

SOM²によるトポロジカルな環境地図の獲得

Acquirement of the Topological Environmental Map by SOM²

増岡 孝紀, 徳永 憲洋, 古川 徹生

Takanori MASUOKA, Kazuhiro TOKUNAGA, Tetsuo FURUKAWA

九州工業大学 大学院生命体工学研究科

Kyushu Institute of Technology

Abstract: A robot has to autonomously acquire the environmental map by interaction with its environment to perform the self-localization and the behavior planning in unknown environment. The methods for acquiring the environmental map using the topology preserving neural network, such as a Self-Organizing Map (SOM), a Neural Gas (NG) and so on, have been proposed in the research topic for acquiring the autonomously environmental map. In these topology preserving neural networks, however, it is difficult to estimate the coordinates and direction of robot from information with multi-dimensional vector. In this presentation, we propose the acquiring the environmental map using SOM², in which each neuron unit is extended to an SOM network module. Moreover the algorithm of the proposed method and the experimental results are shown.

1 はじめに

未知の環境において移動ロボットを自律行動させる場合、移動ロボット自身が自己位置推定および障害物の認識、目的地までの行動計画を行う必要がある。これらのタスクをこなすためには、移動ロボット自身が有するセンサ情報を用いて、外界との相互作用を繰り返すことで作業空間の地図（環境地図）を自己組織的に獲得しなければならない。

環境地図と一口に言ってもその表現形態はさまざまであり、今までに移動ロボットのための環境地図獲得手法は数多く提案されている。表現形態を大別するとメトリック地図（定量幾何学的表現）とトポロジカル地図（位相幾何学的表現）の2種類に分けられる。

メトリック地図とは作業空間内の障害物の位置などを正確に表現した地図のことである。メトリック地図は視覚的に確認しやすいという利点があるが、実現するためには移動ロボット自身が自己位置や方向、障害物との距離を正確に観測できる必要があり、そのためには精度の高いセンサが多数必要である。また、複雑な環境では地図の獲得が困難であると考えられる。

トポロジカル地図とはグラフ構造により環境を抽象化して表現した地図のことである。グラフのノードは環境内の特定の場所を表し、リンクはノード間の行動方法を表す。メトリック地図が定量的なセンサ情報や物理モデルを必要とするのに対して、トポロジカル地図では定性的な情報だけを用いるのでコスト面でも軽減し、複雑な環境にも対応できると考えられる。

以上のことから、未知の環境で移動ロボットを自律的に行動させる場合、環境地図の表現形態はトポロジカル地図が望ましいと考えられる。

このような移動ロボットの自律的環境地図獲得に関する研究課題において、Kohonenの提案した自己組織化マップ (Self-Organizing Map: SOM)[1] や Martinezの提案したニューラルガス (Neural Gas: NG)[2] などのトポロジー保存型ニューラルネットワークを用いて環境をグラフ構造で抽象化表現した地図（トポロジカル地図）を獲得する手法が提案されている [3]。しかし従来のトポロジー保存型ニューラルネットワークでは、ニューロンは環境情報を多次元ベクトルでしか表現できないため、センサ情報だけからロボットの座標および方向を特定することは困難である。

一方、これまでに我々はトポロジー保存型ニューラルネットワークの機能を拡張した SOM²(SOM×SOM) を提案し、理論とアルゴリズムを確立してきた [4, 5]。SOM²とは SOMの参照ベクトルユニットを SOMモジュールに置き換えたものである。このとき各 SOMモジュールのことを子 SOMと呼び、SOM²全体のことを親 SOMと呼ぶ。SOM²全体では (親 SOM)×(子 SOM)の積多様体を表現できる。従って SOM²では多様体の集合でデータを表現できる。

移動ロボットが全方位カメラから得られる情報を用いて環境地図を獲得することを考える。移動ロボットは環境内を動き回り、各地点で全方位カメラの情報を得る。このとき、全方位カメラから得られた情報は多

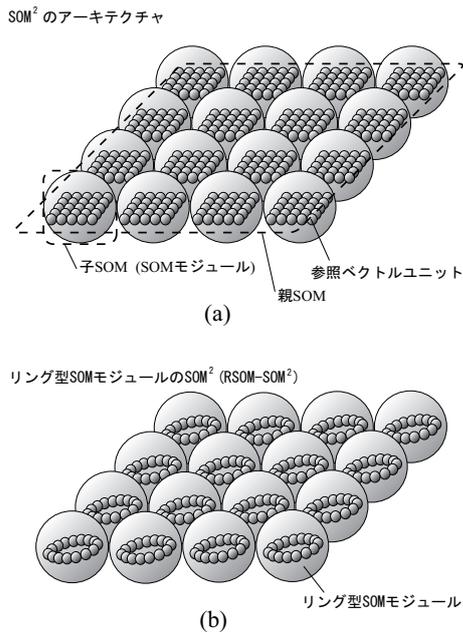


図 1: (a)SOM² のアーキテクチャ (b)RSOM×SOM のアーキテクチャ

次元ベクトル空間上で、リング状の多様体を形成する。SOM² では子 SOM が多様体を近似するので、各々の子 SOM が環境内の各地点を表現することができる。そして親 SOM は実際の環境の位置関係をトポロジを保存したまま写像する。つまり、親 SOM が表現するマップ自体が環境地図となる。

全方位カメラからの情報はリング状の多様体を形成するので、本稿では SOM² の中でも特に、子 SOM を 1 次元のリング型 SOM (Ring SOM : RSOM) とし、親 SOM を 2 次元 SOM としたリング型 SOM モジュール SOM² (RSOM×SOM) を用いる。RSOM とは SOM の参照ベクトルユニットを格子状ではなくリング状に配置したものである。

そこで本稿では RSOM×SOM を用いた自律的環境地図獲得手法を提案し、さらに提案手法のアルゴリズムとシミュレーション結果について報告する。

2 RSOM×SOM による環境地図の獲得

本節ではまず SOM² のアーキテクチャについて述べる。次に環境地図獲得において RSOM×SOM のアーキテクチャと特徴を挙げ、最後に RSOM×SOM のアルゴリズムを説明する。

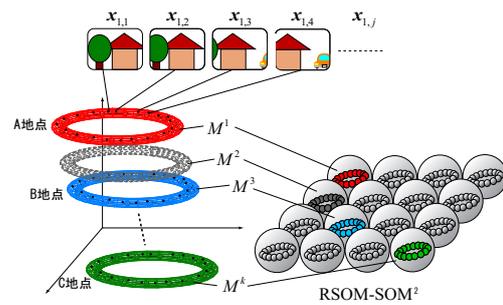


図 2: RSOM×SOM におけるデータと参照ベクトル集合の関係

2.1 SOM² のアーキテクチャ

SOM² は SOM の学習方法を基にしたアルゴリズムである。SOM はオンライン型学習が一般的であるが、SOM² ではバッチ型 SOM の学習方法を基にしている。

SOM² は SOM の参照ベクトルユニットを SOM モジュールに置き換えたものである。図 1(a) に SOM² のアーキテクチャを示す。通常の SOM は参照ベクトルが格子状に配置されているのに対して、SOM² では SOM モジュールが格子状に配置されている。既に述べたとおり、本稿ではこの SOM モジュールのことを子 SOM と呼び、子 SOM で構成される SOM のことを親 SOM と呼ぶ。通常の SOM はベクトル集合を多様体として近似しトポロジを保存したまま特徴マップ上に表現する。一方、SOM² では各々の子 SOM が表現する多様体を、多様体集合として近似しトポロジを保存したまま親 SOM の特徴マップ上に表現する。

2.2 RSOM×SOM のアーキテクチャと特徴

図 1(b) に RSOM×SOM のアーキテクチャを示す。RSOM×SOM では子 SOM がリング型 SOM モジュールとなっている。

図 2 に RSOM×SOM におけるデータと参照ベクトル集合の関係を示す。全方位カメラからの情報より環境地図を獲得する場合、RSOM×SOM では子 SOM である RSOM モジュールが環境内の A 地点を学習し多様体で近似する。このとき RSOM の各参照ベクトルユニットは、その地点におけるそれぞれの方向の視覚情報を学習する。環境内の A, B, C 地点で視覚情報を取得し学習した場合、各子 SOM が近似した多様体の高次元ベクトル空間上での位置関係と、実際の環境の各地点の位置関係は一致する。多様体の位置関係のトポロジを保存したまま親 SOM のマップに写像することで環境地図は獲得できる。

このように RSOM×SOM では各々の子 SOM の順序付けが成されており、また中間的な環境を学習により獲得する特徴があるため、環境地図獲得において優れたアーキテクチャだと言える。

2.3 RSOM×SOM のアルゴリズム

RSOM のアルゴリズムはバッチ型 SOM とほとんど変わらない。唯一違う点は参照ベクトルユニット間のマップ上での距離の取り方である。以下に RSOM のアルゴリズムと RSOM×SOM のアルゴリズムを説明する。

RSOM

RSOM は評価プロセス、競合プロセス、協調プロセス、適応プロセスという 4 つのプロセスを繰り返すことで学習を行う。

(i) 評価プロセス

全学習データ x_i ($i = 1, 2, \dots, I$) と各ユニットの参照ベクトル w^k ($k = 1, 2, \dots, K$) との距離 E_i^k をそれぞれ計算する。

$$E_i^k = \|w^k - x_i\|^2 \quad \forall i, \forall k \quad (1)$$

(ii) 競合プロセス

全ユニットの中で、(1) で求めた距離 E_i^k の値が最も小さいユニットを勝者ユニット (Best Matching Unit: BMU) k_i^* に決定する。これも全学習データに対して行う。

$$k_i^* = \arg \min_k E_i^k \quad \forall i \quad (2)$$

(iii) 協調プロセス

協調プロセスでは各ユニットの学習データに対する学習率 β_i^k を式 (3) で求める。式 (3) において ϕ_i^k は近傍関数であり、各ユニット k と BMU k_i^* 間のマップ上での距離 $d(k, k_i^*)$ を用いて式 (4) で計算される。この距離の求め方が従来の格子状に配置された SOM と RSOM では異なる。従来の SOM の場合、ユニット間の距離の計算には直交座標を用いて計算する。これに対して 1 次元の RSOM の場合は、ユニットがリング状に配置されているため距離の計算は極座標上で求める。 $\sigma(t)$ は単調減少関数であり式 (5) で与えられる。また σ_{min} は最小近傍半径、 σ_{max} は最大近傍半径、 τ_N は近傍半径の時定数を表す。

$$\beta_i^k = \frac{\phi_i^k}{\sum_{i'} \phi_{i'}^k} \quad (3)$$

$$\phi_i^k = \exp\left(-\frac{d(k, k_i^*)^2}{2\sigma^2(t)}\right) \quad (4)$$

$$\sigma(t) = \sigma_{min} + (\sigma_{max} - \sigma_{min}) \exp\left(-\frac{t}{\tau_N}\right) \quad (5)$$

(iv) 適応プロセス

(3) で求めた学習率 β_i^k を用いて、各ユニットの参照ベクトル w^k を更新する。

$$w^k(t+1) = \sum_i \beta_i^k x_i \quad (6)$$

式 (6) は各データの学習率の合計が 1 になるように正規化している。以上の 4 つのプロセスを行うことを 1 回の学習とする。学習を繰り返すことで特徴マップを生成する。

RSOM×SOM

RSOM×SOM は RSOM と同様、評価、競合、協調、適応プロセスを繰り返すことで実現できる。ここでは RSOM×SOM の構成を次のように考える。 J 個の多次元ベクトルデータで 1 つのエピソード (多様体) を構成する学習データが I 個ある。その際 i 番目のエピソードの学習データは $D_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,J}\}$ と書くことができる。また子 SOM は L 個の参照ベクトルユニットを持ち、親 SOM は K 個の子 SOM を持つ。その際、参照ベクトルは $W^k = \{w^{k,1}, w^{k,2}, \dots, w^{k,L}\}$ と書くことができる。RSOM×SOM の学習アルゴリズムをプロセス毎に以下に示す。

(i) 評価・競合プロセス

まず学習データベクトル $x_{i,j}$ と各子 SOM 内の全参照ベクトル $w^{k,l}$ との誤差 $e_{i,j}^{k,l}$ を求める。このとき $e_{i,j}^{k,l}$ が最も小さくなるユニットが $x_{i,j}$ に対する勝者ユニット (BMU) となる。BMU の番号を $l_{i,j}^{k,*}$ とすれば BMU の番号は式 (8) で表せられる。

$$e_{i,j}^{k,l} = \|w^{k,l} - x_{i,j}\|^2 \quad (7)$$

$$l_{i,j}^{k,*} = \arg \min_l e_{i,j}^{k,l} \quad (8)$$

学習データと BMU が持つ参照ベクトルとの距離は $e_{i,j}^{k,*}$ と表すことができる。式 (9) に示すように、1 エピソード分のデータセットに対する誤差 $e_{i,j}^{k,*}$ の平均が各子 SOM の誤差となる。

$$E_i^k = \frac{1}{J} \sum_{j=1}^J e_{i,j}^{k,*} \quad (9)$$

各モジュールの誤差を見て値が1番小さいモジュールが、 i 番目のエピソードに対する勝者モジュール (Best Matching Module : BMM) となる。

$$k_i^* = \arg \min_k E_i^k \quad (10)$$

(ii) 協調プロセス

協調プロセスでは参照ベクトルを更新する際に用いる学習率 α_i^k , $\beta_{i,j}^l$ を求める。親 SOM の場合は BMM と各モジュールの距離より近傍関数 ϕ_i^k が決定し、学習率 α_i^k が求まる。

$$\alpha_i^k = \frac{\phi_i^k}{\sum_{i'} \phi_{i'}^k} \quad (11)$$

$$\phi_i^k = \exp\left(-\frac{d(k, k_i^*)^2}{2\sigma^2(t)}\right) \quad (12)$$

$$\sigma(t) = \sigma_{min} + (\sigma_{max} - \sigma_{min}) \exp\left(-\frac{t}{\tau_p}\right) \quad (13)$$

子 SOM の場合は RSOM なので BMU と各ユニットとの距離を極座標 θ を用いて求め、近傍関数 $\phi_{i,j}^l$ および学習率 $\beta_{i,j}^l$ を計算する。

$$\beta_{i,j}^l = \frac{\phi_{i,j}^l}{\sum_{j'} \phi_{i,j'}^l} \quad (14)$$

$$\phi_{i,j}^l = \exp\left(-\frac{d(l, l_{i,j}^{*,*})^2}{2\sigma^2(t)}\right) \quad (15)$$

$$\sigma(t) = \sigma_{min} + (\sigma_{max} - \sigma_{min}) \exp\left(-\frac{t}{\tau_c}\right) \quad (16)$$

(iii) 適応プロセス

(2) で求めた学習率 α_i^k , $\beta_{i,j}^l$ より参照ベクトル $w^{k,l}$ が更新される。

$$w^{k,l} = \sum_{i=1}^I \sum_{j=1}^J \alpha_i^k \beta_{i,j}^l x_{i,j} \quad (17)$$

この計算プロセスを繰り返すことで RSOM×SOM は学習を行う。

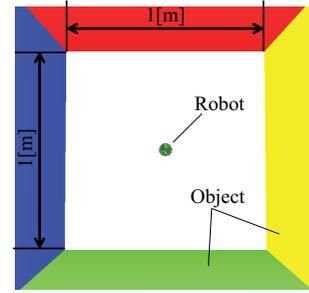


図 3: 本シミュレーションにおける移動ロボットの作業空間 (Webots で作成)

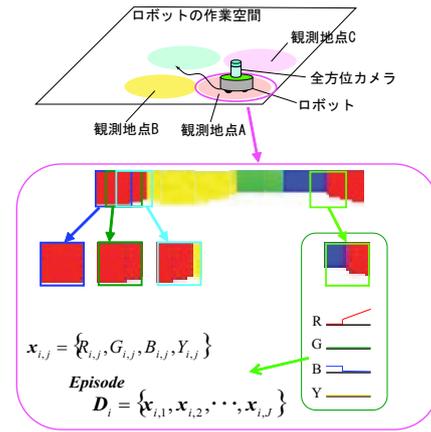


図 4: データ取得方法

3 シミュレーション

提案手法が期待どおりの動作をするかシミュレーションにより検証した。3.1 節では本シミュレーションの枠組みについて述べる。3.2, 3 節ではシミュレーション結果を示す。まず 3.2 節では RSOM×SOM でトポロジカルな地図を獲得できるか確認するため、移動ロボットを定期的に動かすことで学習データを獲得し検証した。3.3 節では移動ロボットを不規則に動かしても、トポロジカルな地図が獲得できるか検証した。

3.1 本シミュレーションでの枠組み

四方が壁に囲まれた正方形の環境に移動ロボットがいる状況を考える (図 3)。このとき四方の壁にはそれぞれ異なる色 (赤, 青, 緑, 黄) が塗られているものとする。移動ロボットには全方位カメラが取り付けられており周囲の視覚情報を獲得することができる。環境内のある任意の地点において移動ロボットは周囲のパノラマ画像を獲得する。パノラマ画像より、ロボットの向いている角度ごとに画像を切り出し、その画像

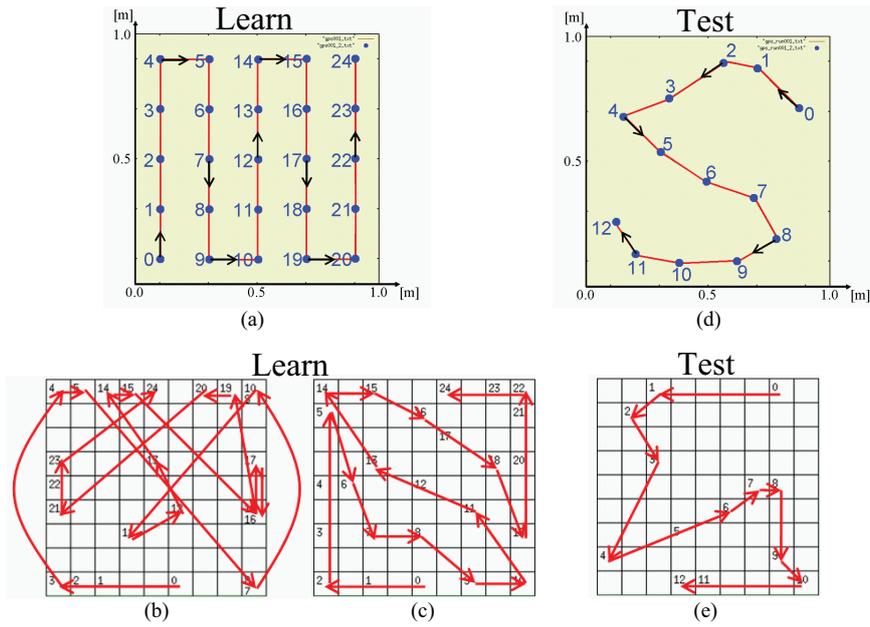


図 5: (a) 学習データを取得した際の移動ロボットの軌跡 (b) バッチ型 SOM での学習結果 (c) RSOM×SOM での学習結果 (d) テストデータを取得した際の移動ロボットの軌跡 (e) テストデータの RSOM×SOM での検証結果

を左から右へと 1 列ずつ見ていき，画像の縦の画素数の内，壁の色が占める割合を調べる．そして学習データベクトルが $x_{i,j} = \{R_{i,j}, G_{i,j}, B_{i,j}, Y_{i,j}\}$ のように獲得できる (図 4 参照)．このとき R, G, B, Y は 0 から 1 までの値である．1 つの地点で切り出した画像の数だけデータベクトルを獲得するので，1 つの地点でのデータセットは $D_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,J}\}$ となり，多様体を形成する．つまり，ロボットが環境内を移動し複数地点から観測されたデータセットは地点毎に多様体を形成する．RSOM×SOM は多様体の集合をトポロジを保存したままマップ上に表現するので，学習後のマップは実際の環境の位置関係を保持したまま表現できると期待される．

本シミュレーションではロボットシミュレータ Webots を用い，上記環境の作成およびシミュレーションを行った．図 3 は Webots を用いて作成した環境である．

3.2 規則的行動からの地図獲得

学習データは図 5(a) に示すように移動ロボットを図 3 の作業空間内で動かし，青い点でデータを獲得した．矢印をロボットの向いている方向，赤い線はロボットの軌跡，番号はデータを取った順番を示す．SOM² に学習させるデータの取得方法は 3.1 節で示したとおりである．取得した学習データを用いて，バッチ型 SOM および RSOM×SOM どちらにおいても学習を行わせ

表 2: RSOM×SOM

表 1: バッチ型 SOM

学習データ	25
データの次元	512
マップの大きさ	9 × 9
時定数 τ_N	30
σ_{max}	13.5
σ_{min}	1.0

エピソード数	25
データ数	20
データの次元	64
親マップ	9 × 9
親 σ_{max}	13.5
親 σ_{min}	1.0
時定数 τ_p	20
子マップ	21 × 1
子 σ_{max}	π
子 σ_{min}	0.065
時定数 τ_c	30

た．学習に用いたバッチ型 SOM のパラメータを表 1，RSOM×SOM のパラメータを表 2 に示す．

図 5(b), (c) が学習後のマップである．バッチ型 SOM の結果 (b) ではマップが崩れて，センサ情報から環境の位置を特定できるマップの獲得ができなかった．RSOM×SOM ではトポロジを保存したマップを獲得することができ，環境の位置を特定することができる．この点から従来の SOM よりも RSOM×SOM の方がより頑健に環境地図を獲得できていることが分かる．

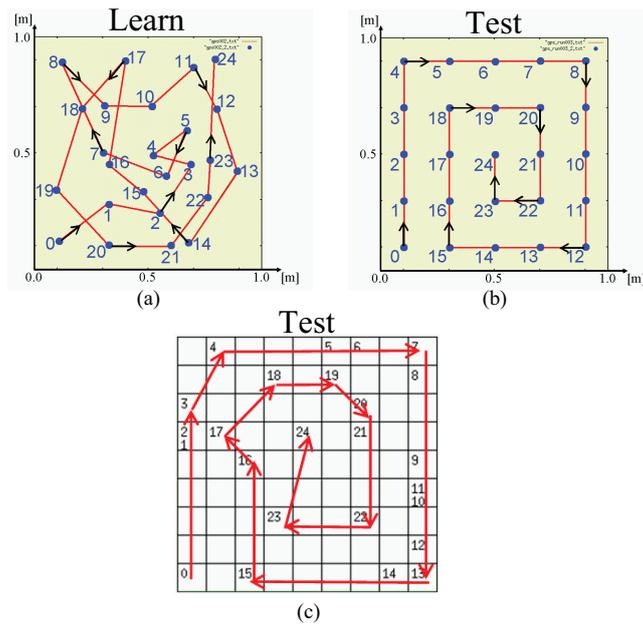


図 6: (a) 学習データを取得した際の移動ロボットの軌跡 (b) テストデータを取得した際の移動ロボットの軌跡 (c) テストデータの RSOM×SOM での検証結果

次に学習で得られた RSOM×SOM のマップを用いて、図 5(d) のようにロボットを動かして、環境とマップ上での位置の対応が取れているかテストを行った。その結果が図 5(e) である。これより学習に用いたデータ以外のテストデータでも環境のトポロジーが保存されていることが確認された。

3.3 不規則な行動からの地図獲得

次に移動ロボットを不規則に動かした場合、期待したマップが得られるかを検証した。本シミュレーションでは、まず図 6(a) のようにロボットを動かして学習データを観測させた後に RSOM×SOM で学習を行う。さらに期待したマップが得られているかを示すために、図 6(b) のようにロボットを動かしてテストを行った。テストの結果を図 6(c) に示す。結果より、不規則にデータを取得しても、環境のトポロジーを保存したマップが得られていることが確認できた。また、このシミュレーションにおいても 3.2 節同様バッチ型 SOM による検証を行ったが望むようなマップは得られなかった。

4 おわりに

移動ロボットが未知の作業空間を自律的に行動するためには、その環境の情報を持つ環境地図が必要である。本稿では RSOM×SOM を用いることでトポロジカルな地図を獲得する手法を提案し、計算機シミュレ

ションによりその有効性を示した。

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. Kohonen, *Self Organizing Maps*, Springer Verlag, 1995
- [2] Thomas M. Martinez, Stanislav G. Berkovich, Klaus J. Schulten, ““Neural-Gas” Network for Vector Quantization and its Application to Time-Series Prediction”, *IEEE TRANSACTIONS ON NEURAL NETWORKS*, 4:558-567, 1993
- [3] 田中敏雄, 西田健次, 栗田多喜男, “場所細胞の位置マップと強化学習を用いた移動ロボットのナビゲーション”, *電子情報通信学会論文誌*, Vol. J88-D- , No.9, pp.1866-1875, 2005
- [4] T. Furukawa, “SOM2 as ‘SOM of SOMs,’” *Proc. of WSOM05*, 41-48, 2005
- [5] T. Furukawa, “SOM of SOMs: An Extension of SOM from ‘Map’ to ‘Homotopy,’” *Proc. of ICONIP2006*, 2006 (*to be appeared*)

連絡先

古川 徹生 (E-mail: furukawa@brain.kyutech.ac.jp)