

# 自己進化型モジュラーネットワークによる環境地図の自己組織的獲得

## The self-organized mapping of environment using Self Evolving Modular Network

川畑 宣之 (PY), 徳永 憲洋, 古川 徹生

Nobuyuki Kawabata(PY), Kazuhiro Tokunaga, and Tetuo Furukawa

九州工業大学大学院生命体工学研究科

kawabata-nobuyuki@edu.brain.kyutech.ac.jp

**Abstract**—The topological map is commonly-used method for build the internal model of the environment in autonomous robotics. We proposed a self-organized topological map building technique using evolvable modular network which called Self Evolving Modular Network (SEEM). In this paper, we introduce the SEEM algorithm and then use to build the internal model.

**Keywords**— map building, self-organizing maps, modular networks, on-line learning, mobile robot

### 1 はじめに

環境から得たセンサ情報のみにより自律行動を行うロボットを考える場合、環境の地図(内部モデル)をセンサ入力から構築する能力は非常に重要なものとなる。このような内部モデル獲得については多くの手法が提案されているが、なかでも記憶量や計算量に優れた方法としてグラフ状の構造を用いるものがある。この場合、ノードは環境の局所モデル、パスはノード間の経路や行動パターンとすることでグラフ状の状態遷移モデルとして環境全体の内部モデルを表現することができる。

さらに自律という点を重視すると、このグラフはあらかじめ設計者によって定義されたランドマークなどの関係を示したものではなく、ロボットが自身のセンサによって自律的にノードを作りパスを張ったものであることが重要である。

そこで我々は自己組織化に着目した汎用的内部モデル獲得アルゴリズムとして、自己進化型モジュラーネットワーク(Self Evolving Modular network: SEEM)を提案する。SEEMはグラフ構造の競合モジュラーネットワークであり、それぞれのノードがニューラルモジュールとなる。これらのノードは競合学習と強調学習の学習則により自己組織的に局所モデルを獲得し、新規性の高い入力があった場合は自動的に追加される。ノード間にはモジュールの内部表現の類似性と勝者モジュールの遷移を表現する二種類のパスが張られる。これらのパスは強度を持っており、ヘブ学習によって重要なパスのみが

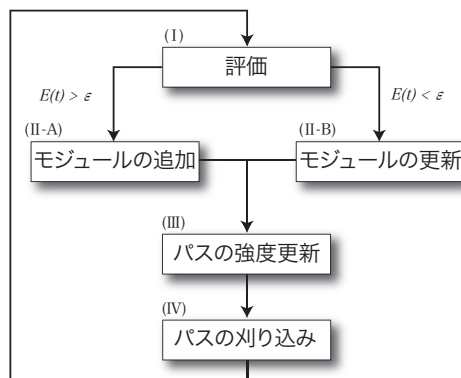


図 1: SEEM アルゴリズムの流れ

残ることになる。さらに SEEM はオンライン学習であり長期間学習を継続しても学習結果が破綻しないようなアルゴリズムになっている。SEEM をロボットの環境内部モデル獲得課題に応用することでグラフ構造の環境内部モデルを自己組織的かつオンラインで獲得可能になることが期待できる。

今回は SEEM のアルゴリズムについての説明とロボットの地図獲得に応用した結果を報告する。

### 2 自己進化型モジュラーネットワーク SEEM

本手法で用いている SEEM はニューラルモジュールをノードとし、ノード間を機能の類似性によって重み付きのパスで結合したグラフ構造のネットワークである。モジュールはネットワークへの入力に対応するモジュールが存在しない場合に自動的に追加され、モジュール間のパスもネットワークへの入力から自己組織的に構成される。SEEM のモジュールには様々なものを用いることが可能で、例えば multilayer perceptron (MLP) やリカレント型のニューラルネットワーク等をモジュールとすることが考えられる。この場合現在のセンサ情報とモータ情報を入力し、次の時刻の行動を予測して出力するようなモデルの自己組織的獲得に応用することができる。今回は基本的な形式の地図として、センサ情報を量子化しながら記憶することが可能な Kohonen の自己組織化マップ(Self-Organizing Map: SOM) をモジュールとした SOM-SEEM についてアルゴリズムを説明する。

### 3 SEEMのアルゴリズム

$K$  個の SOM モジュールを持つ SOM-SEEM があるとする。それぞれのモジュールは  $L$  個のユニットから成る SOM であり、 $k$ -th モジュールの  $l$ -th ユニットの参照ベクトルは  $w^{k,l}$  とする。また、 $k$ -th モジュールと  $k'$ -th モジュールを結ぶ類似度に基づくパスの強度は  $s_s^{k,k'}$ 、勝者モジュールの遷移に基づくパスの強度は  $s_a^{k,k'}$  である。ただし、学習開始時点では  $K = 0$  である。SOM-SEEM に入力されるデータはベクトル集合  $D(t) = \{x_1, x_2, \dots, x_J\}$  であり、この  $D(t)$  をエピソードと呼ぶ。SOM-SEEM は  $D(t)$  が入力される度に図 1 の流れで学習が行われる。

エピソード  $D(t)$  が入力されると以下の式により (I) の評価が行われる。 $E^k(t)$  はエピソードと各 SOM モジュールのモデル誤差であり、 $E^k(t)$  が小さい順に第 1、第 2 勝者  $k^*, k^{*2}$  とする。

$$E^k(t) = \sum_{j=1}^J e_j^{k,l^*} \quad (1)$$

$$l^* = \arg \min_l e_j^{k,l} \quad (2)$$

$$e_j^{k,l} = \|w^{k,l} - x_j\| \quad (3)$$

ここですべての SOM モジュールに対して  $E^k(t) > \varepsilon$  であれば (II-A) の追加、そうでなければ (II-B) の更新が行われる。

(II-A) では新しく作成した SOM モジュールに  $D(t)$  を学習データとして与え、 $D(t)$  に対するモデル誤差がなるべく小さくなるように学習を行う。この際の学習アルゴリズムは一般的な SOM のバッチ学習である。SOM の学習アルゴリズムについては省略する。

(II-B) では以下の式によって SOM モジュールの更新を行う。

$$w^{k,l} = w^{k,l} + \Delta w^{k,l} \quad (4)$$

$$\Delta w^{k,l} = \gamma h^k \sum_j \phi_j^{k,l} (x_j - w^{k,l}) \quad (5)$$

$$h^k = \begin{cases} \frac{a^k(x_j)}{\sum_{k'} a^{k'}(x_j)} & \text{if } k' \in \Omega(b) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

ここで  $\phi_j^{k,l}$  は SOM モジュールの近傍関数、 $\gamma$  は SEEM の学習率であり  $0 < \gamma \leq 1$  である。また、 $a^k(E(t))$  は  $k$ -th モジュールの  $E(t)$  に対する活性値を表し、以下の式で定義される。

$$a^k(E(t)) = e^{-2(E^k(t)/\varepsilon)^2} \quad (7)$$

また、 $\Omega(b)$  は勝者モジュールに結合しているモジュールを示しており、

$$\Omega(i, t) = \{j | s_s^{i,j} > 0\} \quad (8)$$

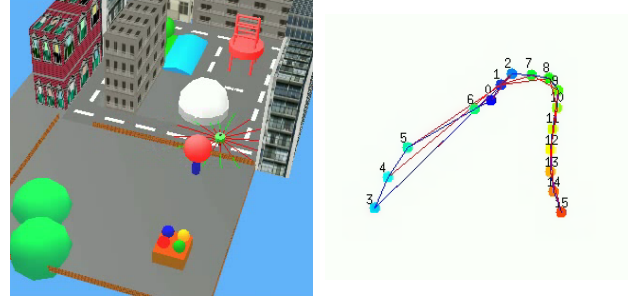


図 2: SOM-SEEM による内部モデル獲得

である。

(III) のパスの強度更新は次の式で行う。

$$s_s^{k,k'}(t) = (1 - \beta)a^k a^{k'} + \beta s_s^{k,k'}(t-1) \quad (9)$$

$$\omega(i, t) = \begin{cases} 1 & i = k^*(t) \\ 0 & i \neq k^*(t) \end{cases} \quad (10)$$

$$s_a^{k,k'}(t) = (1 - \beta)\omega(k, t-1)\omega(k', t) + \beta s_a^{k,k'}(t-1) \quad (11)$$

$\beta$  は  $0 < \beta < 1.0$  の忘却定数であり、通常 0.8 程度に設定しておく。ただし、(II-A) でモジュールの追加が行われた場合は新規モジュールからの勝者、第 2 勝者に繋がるパス強度だけを更新する。この場合、式 (9) の右辺の  $s_s^{k,k'}$  は 0 とする。

次に行うパスの刈り込み (IV) は、全てのパスのうちもっとも強度の弱いものを削除するという方法で行う。(IV) は毎回行うのではなく数ステップに一度だけ行えばよい。行わない場合は (III) までで 1 ステップの学習とする。

### 4 SEEM による地図獲得

この実験は移動ロボットに備えられた全方位カメラの画像データを SEEM に学習させることにより、移動ロボットが自己組織的に内部モデルを獲得できることを示すものである。図 (2) は移動ロボットが活動するフィールドと獲得した内部モデルを示したものであり、実際に内部モデルとしてグラフ構造状のトポロジカル地図を獲得できていることがわかる。ここで、画像データは常に SEEM に入力されており全ての学習はオンラインで行われていることに注目して欲しい。これは SEEM の持つ非常に高速な追加学習能力により実現されているものであり、実際のロボットへ応用する場合 SEEM が強力な手法となり得る事を示していると言えるだろう。

#### 参考文献

- [1] 川畑宣之, 徳永憲洋, 古川徹生 (2008) “自己進化する新しいモジュラーネットワークの提案.” ニューロコンピューティング研究会 信学技報, vol. 107, no. 413, NC2007-111, pp. 141-146, 2008 年 1 月