習フローチャートFig. 3 Flowchart of training the parameter  $\lambda_m$ .

いて、確率一般化学習ネットワークを使用したコントローラの最適設計問題について検討する。コントローラはリカレントニューラルネットワークで構成し、通常のニューラルネットワーク制御と確率一般化学習ネットワークを活用した制御の比較検討を行っている。

ノイズの混入した非線形クレーンシステムとそのコントローラ内の信号の平均、共分散を比較的簡単に計算し、かつこれら平均、共分散で定義された評価指標を最適化するニューラルネットワークコントローラを求めるために、非線形クレーンとコントローラを一般化学習ネットワークでモデル化している。

もちろん、非線形クレーンシステムとそのコントローラを状態変数表現することも可能であるが、ニューラルネットワークのようなネットワークコントローラを前提とする場合には、制御対象も含めてすべてのシステムをネットワーク表現する方が、順伝播・逆伝播のネットワーク学習アルゴリズムを活用する点からも見通しが良く自然である。

また、本例題の非線形クレーンシステムはプラントモデルが既知としており、状態方程式によって表現さ

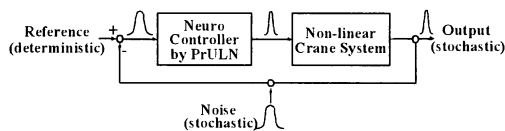


図4 ランダム入力を持つ制御系

Fig. 4 Control system with stochastic input.

れているが、未知の非線形システムの場合には、その入出力特性をニューラルネットワークでモデル化することが広く行われている。このような場合にも、制御システム全体をネットワーク表現する一般化学習ネットワークの活用が自然である。

なお、ノイズの混入した非線形確率クレーンシステムの最適化問題を状態方程式表現に基づく従来の確率システム理論で検討することも可能であるが、前述したように、システムの線形近似が必要であり、かつ2次形式制御規範に限定される等の点で課題は多い。

さて、本論文のシミュレーションでは、対象を非線形クレーンシステム、入力される確率変数を平均値および分散が既知の外乱とする。図4に示すように、クレーンシステムの出力に外乱が混入しても出力値の確率分布の平均値は本来の目標値に、また分散は可能な限り小さくなるようなコントローラを一般化学習ネットワークによって設計する。これにより一般化学習ネットワークを活用した設計方式の有用性を実証する。

### 5.1 対象システム

対象となる非線形クレーンシステムを図5に示す。このクレーンシステムの運動方程式は、台車の位置を  $x$  [m]、荷の振り角を  $\theta$  [rad]、荷の巻き上げ位置を  $\ell$  [m] とすると、

$$\frac{d^2x}{dt^2} = -\frac{mg}{M}\theta - \frac{D+G}{M}\frac{dx}{dt} + \frac{G}{M}u_1 \quad (21)$$

$$\frac{d^2\theta}{dt^2} = -\frac{M+m}{\ell M}g\theta - \frac{D+G}{\ell M}\frac{dx}{dt} + \frac{G}{\ell M}u_1 \quad (22)$$

$$\frac{d^2\ell}{dt^2} = -\frac{C+G_m}{m}\frac{d\ell}{dt} + \frac{G_m}{m}u_2 \quad (23)$$

と表される。ただし、

- $u_1, u_2$  : 横走行、巻き上げモータ入力電圧 [V]
- $C$  : 巻き上げ摩擦係数 [kg/sec]
- $M$  : 台車の質量 [kg]
- $D$  : 横走行摩擦係数 [kg/sec]
- $m$  : 荷の質量 [kg]
- $G, G_m$  : 台車・巻き上げ系の電圧-力係数 [N/V]
- $g$  : 重力加速度 [m/sec<sup>2</sup>]

である。これを離散時間化し、一般化学習ネットワークでは、6個のノードで表現した。コントローラにつ



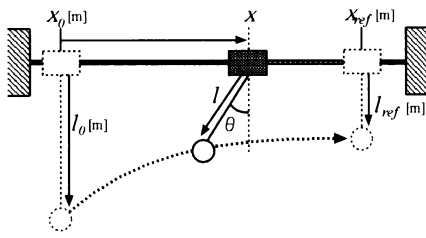


図5 非線形クレーンシステム  
Fig. 5 Nonlinear crane system.

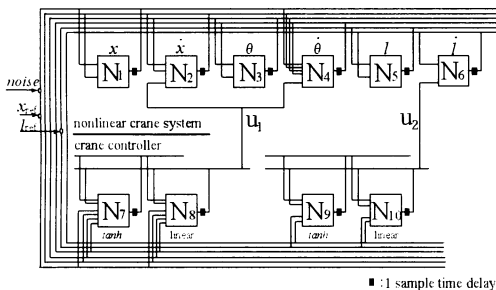


図6 ULNsを用いたクレーンシステムの制御モデル  
Fig. 6 Control model of the crane system by ULNs.

いては、入力の線形加重和をとる処理と、 $\tanh$  関数により非線形処理を行う 2 個のノードをリカレント接続したものを 2 組用意し、それぞれが  $u_1, u_2$  を出力する。各コントローラノードに接続する結合重みとノードの閾値パラメータの値は学習によって獲得する。図 6 に一般化学習ネットワークで構成されたリカレント型システムモデルを示す。遅れ時間はすべて 1 サンプル遅れとした。

図 6 において、 $N_1 \sim N_6$  が制御対象であるシステムノード、 $N_7 \sim N_{10}$  がコントローラノードである。信号の流れは、コントローラノードの出力  $u_1, u_2$  がシステムノードへ入力され、システムダイナミクスの 1 サンプル動作後、システムノード出力がコントローラ側にフィードバックされる。このとき、参照入力  $x_{ref}, l_{ref}$  とシステムのノード出力  $x, \ell$  との偏差がそれぞれのコントローラノードへの入力となる。また、今回のシミュレーションでは、全状態が観測可能なものとして制御系を構成している。数値条件は

$$\begin{aligned} C &= 0.42 \text{ [kg/sec]}, D = 300 \text{ [kg/sec]}, \\ G &= 700 \text{ [N/V]} \\ G_m &= 0.98 \text{ [N/V]}, M = 40 \text{ [kg]}, m = 2.0 \text{ [kg]} \end{aligned}$$

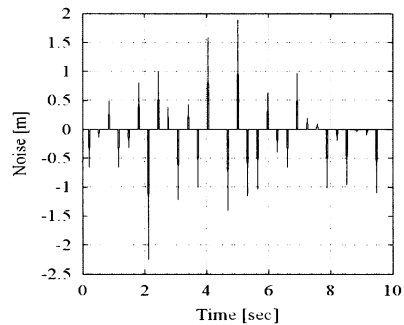


図7 ガウスノイズの時系列  
Fig. 7 Sample time series of Gaussian Noise.

$$\begin{aligned} g &= 9.8 \text{ [m/sec}^2\text{]}, \Delta t = 0.02 \text{ [sec]} \\ x_0 &= 0 \text{ [m]}, x_{ref} = 1 \text{ [m]}, \ell_0 = 1 \text{ [m]}, \\ \ell_{ref} &= 0.5 \text{ [m]} \end{aligned}$$

とした。

外乱は、 $x$  の観測値に混入するノイズを想定する。制御時間は 10 [sec] で、サンプリング総数  $T = 500$  としたが、ノイズの混入はそのうち 30 サンプル時刻とした。ノイズの大きさは、平均 0、分散 1 のガウス分布に従い、異なる時刻のノイズどうしは互いに独立とした。生成したノイズ時系列の一例を図 7 に示す。ノイズの値は、 $x$  の初期値  $x_0 = 0$ 、目標値  $x_{ref} = 1$  であることを考慮すると過大な値であるが、後の制御結果においてノイズの影響を明瞭にするためにこのような値とした。

平均と分散を制御することでノイズには十分に対処できると考えられたので、今回は共分散の計算および目標設定は行っていない。

## 5.2 評価指標の設定

今回のシミュレーションでは、分散および平均の計算の効果を個別に検証するため、評価関数を 3 種類設定してコントローラを設計した。

以下の評価指標の式において、 $T$  は制御全時刻、 $t_f$  は最終制御時刻を表す。

- ノイズが混入しないと想定する場合

式 (24) の評価指標は、最小限のエネルギーで台車と荷を目標位置に移動させ停止させるためのものであり、通常行われているニューロコントローラを設計するための評価指標である<sup>2)</sup>。

$$\begin{aligned}
E_N &= \frac{1}{2} \left[ \sum_{s \in T} \{ Q_{n1} (x_{ref} - x(s))^2 + Q_{n3} \theta(s)^2 \right. \\
&\quad + Q_{n4} \dot{\theta}(s)^2 + Q_{n5} (\ell_{ref} - \ell(s))^2 \\
&\quad + R_{n1} u_1(s)^2 + R_{n2} u_2(s)^2 \} \\
&\quad \left. + Q_{n2} \dot{x}(t_f)^2 + Q_{n6} \dot{\ell}(t_f)^2 \right] \quad (24)
\end{aligned}$$

ここで、各項の係数は

$$Q_{n1} \sim Q_{n6} = 1.0, R_{n1}, R_{n2} = 0.001 \quad (25)$$

とした。

- 分散を考慮する場合

分散のみの評価でどれほどの効果が得られるかを明らかにする目的で、式 (26)、(27) に示す評価指標を設定し、コントローラ的设计を行った。

$$L_V = E_N + E_V \quad (26)$$

$$\begin{aligned}
E_V &= \sum_{s \in T} \{ Q_{v1} \text{Var}[x(s)] + Q_{v2} \text{Var}[\dot{x}(s)] \\
&\quad + Q_{v3} \text{Var}[\theta(s)] + Q_{v4} \text{Var}[\dot{\theta}(s)] \} \quad (27)
\end{aligned}$$

$E_V$  内の各項の係数は、試行錯誤的に調整し

$$Q_{v1}, Q_{v3} = 20.0, Q_{v2}, Q_{v4} = 4.0 \quad (28)$$

とした。

- 平均と分散を考慮する場合

確率一般化学習ネットワーク理論で提案しているとおり式 (12) に従って評価指標を設定すると、以下のようになる。

$$L_{MV} = E_M + E_V \quad (29)$$

$$\begin{aligned}
E_M &= \frac{1}{2} \left[ \sum_{s \in T} \{ Q_{m1} (x_{ref} - E[x(s)])^2 \right. \\
&\quad + Q_{m3} (E[\theta(s)])^2 + Q_{m4} (E[\dot{\theta}(s)])^2 \\
&\quad + Q_{m5} (\ell_{ref} - E[\ell(s)])^2 + R_{m1} (E[u_1(s)])^2 \\
&\quad \left. + R_{m2} (E[u_2(s)])^2 \} \right. \\
&\quad \left. + Q_{m2} (E[\dot{x}(t_f)])^2 + Q_{m6} (E[\dot{\ell}(t_f)])^2 \right] \quad (30)
\end{aligned}$$

$E_V$  については、式 (27) と同じものを用いた。 $E_M$ 、 $E_V$  内の各項の係数は、試行錯誤により最適な値に調整した結果、以下の値となった。特に値を記していないものは式 (25) の対応する係数と同じ値とした。

$$\begin{aligned}
Q_{m1} &= 4.0, Q_{m5} = 2.0, \\
Q_{v1} &= 8.0, Q_{v2} = 4.0, \quad (31) \\
Q_{v3} &= 6.0, Q_{v4} = 2.0
\end{aligned}$$

これらの評価指標は RasID を用いて最適化する。

### 5.3 結果および考察

各コントローラの重み・閾値パラメータの初期値は  $[-0.1, 0.1]$  の乱数で与え、乱数のパターンを変えてそ

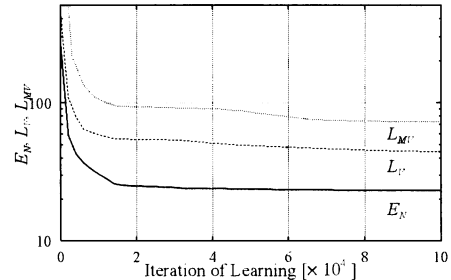
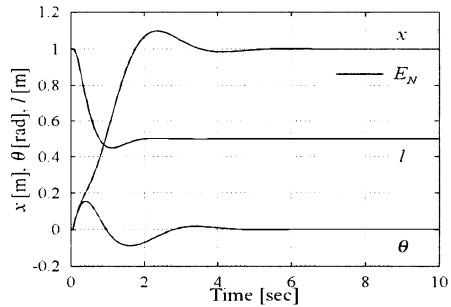
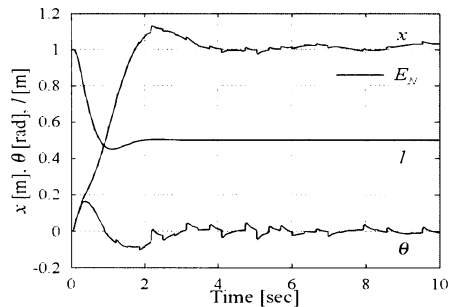


図8  $E_N, L_V, L_{MV}$  の学習曲線  
Fig. 8 Learning curve of  $E_N, L_V, L_{MV}$ .



(a) Results without Noise



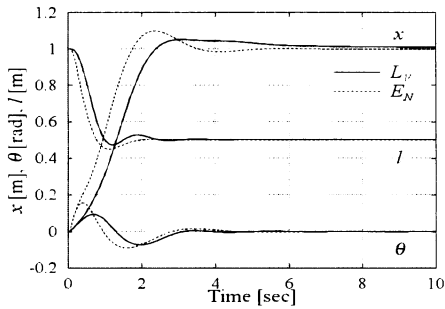
(b) Results with Noise

図9 評価関数が  $E_N$  時の制御  
Fig. 9 Control results in case of criterion  $E_N$ .

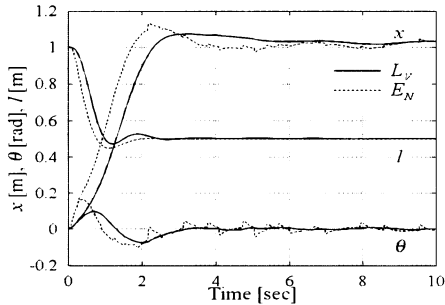
れぞれの評価指標に対して5回のシミュレーションを行ったが、いずれも評価値はほぼ同じ値に収束した。 $x, \theta, \ell$  の制御結果については5つの結果のうちの一例を掲載している。なお、学習回数は十分な収束が得られるよう、10万回実行した。

図8に、 $E_N, L_V, L_{MV}$  の学習曲線を示す。各曲線はパラメータの初期値を変更して行った5回のシミュレーションの平均値である。10万回の学習によりどの評価指標もほぼ収束し、それぞれの指標を満足するコントローラが構成されている。

次に、各評価指標による制御の結果をノイズが混入



(a) Results without Noise



(b) Results with Noise

図 10 評価関数が  $L_V$  時の制御Fig. 10 Control results in case of criterion  $L_V$ .

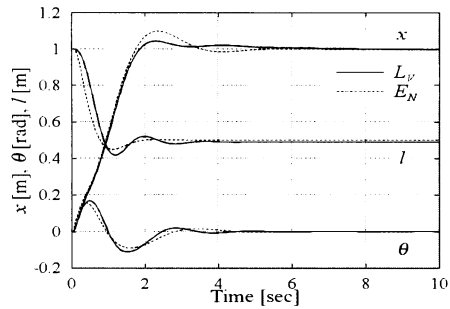
しない場合と混入する場合に分けて示す。平均 0、分散 1 のノイズのパターンを 5 種類以上生成し、学習終了後のコントローラを用いて制御を行ったが、5 種類の結果はいずれも大差ないものであった。以降の結果については、図 7 に示したノイズを使用している。

- ノイズが混入しないと想定する場合

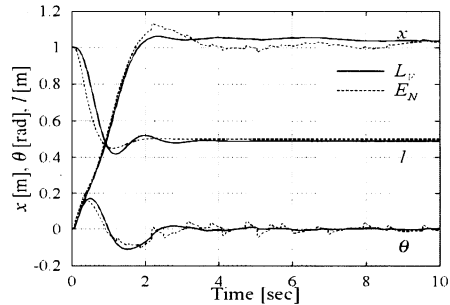
まず、確率分布を考慮しない、通常の評価指標である式 (24) に基づいて構成したコントローラによる制御の結果を示す。図 9 (a) がノイズの混入しない場合、(b) がノイズの混入した場合の制御結果である。ノイズにより  $x$  の目標値が変動しているため、台車の位置  $x$  が振動し、その影響を受けて荷の振れ角  $\theta$  も目標値から大きく外れ、ノイズに反応して小刻みに振動を続けている。

- 分散を考慮する場合

分散評価の効果を明らかにするため、 $E_N$  の結果と重ねて図 10 に示す。式 (27) を零に近づけるということはノイズに対する出力の感度をできるだけ小さくすることである。したがって、図 10 においてはノイズによる荷の振れ角  $\theta$  の振動は抑えられているが、ノイズが混入しない場合でも  $x$  の定常偏差、速応性、減衰性が犠牲となっているようすがうかがえる。



(a) Results without Noise



(b) Results with Noise

図 11 評価関数が  $L_V$  ( $x$  を重視) 時の制御Fig. 11 Control results in case of  $L_V$  ( $x$  prioritized).

$x$  の応答を重視して式 (32) のように  $Q_{n1}$  の値を大きくした場合、図 11 に示すように  $x$  の応答は改善されるが、 $l$  の評価の相対的な重みが減少したことにより、 $l$  のオーバシュート、減衰性、定常偏差等が犠牲となっている。

$$Q_{n1} = 2.0, Q_{r1}, Q_{r3} = 15.0,$$

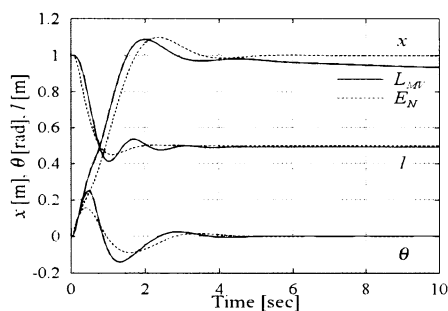
$$Q_{r2}, Q_{r4} = 5.0 \quad (32)$$

コントローラを設計するには  $\theta$  の振動抑制と  $x$ 、 $l$  の応答のトレードオフから、仕様に応じて各項の係数を決定することが肝要である。

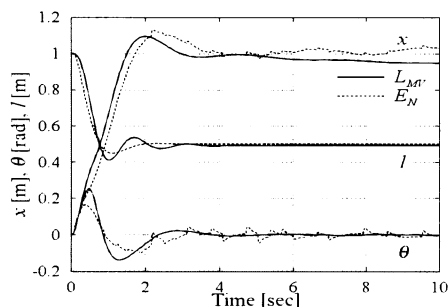
- 平均と分散を考慮する場合

結果を図 12 に示す。

図 12 では、ノイズが混入する場合としない場合の  $x$  の応答に、分散のみの評価の場合とは異なる点が見られる。ノイズは互いに独立であるという前提から、 $x$  の平均値は式 (8) を基に計算しているが、その第 1 項は、ノイズの混入がない場合のノード出力であり、第 2 項はノイズを考慮した補正項である。したがって、ノイズの補正項を考慮した平均をも評価する本ケースでは、ノイズの混入がないと、第 2 項の値の分だけ出力が目標値から外れることになる。これは逆の視点から見ると、「つねに統計的に同じノイズが混入すること



(a) Results without Noise



(b) Results with Noise

図 12 評価関数が  $L_{MV}$  時の制御Fig. 12 Control results in case of criterion  $L_{MV}$ .

によって、より望ましい値を出力する」ようなコントローラ的设计がなされているということである。

## 6. まとめ

本論文では、非線形システムのモデル化と制御を行うための一般化学習ネットワークに確率の伝播を導入した確率一般化学習ネットワーク理論について述べ、次に、これを非線形クレーンシステムにおけるコントローラ的设计問題へ適用し、平均値や分散の有効な制御が可能であることを示している。

本論文のシミュレーションでは、共分散の計算が可能であるにもかかわらず、それを活用していない。ノード間、および同じノードの2時刻間等の共分散の値をどのように有効活用するかは今後の課題である。

また、従来からロバスト制御<sup>6)</sup>やカオス制御<sup>7)</sup>などで高次微分を活用した方式が提案されているが、確率一般化学習ネットワークの枠組みはこれらの拡張であり、変数の変動を特定方向・特定の大きさに限定せず確率変動としている点に特徴がある。

今後、本手法を、確率変数の混在した各種のネットワークあるいは制御システムに展開するための研究を積極的に進めてゆく予定である。

## 参考文献

- 1) Hirasawa, K., Murata, J., Hu, J. and Jin, C.: Chaos Control on Universal Learning Networks, *IEEE Trans. Systems, Man, and Cybernetics - Part C*, Vol.30, No.1 (2000).
- 2) Hirasawa, K., Wang, X., Murata, J., Hu, J. and Jin, C.: Universal Learning Networks and its Application to Chaos Control, *Neural Networks*, Vol.13, No.2 (2000).
- 3) Hirasawa, K., Ohbayashi, M. and Murata, J.: Universal Learning Network and Computation of its Higher Order Derivatives, *Proc. 1995 IEEE International Conference on Neural Networks*, Vol.3, pp.1273-1277 (1995).
- 4) Hirasawa, K. and Ohbayashi, M.: Forward Propagation Universal Learning Network, *Proc. 1996 IEEE International Conference on Neural Networks*, pp.353-358 (1996).
- 5) Hirasawa, K., Hu, J., Ohbayashi, M. and Murata, J.: Computing High Order Derivatives in Universal Learning Networks, *Journal of Advanced Computational Intelligence*, Vol.2, No.2, pp.47-53 (1998).
- 6) 大林, 平澤: 一般化学習ネットワークの2次微分を用いた非線形ダイナミカルシステムの抑制/活性化制御, *計測と制御*, Vol.37, No.8, pp.586-600 (1998).
- 7) 古賀, 平澤, 大林, 村田: 一般化学習ネットワークにおけるカオス制御方式, *計測自動制御学会論文集*, Vol.32, No.6, pp.844-853 (1996).
- 8) Naredra, K. and Parthasathy, K.: Identification and Control of Dynamical Systems Using Neural Networks, *IEEE Trans. Neural Networks*, Vol.1, No.1, pp.4-27 (1990).
- 9) Miller III, W.T., Sutton, R.S. and Werbos, P.J. (Eds.): *Neural Networks for Control*, The MIT Press, Massachusetts (1990).
- 10) 西川, 北村: ニューラルネットと計測制御, 朝倉書店 (1995).
- 11) Lin, C.-J. and Lin, C.-T.: An ART-Based Fuzzy Adaptive Learning Control Network, *IEEE Trans. Fuzzy Systems*, Vol.5, No.4, pp.477-496 (1997).
- 12) 林, 古橋: ファジィ・ニューラルネットワーク, 朝倉書店 (1996).
- 13) 黒江: リカレントニューラルネットワークの学習法, *システム/制御/情報*, Vol.36, No.10, pp.634-643 (1992).
- 14) 平澤, 大林, 村田, 胡: 確率一般化学習ネットワーク理論, *電気学会論文誌*, Vol.118C, No.2, pp.224-231 (1998).
- 15) 榎木, 添田, 中溝: 統計的自動制御理論, コロナ社 (1966).

- 16) 砂原：確率システム理論，電子情報通信学会 (1979).
- 17) Hahn, G.J. and Shapiro, S.S.: *Stability Models in Engineering*. Wiley (1967).
- 18) Buntine, W.L. and Weigend, A.S.: Computing Second Derivatives in Feed-Forward Networks: A Review. *IEEE Trans. Neural Networks*, Vol.5, No.3 (1994).
- 19) Wang, Y.-J. and Lin, C.-T.: A Second Order Learning Algorithm for Multilayer networks based on Bloc Hessian Matrix. *Neural Networks*, No.11, pp.1607-1622 (1998).
- 20) 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，Vol.34, No.8, pp.1088-1096 (1998).
- 21) Hu, J., Hirasawa, K. and Murata, J.: RasID - Random Search for Neural Network Training. *Journal of Advanced Computational Intelligence*, Vol.2, No.4 (1998).
- 22) Werbos, P.: Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Science, Ph.D. dissertation. Harvard University (1974).
- 23) Panaweera, D.K., et al.: Effect of Probabilistic Input on Neural Network based Electric Load Forecasting. *IEEE Trans. Neural Networks*, Vol.7, No.6 (1996).

## 付 録

### A.1 適応的ランダム探索最適化手法 RasID<sup>20),21)</sup>

RasID (Random Search with Intensification and Diversification) はランダム探索の一種であるが、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した枠組みで実現する手法である<sup>20),21)</sup>。つまり、解の近傍に改善解が存在する可能性が高いときには、解近傍を適切な探索幅で重点的に集中化探索し、解がローカルミニマムに陥り、解近傍に改善解が存在しない場合には大きい探索幅で広い領域を多様に探索し、大域的な最適解を求める。以下、集中化探索と多様化探索をどのように行うのかを簡単に説明する。

#### A.1.1 アルゴリズム

パラメータベクトルを  $\Theta(k) = [\lambda_1(k), \dots, \lambda_m(k), \dots]^T$  とする。ただし、 $\Theta \in \mathcal{W}$  ( $\mathcal{W}$  はパラメータベクトルの許容領域) であり、 $k$  は  $k$  番目の探索の結果を示す。評価指標を  $\mathcal{L}$  としたときの計算手順の概要は次のようになる。

- (1) 初期値  $\Theta(0)$  を与えて、評価指標  $\mathcal{L}$  を計算する。
- (2) 探索ベクトル  $\mathbf{x}(k)$  を発生させる。このとき、

$(\Theta(k) + \mathbf{x}(k)) \notin \mathcal{W}$  ならば、手順 (3) に移る。

- (a)  $\mathcal{L}(\Theta(k) + \mathbf{x}(k))$  を計算し、 $\mathcal{L}(\Theta(k) + \mathbf{x}(k)) < \mathcal{L}(\Theta(k))$  ならば探索が成功したとし、 $y = 1$ ,  $\Theta(k+1) = \Theta(k) + \mathbf{x}(k)$  とする。そうでなければ、
- (b)  $\mathcal{L}(\Theta(k) - \mathbf{x}(k))$  を計算し、 $\mathcal{L}(\Theta(k) - \mathbf{x}(k)) < \mathcal{L}(\Theta(k))$  ならば探索が成功したとし、 $y = 1$ ,  $\Theta(k+1) = \Theta(k) - \mathbf{x}(k)$  とする。これを満たさなければ、
- (c) 探索は失敗したとして、 $y = 0$  とし、以下の条件式により  $\Theta(k+1)$  を決定する。

$$\Theta(k+1) = \begin{cases} \Theta(k), & (K_{er}^+ > K_{er} \text{ かつ} \\ & K_{er}^- > K_{er}); \\ \Theta(k) + \mathbf{x}(k), & (K_{er}^+ < K_{er}^-); \\ \Theta(k) - \mathbf{x}(k), & (K_{er}^+ \geq K_{er}^-). \end{cases}$$

ただし、 $K_{er} \geq 1$  であり、 $K_{er}^+$  および  $K_{er}^-$  は以下に定義する。

$$K_{er}^+ = \frac{\mathcal{L}(\Theta(k) + \mathbf{x}(k))}{\mathcal{L}(\Theta(k))},$$

$$K_{er}^- = \frac{\mathcal{L}(\Theta(k) - \mathbf{x}(k))}{\mathcal{L}(\Theta(k))}.$$

- (3) あらかじめ設定されていた条件 (学習回数や判定誤差など) を満たせば終了する。そうでなければ、手順 (2) に戻って繰り返す。

#### A.1.2 探索ベクトルの生成

ここでは、探索ベクトルの生成について述べる。探索ベクトルを  $\mathbf{x}(k) = [x_1, \dots, x_m, \dots]^T$  とし、その探索確率密度関数は下式で与える。

$$f(x_m) = \begin{cases} (1 - q_m)\beta e^{\beta x_m}, & \text{If } (x_m \leq 0); \\ q_m \beta e^{-\beta x_m}, & \text{If } (x_m > 0). \end{cases} \quad (33)$$

式中のパラメータ  $q_m$  と  $\beta$  は、集中化と多様化を実現するためのパラメータである。この確率密度関数を積分すれば以下の確率分布関数が得られる。

$$F(x_m) = \begin{cases} (1 - q_m)e^{\beta x_m}, & \text{If } (x_m \leq 0); \\ 1 - q_m e^{-\beta x_m}, & \text{If } (x_m > 0). \end{cases} \quad (34)$$

これより、探索ベクトル  $x_m$  は次のように表される。ただし、 $z_m$  は  $[0,1]$  の一様乱数である。

$$\mathbf{x}_m = \begin{cases} \frac{1}{\beta} \ln \left( \frac{z_m}{1 - q_m} \right), & \text{If } (0 \leq z_m \leq 1 - q_m); \\ -\frac{1}{\beta} \ln \left( \frac{1 - z_m}{q_m} \right), & \text{If } (1 - q_m < z_m \leq 1). \end{cases} \quad (35)$$

### A.1.3 探索ベクトルの生成

ここでは、集中化探索と多様化探索について簡単に説明する。式(35)を見れば、 $\beta$ の値が大きくなれば局所探索の範囲が狭くなること分かる。そこで、この $\beta$ を以下のように設定する。

$$\beta = \beta_0 + (\beta_1 - \beta_0)e^{-\phi I_{sf}}, \quad (36)$$

つまり $\phi$ と $I_{sf}$ により $\beta$ の値を変化させる。この $\phi$ と $I_{sf}$ はそれぞれ集中化探索と多様化探索に対応しており、それぞれの探索に応じて変化し、集中化と多様化の役割を演じる。

#### ● 集中化探索

集中化探索を担うパラメータ $\phi$ は以下の条件式により変更される。

$$\phi = \begin{cases} c_i \phi, & (P_{sf} \geq P_{sf0}, c_i > 1); \\ \phi, & (\phi \leq \phi_{min}); \\ c_d \phi, & (P_{sf} < P_{sf0} \\ & \text{and } \phi > \phi_{min}, c_d < 1). \end{cases} \quad (37)$$

$P_{sf0}$ はあらかじめ設定された値で、探索の成功率のしきい値である。探索が成功して、 $P_{sf} > P_{sf0}$ が成立していれば、 $\phi$ の値が大きくなり $\beta$ が小さくなり、探索を加速する集中化探索が進むようになる。反対に、ローカルミニマムに近づき、探索の失敗が増えて、 $P_{sf}$ の値が $P_{sf0}$ の値よりも小さくなれば、 $\phi$ の値が小さくなり $\beta$ が大きくなり、きめ細かな探索を行うようになる。また、 $\phi$ が決められた値( $\phi_{min}$ )よりも小さくなれば、集中化探索が進まなくなったと判断し、多様化探索に移る。

#### ● 多様化探索

多様化探索に移って、探索の失敗が続けば、式(38)により $I_{sf}$ の値が大きくなり、したがって式(36)より $\beta$ の値が小さくなり、探索範囲が広がる。このためローカルミニマムからの脱出がしやすくなる。具体的には、 $I_{sf} > I_{sfmax}$ となり探索が成功すれば、集中化探索に切り替わる。また、 $I_{sf} \leq I_{sfmax}$ で、探索が成功すれば、式(38)により $I_{sf}$ の値が小さくなる。したがって、式(36)より、 $\beta$ の値が大きくなり、探索の範囲が狭くなっていく。この結果、探索の成功

が続くようになれば、前述の $P_{sf}$ の値が大きくなるので、 $P_{sf} > P_{sf0}$ が成立し、 $\phi$ 増加し、 $\phi > \phi_{min}$ が成立した時点で、集中化探索に切り替わる。

$$I_{sf} = \begin{cases} I_{sf0}, & (\phi > \phi_{min}) \text{ or} \\ & (y = 1 \text{ and } I_{sf} > I_{sfmax}); \\ I_{sf} - \Delta I_{sf1}, & (y = 1 \text{ and } \phi \leq \phi_{min}); \\ I_{sf} + \Delta I_{sf2}, & (y = 0 \text{ and } \phi \leq \phi_{min}). \end{cases} \quad (38)$$

#### ● $q_m$ の決定法

ここでは、 $q_m$ の変更について述べる。 $q_m$ は探索ベクトルの方向を決めるパラメータであり、生成されたベクトルが探索に成功すれば、その方向のベクトルが発生する確率が高くなるように $q_m$ を変更する。

$$q_m = \begin{cases} \alpha q_m, & (\text{If } x_m(k) < 0 \text{ で探索成功}); \\ q_m, & (\text{If } x_m(k) = 0 \text{ で探索成功}); \\ \alpha q_m + (1 - \alpha), & (\text{If } x_m(k) > 0 \text{ で探索成功}). \end{cases} \quad (39)$$

$$0 < \alpha < 1.$$

(平成12年1月7日受付)

(平成12年3月2日採録)



金 春植

1983年、中国大連理工大学電子工程系工業自動化専攻卒業。同年同大学助手、88年同大学講師。95年九州大学大学院工学研究科博士後期課程電気工学専攻修了。同年九州工業大学情報工学部助手。98年九州大学大学院システム情報科学研究科助手、現在に至る。博士(工学)。計測自動制御学会、システム制御情報学会、日本機械学会各会員。

**平澤宏太郎（正会員）**

1966年九州大学大学院工学研究科修士課程修了。同年（株）日立製作所入社日立研究所勤務。89年同研究所副所長。92年同大みか工場主管技師長を経て、92年九州大学工学部教授。96年同大学院システム情報科学研究科教授。現在に至る。工学博士。電気学会、計測自動制御学会、IEEE各会員。

**村田 純一**

1986年九州大学大学院工学研究科博士後期課程電気工学専攻修了。同年、同大学工学部助手。88年同大学工学部助教授。96年同大学システム情報科学研究科助教授。現在に至る。工学博士。電気学会、計測自動制御学会、システム制御情報学会、IEEE各会員。

**胡 敬炉**

1985年中国中山大学大学院修士課程修了。同年、同大学電子工学科助手。88年同講師。93年来日。97年九州工業大学情報工学研究科博士後期課程修了。同年九州大学ベンチャービジネスラボラトリ非常勤研究員を経て、同大学システム情報科学研究科助手。現在に至る。情報工学博士。電気学会、計測自動制御学会各会員。

**松岡 拓哉**

1996年九州大学工学部電気工学科卒業。98年同大学大学院システム情報科学研究科修士課程電気電子システム工学専攻修了。同年トヨタ自動車（株）勤務。現在に至る。