

隠れ変数を有するダイナミカルシステム集合の自己組織化マップ

Self-Organizing Maps of dynamical systems with hidden variables

辻 純一, 大久保 貴之, 古川 徹生

Jun-ichi TSUJI, Takashi OHKUBO, Tetsuo FURUKAWA

九州工業大学 大学院生命体工学研究科

Kyushu Institute of Technology

Abstract: The purpose of this research is to realize a self-organizing map of dynamical systems. In other words, the architecture to identify the dynamical systems using the observed time series data set, and to acquire a self-organizing map based on the degree of similarity between systems is developed. This architecture can be applied to various fields like the orbit generation of robot arms and electroencephalogram analyses, etc. In this research, we tried to realize the self-organizing map of the dynamical systems using mnSOM. However we can not use the conventional mnSOM since its algorithm is not considered the problem of hidden variables which are necessary for identification of dynamical systems. Then, we developed the algorithm in consideration of hidden variables, and verified the effectiveness of the new algorithm.

1 はじめに

Kohonen の自己組織化マップ (Self-Organizing Map: SOM) は高次元ベクトルデータの集合を, データ相互間の位相関係を保持したまま自己組織的に低次元空間へ写像するアーキテクチャである. SOM は様々な分野で応用可能な有用なアルゴリズムであるが, これをベクトル以外のデータをも扱えるように拡張することで, 一層広い範囲での応用が期待できる [1, 2, 3]. その中のひとつとしてダイナミカルシステムへの応用がある. もしそれが可能になれば観測システム間を内挿補間する中間的システムの生成が可能となり, ロボットアームの軌道生成などダイナミカルシステムを扱う様々な分野で応用が期待される.

われわれの提案するモジュラーネットワーク型 SOM (modular network SOM: mnSOM) は, まさにこの枠組みで SOM の一般化を目指すものである [4, 5]. mnSOM は, 従来型 SOM の参照ベクトルユニットをニューラルネットのモジュールに置き換えたものであり, ユーザーはマップしたい対象に応じてモジュールのアーキテクチャを自由にデザインすることができる. この視点に立てば, 従来型 SOM は Hebb ニューロンをモジュールとする mnSOM とみなすことができる. すなわち mnSOM は従来型 SOM も包含する本質的な SOM の一般化とみることができる.

現在の mnSOM に関するわれわれの関心は, 任意のモジュールアーキテクチャ X に対して, その mnSOM の学習アルゴリズムを X に依存しない形で記述できる

かどうかである. 中でも隠れ変数を伴う課題は本質的な難しさを伴っており, 従来型 SOM のナイーブな拡張では解決できない. もしこれが解決できればダイナミカルシステム集合の自己組織化マップも実現可能となる. そこで本稿では, ダイナミカルシステム集合の自己組織化マップを目標に据えつつ, より一般化した状況での隠れ変数問題について取り扱う.

2 問題の枠組み

今, mnSOM で扱いたいシステムの集合 $\mathcal{O} = \{O_1, \dots, O_I\}$ が与えられたとする. また各システムからは観測データ集合 $D_i = \{x_{i,1}, x_{i,2}, \dots\}$ が得られているとする. たとえばマップしたいシステムが I 個の非線形ダイナミカルシステムならば, \mathcal{O} はそれらシステムの集合であり, D_i は i -th システムから観測された入出力時系列に相当する. 目標はこれら I 個の対象を D_i から同定し, さらに互いの類似関係を元に低次元空間上に配置することである. すなわち互いに似た 2 つのシステムがあればマップ上でも近くに, また互いに異なるシステムはマップ上でも遠くに配置されるようにしたい.

このとき SOM のナイーブな拡張を考えれば, 次のようなアルゴリズムが思いつく.

- (i) マップを行いたいシステム $\{O_i\}$ を表現しうるアーキテクチャをモジュールとする mnSOM を作る.
- (ii) データ集合 D_i を一つ選び, それを最もよく表現するモジュールを O_i に対する勝者とし, 近傍関

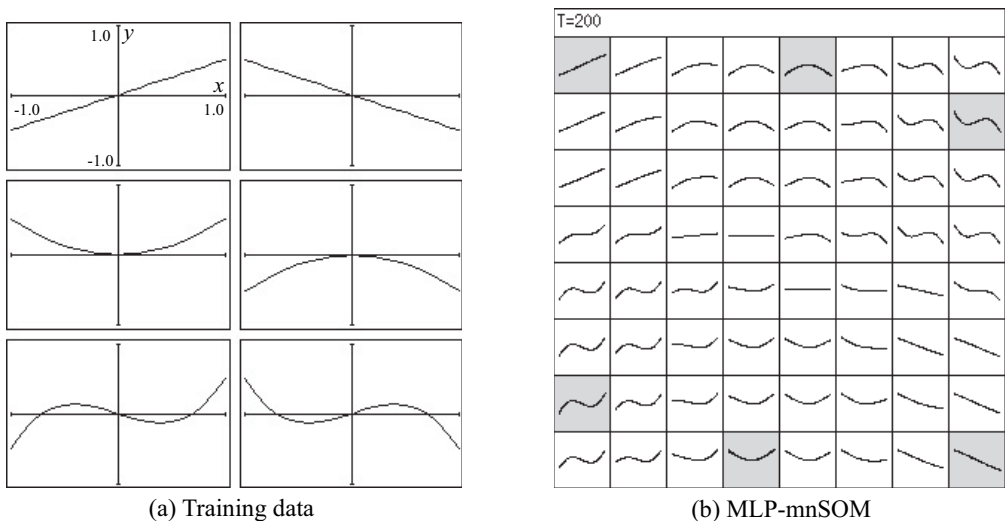


図 1: MLP をモジュールとした mnSOM の学習例 - 1

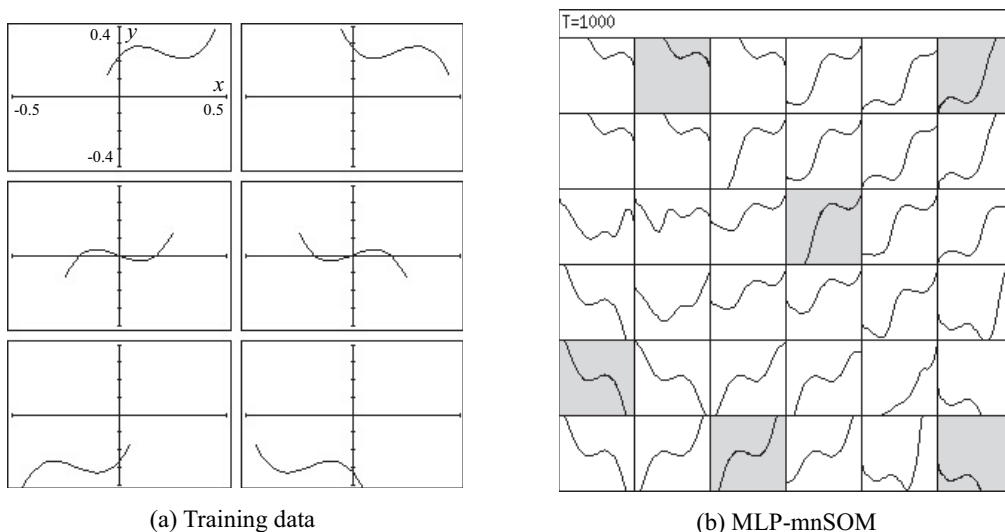


図 2: MLP をモジュールとした mnSOM の学習例 - 2

- 数を用いて各モジュールが学ぶべき学習量を計算する。
- (iii) データ D_i について (ii) で計算した学習量に応じて各モジュールを学習させる。

ダイナミカルシステムの場合であれば、リカレントネットワークをモジュールとする mnSOM (RNN-mnSOM) において、時系列 D_i を最も小さな誤差で再現したりリカレントモジュールを勝者とし、勝者とその近傍のモジュールに対して D_i を学習させるアルゴリズムとなる。

図 1 はこのようなナイーブな mnSOM アルゴリズムを用いて 6 個の非線形関数 $y = f_i(x)$ の自己組織化マップ

を生成した例である。モジュールには多層パーセプトロン (Multi-Layer Perceptron: MLP) を用いた (図 3)。この場合は期待通りのマップが得られる。すなわち似た関数が近くに、反対の特徴を持つ関数が対角に配置され、全体として関数形の連続的な変化を表現する。この状況下においては、SOM アルゴリズムのナイーブな拡張でも期待通りの結果が得られることが理論的に保証されている [5]。ただし関数の独立変数 x の確率密度 $p(x)$ がクラスに対して非依存で、すべてのクラスに対して共通の $p(x)$ で与えられていなければならない。

しかしナイーブなアルゴリズムは図 2 のような場合

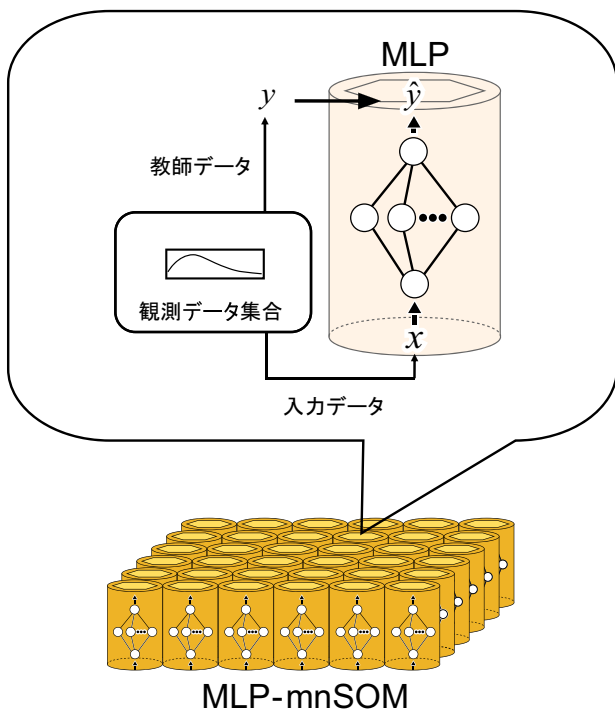


図 3: MLP-mnSOM のアーキテクチャ

において破綻する．ユーザーが期待するマップはおそらく、サイン型の関数形を保ったまま平行移動するようなものであるだろうが、そのような結果は理論的にも期待できない．その最大の理由は入力 x の確率密度がクラスに依存しているためである．言い換えるならば、クラス間で共通の隠れ変数 ξ があり、 $p(\xi)$ は各クラスとも等しいが、 $x = g_i(\xi)$ がクラスによって異なると考えることができる．この $x = g_i(\xi)$ が、ユーザーが暗に期待するシステム（この場合は関数）間の距離測度を決定している．すなわち mnSOM が計算しているアルゴリズム上の距離測度と、システムに内在する（そしてユーザーが期待する）隠れ変数を介した距離測度が食い違っているのである．そのためナイーブなアルゴリズムを使う限りこの問題は解決できない [6] ．

われわれの目的は、これら二つのどちらのケースでも対応できるような SOM の一般化したアルゴリズムを記述し、ダイナミカルシステム集合の自己組織化マップに応用することである．そこで次節ではまず、隠れ変数問題を有する代表的なアーキテクチャである ANN-mnSOM を用いて、隠れ変数問題の検証と学習アルゴリズムの改良を行う．そしてそれらを踏まえ、RNN-mnSOM によるダイナミカルシステム集合の自己組織化マップへの応用についての議論を行う．

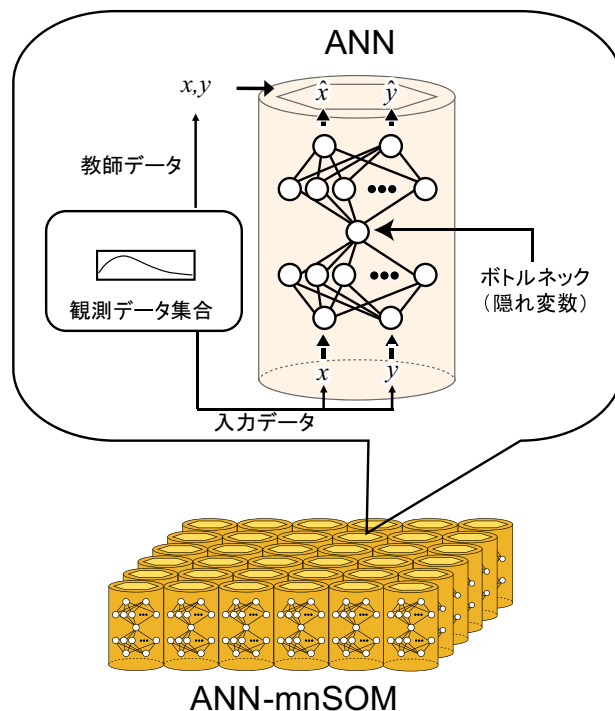


図 4: ANN-mnSOM のアーキテクチャ

3 ANN-mnSOM による検証

本節では自己想起型ニューラルネット (Autoassociative Neural Network: ANN) をモジュールとする mnSOM (ANN-mnSOM) によって図 5 に示す 3 つのデータ集合及び図 7 に示す 6 つのデータ集合のマップを生成した．図 4 に ANN-mnSOM のアーキテクチャを示す．用いた ANN モジュールは 5 層構造を持ち、各層のユニット数は第 1,5 層を 2 個（それぞれデータ点の x 座標, y 座標に対応）、第 2,4 層を 8 個、第 3 層（ボトルネック層）を 1 個とした．

比較する ANN-mnSOM のアルゴリズムは次の 2 つである．1 つは前述したナイーブなアルゴリズムであり、もう 1 つは隠れ変数を考慮したアルゴリズムである．具体的には隠れ変数の推定という要素を取り入れ、隠れ変数の推定とマップの生成を交互に行う EM ライクなアルゴリズムである．本課題にとって隠れ変数はボトルネックユニットの出力値に相当する．そこで隠れ変数を考慮したアルゴリズムは次のように記述される．

- (i) 与えられたデータ集合を最も小さい誤差で表現する ANN モジュールを勝者モジュールとする．
- (ii) 勝者モジュールのボトルネックユニットの値が推定された隠れ変数値とみなす．ただし隠れ変数値

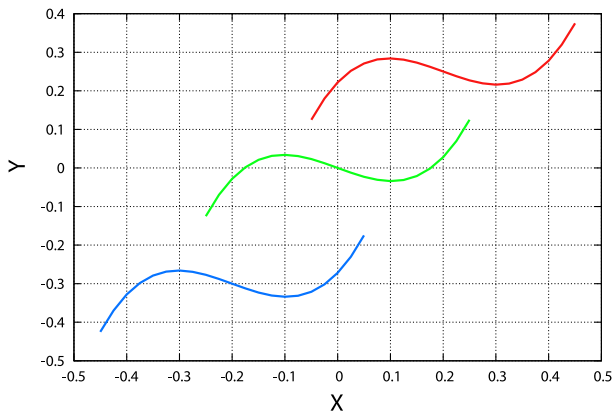


図 5: 学習データ — 1

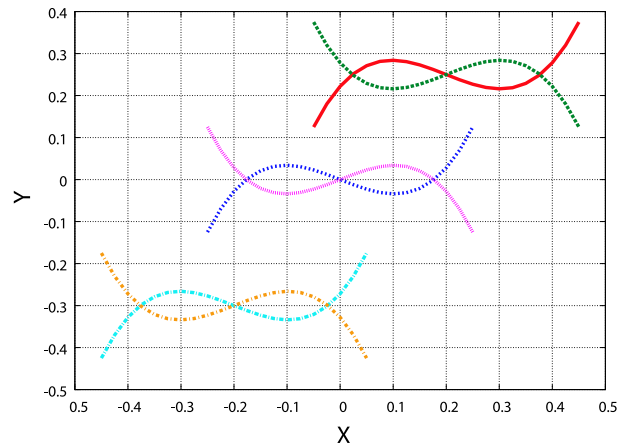
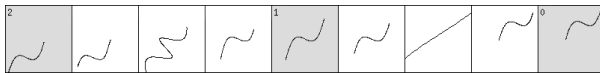
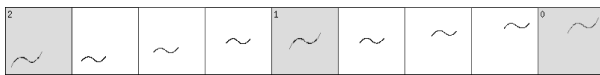


図 7: 学習データ — 2



(a) ナイーブな ANN-mnSOM



(b) 隠れ変数を考慮した ANN-mnSOM

図 6: 2つのアルゴリズムによる計算結果 — 1

が $[-1, +1]$ の全領域を取るように規格化する .

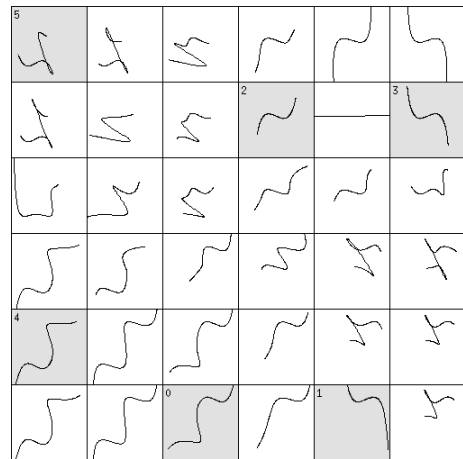
- (iii) 勝者以外のモジュールは, ボトルネックユニットの値を勝者モジュールの値に固定し (クランプし), この状態で誤差逆伝播学習を行う . またこのときの学習量は近傍関数で決定する .

このアルゴリズムの本質は, 隠れ変数の推定を各モジュールに好きに任せるのではなく, 勝者モジュールの推定値を他のモジュールにも適用することで, モジュール間で一貫した隠れ変数推定を行うことにある .

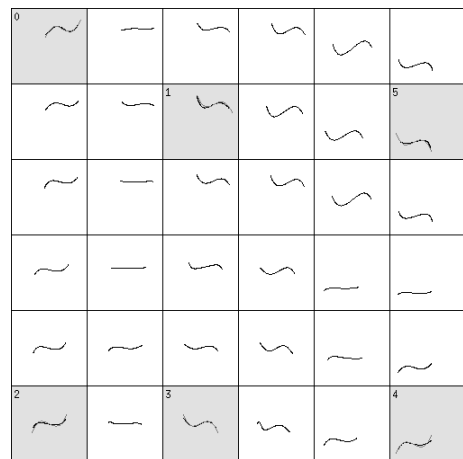
2つのアルゴリズムで計算した結果を図 6, 図 8 に示す . 従来のナイーブなアルゴリズムでは明らかに望ましい結果が得られておらず, 隠れ変数を考慮したアルゴリズムの有効性が確認できた .

4 RNN-mnSOM による検証

本節ではリカレントネット (Recurrent Neural Network: RNN) をモジュールとする mnSOM (RNN-mnSOM) をダイナミカルシステム集合の自己組織化マップに応用する . 図 9 に RNN-mnSOM のアーキテ



(a) ナイーブな ANN-mnSOM



(b) 隠れ変数を考慮した ANN-mnSOM

図 8: 2つのアルゴリズムによる計算結果 — 2

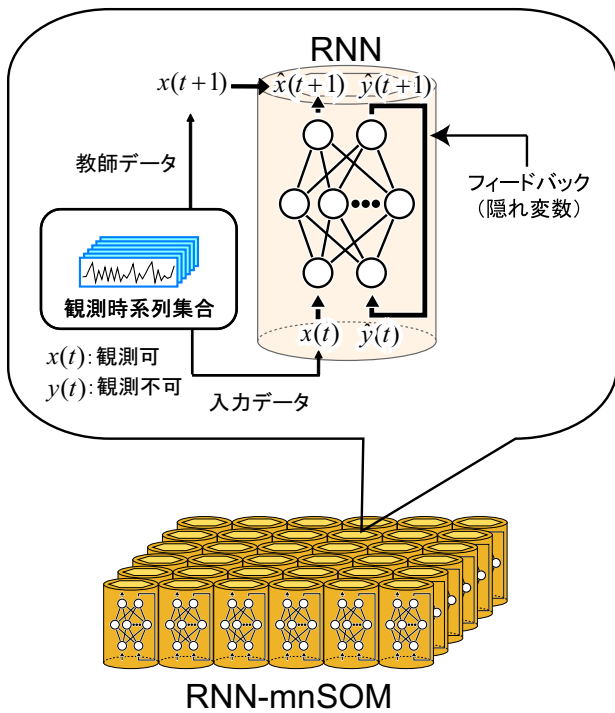


図 9: RNN-mnSOM のアーキテクチャ

クチャを示す。

学習課題として式 (1) に示すエノン写像を用い、パラメータ a の異なる複数のシステムから観測された時系列集合を RNN-mnSOM に与えた。本研究では、 $x(t)$ を観測可能変数、 $y(t)$ を観測不可能な隠れ変数と仮定した。

$$\begin{cases} x(t+1) = 1 - ax(t)^2 + y(t) \\ y(t+1) = bx(t) \end{cases} \quad (1)$$

mnSOM の性能は観測システム数、マップサイズに依存しないため、検証の容易さを考慮し観測システム数を 2 個 ($S_1(a = 0.4, b = 0.3)$, $S_2(a = 0.8, b = 0.3)$)、モジュール数を 3 個 (M^1, M^2, M^3) とした。以上の条件下で、 M^1, M^3 が勝者モジュールとなり各観測システムを同定し、 M^2 が S_1, S_2 の中間的システム ($a = 0.6, b = 0.3$) を生成すれば、一般的な状況においても RNN-mnSOM の性能が保証されたことになる。

RNN-mnSOM においてもナイーブなアルゴリズムと隠れ変数を考慮したアルゴリズムを用いて検証を行った。RNN-mnSOM における隠れ変数はフィードバックユニットの出力値に相当する。

まず、ナイーブなアルゴリズムによる計算結果を図 10 に示す。黒線は各モジュールの出力波形、赤線はエノン写像から観測された各モジュールが出力すべき波

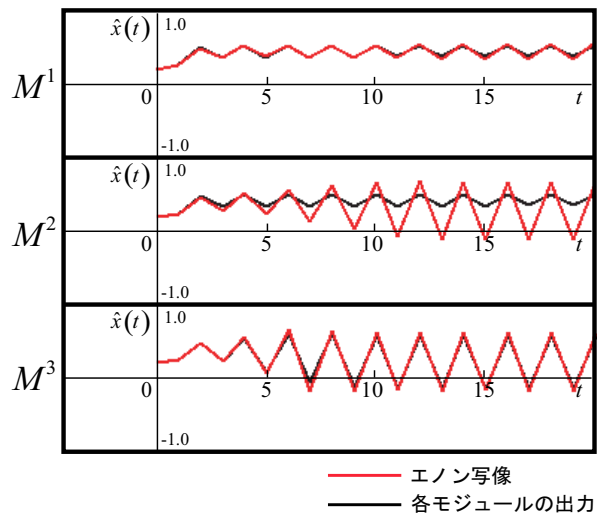


図 10: ナイーブな RNN-mnSOM における各モジュールの出力波形

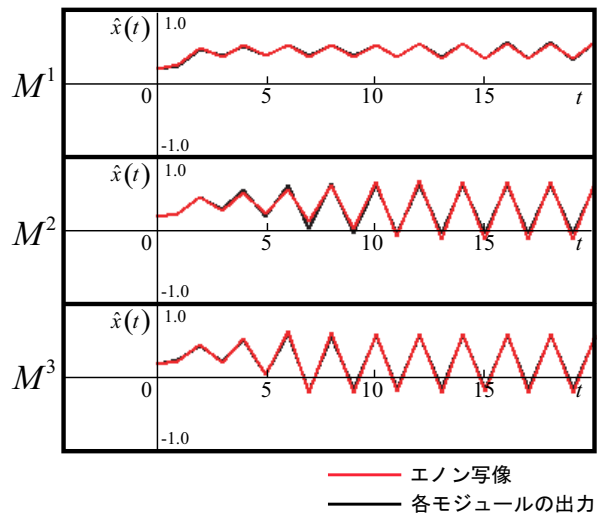


図 11: 隠れ変数を考慮した RNN-mnSOM における各モジュールの出力波形

形である。 M^1 は S_1 、 M^3 は S_2 の勝者モジュールとなり、出力波形がエノン写像から観測された波形と一致した。しかし、 M^2 の出力波形は生成すべき中間的システムの波形と一致しなかった。次に、隠れ変数を考慮したアルゴリズムによる計算結果を図 11 に示す。 M^2 の出力波形が中間的システムの波形と概ね一致しており、ナイーブなアルゴリズムよりも望ましい結果が得られているのがわかる。

5 まとめ

SOMの一般化において、最も重要な点は2つのマップ対象間に距離測度をどのように定義するかということである。2つの対象間の対応を取るのが隠れ変数の役目であり、隠れ変数推定問題はSOMの一般化において本質的な問題である。従ってmnSOMを応用する際は、何が距離測度なのかを考えた上で正しいアルゴリズムを導出する必要がある。本研究により隠れ変数を考慮したアルゴリズムの有効性が示され、RNN-mnSOMを用いたダイナミカルシステム集合の自己組織化マップを得ることができた。現在はRNN-mnSOMの実課題への応用としてロボットアームの軌道生成や適応制御におけるシステム同定などの課題に取り組んでいる。

謝辞 本研究の一部は九州工業大学 21 世紀 COE プログラムおよび科研費基盤 (C) (課題番号 17500193) の支援を受けて行われた。

参考文献

- [1] T. Furukawa and K. Tokunaga: “A new development of self-organizing maps realized through a marriage with modular-networks”, Proceedings of the 2007 IEEE Symposium on Foundations of Computational Intelligence, pp. 637–644, 2007
- [2] T. Furukawa, K. Tokunaga, S. Kaneko, K. Kimotsuki and S. Yasui: “Generalized self-organizing maps(mnSOM) for dealing with dynamical systems”, Proceedings of International Symposium on Nonlinear Theory and its Applications, pp. 231–234, 2004
- [3] T. Minatohara, T. Furukawa: “Self-Organizing Adaptive Controllers: Application to the Inverted Pendulum”, Proceedings of the 5th Workshop on Self-Organizing Maps, pp. 41–48, 2005
- [4] K. Tokunaga, T. Furukawa and S. Yasui: “Modular network SOM: Self-organizing maps in function space”, Neural Information Processing – Letters and Reviews, **9**, pp. 15–22, 2005
- [5] T. Furukawa, K. Tokunaga, K. Morishita, S. Yasui: “Modular Network SOM(mnSOM): From Vector Space to Function Space”, Proceedings of International Joint Conference on Neural Networks(IJCNN), pp. 1581–1586, 2005
- [6] 大久保貴之, 古川徹生: “モジュラーネットワーク SOMの一般化: 隠れ変数を考慮する事の必要性”, 日本神経回路学会第16回全国大会講演論文集, pp. 34–35, 2006

連絡先

辻 純一

E-mail: tsuji-junichi@edu.brain.kyutech.ac.jp

大久保 貴之

E-mail: ohkubo-takashi@edu.brain.kyutech.ac.jp

古川 徹生

E-mail: furukawa@brain.kyutech.ac.jp