1D3-04

基底関数間相互作用に基づく状態空間自己組織化

State Space Self Organization based on the Interaction between Basis Functions

関野 正志　片上 大輔　新田 克己
Masashi SEKINO　Daisuke KATAGAMI　Katsumi NITTA

東京工業大学大学院 総合理工学研究科
Department of Computational Intelligence and Systems Science Tokyo Institute of Technology

This paper proposes a on-line method for constructing the continuous state space. We have so far developed *Self Organizing Basis Network (SOBN)* which is the construction method for a function approximator by self organizing a basis function network. The method constructs a basis function network by connecting neighbor basis functions using an edge. The basis function network enables the method to create a function approximator by top-down process as regional division and bottom-up process as adaptation of location of each basis. We employ *SOBN* for constructing the continuous state space, and construct the state space by environment and output of itself in perception-action cycle. Assuming that *SOBN*'s state space is applied to *Reinforcement Learning*, we apply the method to the function approximation problem and evaluate approximation performance.

1. はじめに

強化学習は，環境から得られる報酬信号を手がかりとして，状態から出力への写像を獲得することを目的とする．このとき，状態空間は，強化学習の進行に伴い変化する状態価値関数や方策関数を表現可能でなければならない．実環境ではこの写像において，連続値の状態入力と行動出力を扱うことが求められる．TD 学習法の実現方法の 1 つである Actor-Critic 法は，Actor と呼ばれる制御器と Critic と呼ばれる状態価値推定器に関数近似器を用いることで，連続値を扱うことが可能 [Doya 96] である．関数近似タスクの収束と学習コストは，関数近似器の分解能などの設計に強く依存するため，関数近似器は目標関数に応じて設計されることが望まれる．さらに，強化学習に関数近似器を用いる場合には，学習の進行に伴い変化する目標関数に応じて，オンラインで関数近似器の設計を適応的に変化させられることが望ましい．

近似器を適応的に構成することに向いた構造的特徴を有するモデルに，入力空間にガウス動径基底関数（Gaussian Radial Basis Function: GRBF）を配置し，各 GRBF の出力を正規化して出力を構成する正規化ガウス関数ネットワーク（Normalized Gaussian network: NGnet）がある．実際に，NGnet を Actor-Critic 法に導入し，関数近似器を適応的に構成しながら強化学習を行う手法が提案されている．各基底関数の配置と分散共分散行列をそれぞれオンライン EM アルゴリズム [吉本 03]，勾配法 [Morimoto 98]，進化的手法 [近藤 03] により調整し，既存基底の外挿的出力では誤差が大きい場合に基底を追加するボトムアップ構成法と，初期状態は 1 つの大きな基底によって全領域をカバーし，誤差の分布に基づいて基底を適応的に分割していくトップダウン構成法 [Samejima 99] がある．ボトムアップ構成法では分散共分散行列を各基底が独立に学習するため，新たに追加する基底の大きさに関する指針を持たない．そのため，各基底が学習した結果，基底の担当領域が他と重なりあってしまい，基底が不要になってしまうことが起こり得る．トップダウン構成法では，各基底のボトムアップな適応がないために，徐々に変化する目標関数に追従して，最小コストで近似器を再構成することが困難である．

筆者らはこれまでに，基底関数に基づく関数近似器を，基底相互の近傍関係をエッジで結んだネットワークとして構成し，トップダウンな領域分割の側面とボトムアップな適応性を持つ更新則により，関数近似器を自己組織的に構成する Self Organizing Basis Network（SOBN）を開発してきた．本稿では，SOBN を Actor-Critic 法に実装することを想定し，徐々に変化する目標関数への追従性能を評価する．

連絡先: 関野 正志，東京工業大学総合理工学研究科，横浜市緑区長津田町 4259，sekino@ntt.dis.titech.ac.jp

2. 連続値状態空間

2.1 Actor-Critic 法

Actor-Critic 法は図 1 に示すような，Actor と呼ばれる制御器と Critic と呼ばれる状態価値推定器を用いる．Critic は状態 s に対して現在の Actor の下での状態価値関数 $V(s)$ を推定し，Actor はその状態価値関数に基づいて計算される TD 誤差に従って方策 $A(s)$ を改善する．Actor-Critic 法は Actor 及び Critic に関数近似器を用いることで，連続値を扱うことができる．TD 誤差 $\epsilon(t)$ は（1）式で計算される．

$$\epsilon(t) = r(t+1) + \gamma V(t+1) - V(t) \tag{1}$$

ここで r は報酬，γ は割引率である．

2.2 正規化ガウス関数ネットワーク

NGnet は入力空間に GRBF を配置し，各 GRBF の出力を正規化して出力する．入力 $x(t) \in R^M$ に対して，基底関数 i の活性 $\phi_i(t)$ は（2）式で与えられる．

$$\phi_i(t) = \exp\left\{-\frac{1}{2}(x(t)-\mu_i)^T\Sigma_i^{-1}(x(t)-\mu_i)\right\} \tag{2}$$

基底 i は，中心 μ_i と分散共分散行列 Σ_i によって自身が活性化する領域を管理する．

NGnet では，各基底の出力を（3）式により正規化する．

$$b_i(t) = \frac{\phi_i(t)}{\sum_j \phi_j(t)} \tag{3}$$

GRBF の活性（2）式では大域的汎化能力は望めないが，(3) 式の正規化処理により基底が配置されていない領域に対しては外挿的にシグモイド状の性質を，基底関数が配置されている領域に対してはガウス状の性質をエミュレーションすることができる．

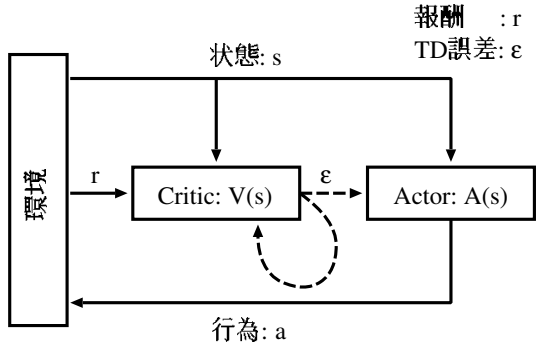


図 1: Actor-Critic 法

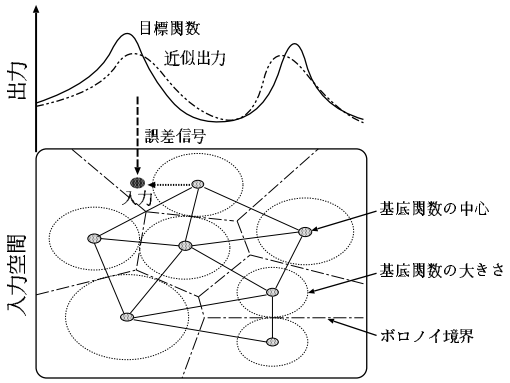


図 2: Self Organizing Basis Network

NGnet は GRBF が目標関数と要求精度に相応な数 N_B 個あるとき，パラメータ（中心ベクトル μ，分散共分散行列 Σ）と結合重み w を適切に決めることで任意の連続写像を近似できる．NGnet の出力 $y^*(t)$ は各正規化出力 $b_i(t)$ の線形和として（4）式で計算される．

$$y^*(t) = \sum_{i=1}^{N_B} w_i b_i(t) \tag{4}$$

結合重み w は誤差信号 $\epsilon(t)$ に対して（5）式により修正される．

$$\Delta w_i(t) = \eta b_i(t) \epsilon(t) \tag{5}$$

ここで η は学習率である．

2.3 Self Organizing Basis Network

SOBN は，基底関数に基づく関数近似器を，基底相互の近傍関係をエッジで結んだ基底関数ネットワークとして構成し，基底の追加位置と各基底の担当領域を決める指針として用いる．つまり基底の追加と各基底の担当領域は，トップダウンな領域分割の視点から行われる．また，各入力に対する最近傍基底が誤差信号に基づき配置を修正することで，近似器構成をボトムアップに改善する．ここで，トップダウンに決まる担当領域は，ボトムアップに更新された基底関数ネットワークに依存し，ボトムアップな配置修正は，現在の近似器構成から得られる誤差に基づく．SOBN は，この相互依存性を適切な近似器構成の探索に利用することで，トップダウンな領域分割とボトムアップな適応性の両側面を近似器構成に反映させる．

SOBN の概略図を図 2 に示し，アルゴリズムを図 3 に簡単にまとめる．本稿では SOBN により NGnet を構築する．各

step 1 基底関数ネットワークの構成
教師なし学習の枠組であるトポロジーマッピングの手法を用いて，基底関数間の近傍関係をエッジで結び基底関数のネットワークを構成する．
step 2 配置更新
ボトムアップに近似器構成を改善するために，入力信号と誤差信号に基づいて配置更新を行う．
step 3 分散パラメータの更新
基底関数ネットワークに基づいて分散パラメータの設定を行う．
step 4 基底の追加
基底関数ネットワークに基づいて，新たな基底が必要な場所を絞りこみ，基底の追加基準が満たされた場合に基底を追加する．

図 3: Self Organizing Basis Network のアルゴリズム

CHL:
（1）各入力信号に対し，nearest center と second-nearest center を探す．
（2）両者がエッジで結ばれていなければエッジを追加して両者を結ぶ．

図 4: 競合ヘッブ学習アルゴリズム

aging scheme:
（1）各入力信号に対し，nearest center が持つ全てのエッジに年齢を加算．
（2）nearest center と second-nearest center 間のエッジの年齢をリセット．
（3）年齢が閾値を越えたエッジを削除．

図 5: aging scheme アルゴリズム

ガウス基底の大きさは，図 2 に示すように標準偏差をボロノイ境界に内接するように設定する．以下，各 step について詳しく説明する．

step 1：基底関数ネットワークの構成

基底間の近傍関係を，トポロジーマッピングの手法である競合ヘッブ学習アルゴリズム（Competitive Hebbian Learning: CHL）及び aging scheme[Martinetz 94] を用いることでエッジで結び，ネットワークとして構成する．CHL 及び aging scheme のアルゴリズムを図 4，図 5 に示す．

step 2：配置更新

各基底は近似誤差 $\epsilon(t)$ に対し，(6) 式で逐次更新される二乗誤差の加重移動平均 f_i 及び (7) 式で定義される更新コストの評価パラメータ L_i を保持し，配置と分散パラメータの更新を行う指針とする．

$$f_i(t+1) = (1-\lambda) f_i(t) + \lambda \epsilon(t)^2 \tag{6}$$

$$\Delta L_i(t) = b_i(t) \tag{7}$$

基底 i が入力 $x(t)$ に対して最近傍基底であるとき，f_i 及び L_i を更新する．L_i は，基底 i が新たな基底を追加したとき初期化される．

各入力に対する最近傍基底 i は，L_i が更新コストの閾値 θ_L を越えていて，二乗誤差の加重移動平均 f_i が閾値 θ_f を上回るとき，最近傍基底 i の配置及び分散パラメータを更新する．具体的には，入力 $x(t)$ に対する最近傍基底 i の中心 μ_i を，(8) 式に従って更新する．

$$\mu_i(t+1) = \alpha \times \delta(t) \times (x(t) - \mu_i(t)) \tag{8}$$

α は入力分布のトポロジー学習のための移動率であり，近似誤差 $\epsilon(t)$ の評価値 $\delta(t)$ によって重みづけされる．$\delta(t)$ を（9）式で定義する．

$$\delta(t) = \frac{\epsilon(t)^2}{f_i(t)} \tag{9}$$

δ は，誤差の大きな領域への移動率を高める働きをする．本稿では（9）式により，現在の近似器構成で誤差の大きな領域にエキスパートを配置し，近似器構成をボトムアップに改善するものとする．

step 3：分散パラメータの更新

基底関数ネットワークに基づいて，トップダウンな視点から分散パラメータを更新する．本稿では（10）式のような簡単化した共分散行列 Σ_i を用いる．

$$\Sigma_i = \sigma_i^2 I \tag{10}$$

I は単位行列，σ_i は基底 i の標準偏差である．本稿では，分散パラメータは “隣接基底のうち一番近い基底とのユークリッド距離の半分を標準偏差” に定める．つまり，

$$\sigma = \frac{\arg\min(\ distance\ to\ neighbor\ )}{2} \tag{11}$$

により定める．このように分散パラメータを決めると，図 2 に示すように各基底関数の分散はボロノイ境界に内接する標準偏差を持つことになる．

step 4：基底の追加

基底関数ネットワークに基づいて新たな基底が必要な場所に基底を追加する．各入力に対する最近傍基底 i の，更新コスト L_i が閾値 θ_L を，二乗誤差の加重移動平均 f_i が閾値 θ_f を越えていて，さらに（12）式で表される基準が満たされたとき，隣接する基底の中で最大の f_j を持つ基底 j とのエッジの中間に新しい基底を追加する．

$$\frac{g_i^2}{f_i} < \theta_d \tag{12}$$

ここで g_i は，基底 i が入力 $x(t)$ に対して最近傍基底であるときに（13）式で更新される誤差 $\epsilon(t)$ の加重移動平均である．

$$g_i(t+1) = (1-\lambda)g_i(t) + \lambda\epsilon(t) \tag{13}$$

この基準は，重み修正が十分になされ平均誤差が収束しても，分散が大きいままならば分解能が足りないと判断することに相当する．

2.4 SOBN による Actor-Critic 状態空間構成

SOBN による Actor-Critic 状態空間構成法を図 6 に示す．Actor 及び Critic の出力 $A(s)$，$V(s)$ は，SOBN 層ユニットの出力を結合重み w_i，v_i で重み付けして線形和を取る．TD 誤差に基づいて Actor は状態価値をより高めるように，Critic は状態価値を正しく見積もるようにそれぞれ w_i，v_i を修正する．

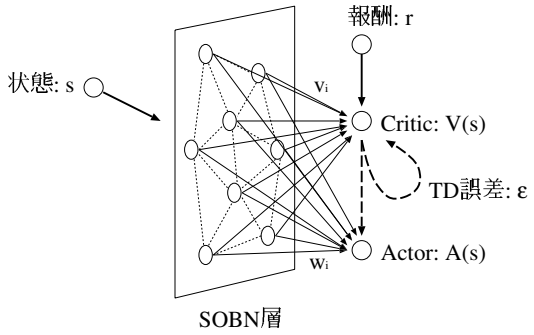


図 6: SOBN による Actor-Critic 状態空間構成

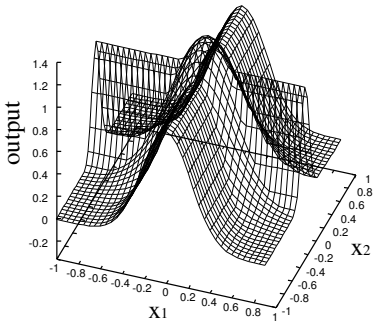


図 7: 目標関数

3. シミュレーション

3.1 タスク

SOBN によって状態空間を構成することの有効性を確認するために，(14) 式を目標関数として，入力の分布が徐々に変わる関数近似タスク [Sato 00] を行う．

$$y = \max\left\{e^{(-50x_1^2)}, e^{(-10x_2^2)}, 1.25e^{(-5(x_1^2+x_2^2))}\right\} \tag{14}$$

目標関数を図 7 に示す．学習データに用いる出力変数 y には，平均 0，標準偏差 0.1 の大きなガウス雑音を付加する．以下，ノイズを付加したものを $\tilde{y}$，真の値を y と記す．関数近似タスクは，以下の step 1〜4 を 1 loop として繰りかえすことにより行われる．

step 1: 入力変数の分布に従って $x(t)$ を発生させ，NGnet に入力する．

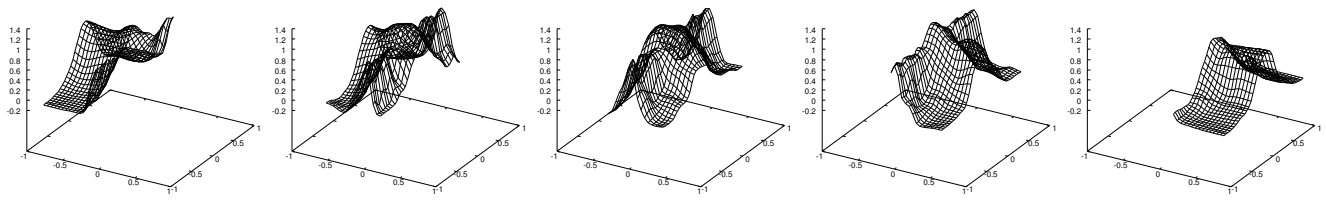
step 2: NGnet は出力 $y^*(t)$ を返し，$\tilde{y}(t)$ を受け取る．

step 3: SOBN によりパラメータの更新，基底の追加を行う．

step 4: 結合重み w を（5）式により修正する．

入力変数 x_1 の分布は $[-1, 1]$ の一様分布に固定であり，入力変数 x_2 の分布を変化させる．総 loop 数と分布の移動速度は参考文献 [Sato 00] に合わせ，総 loop 数を 250000 とした．移動速度は 500 loop を 1 epoch として，500 epoch の間に入力変数 x_2 の分布を $[-1, -0.2]$ の一様分布から $[0.2, 1]$ の一様分布へと連続的に変化させる．

図 8 に，シミュレーションに用いたパラメータをまとめた．



(a) 50000 loop (b) 100000 loop (c) 150000 loop (d) 200000 loop (e) 250000 loop

図 9: SOBN により学習された関数

α :	トポロジーの学習率	0.2
λ :	加重移動平均の重み	0.1
η :	結合重みの学習率	0.1
θ_f:	更新のための閾値	0.1
θ_L:	更新コストの閾値	10
θ_d:	基底追加の閾値	0.5

図 8: シミュレーションに用いたパラメータ

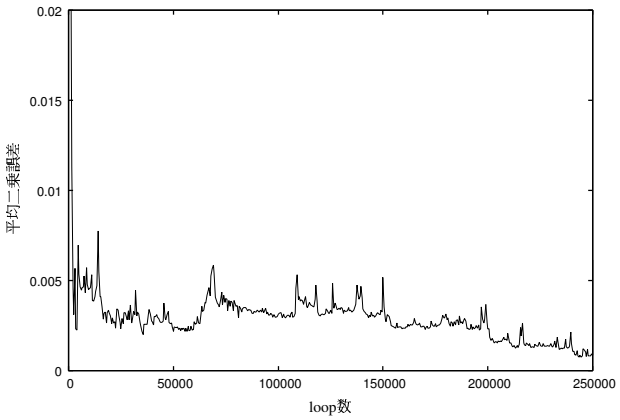


図 10: 平均二乗誤差の推移

3.2 シミュレーション結果

図 9 に入力分布上での近似出力の推移を示す．入力の分布は移り変わっていくが，SOBN は近似に必要な近似器構成を獲得し，安定して近似を実現していることがわかる．真の値 y に対する平均二乗誤差の推移を図 10 に示す．横軸は loop 数であり，縦軸は入力分布上での平均二乗誤差を示す．入力の分布が移動しているにも関わらず，誤差が安定して低く保たれていることがわかる．入力空間に配置された基底の総数は 85 個であった．

4. おわりに

基底関数に基づく関数近似器を，トップダウンとボトムアップに構成する SOBN により，連続値状態空間を構成する手法を提案した．そして，Actor-Critic 法への実装法を示し，強化学習問題への適用に向けて，目標関数が徐々に変化する関数近似タスクを行った．提案手法が，関数近似器構成を追従的に変化させることで，徐々に変化する目標関数に対しても安定して近似する能力を持つことを確認した．

今後の課題として，等方等分散ではないガウス基底の導入，基底の削除機構の導入を検討している．等方等分散ではなく，次元間の相関を考慮したガウス基底により基底の数を減らすことができる．また，目標関数が容易に近似できる形状に変化した領域は，基底を減らすことで学習コストを低減することができる．本稿では各基底の配置と分散パラメータの更新を行うかどうかに関して，閾値 θ_f を導入したが，配置更新則の最適性についてさらに言及していきたいと考えている．そして，実際に強化学習問題に適用し，提案手法の有効性を確認することが必要である．

参考文献

[Doya 96] Doya, K.: Temporal Difference Learning in Continuous Time and Space, in Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E. eds., *Advances in Neural Information Processing Systems 8*, pp. 1073–1079, The MIT Press, Cambridge, MA (1996)

[Martinetz 94] Martinetz, T. and Schulten, K.: Topology Representing Networks, *Neural Networks*, Vol. 7, No. 3, pp. 507–522 (1994)

[Morimoto 98] Morimoto, J. and Doya, K.: Reinforcement learning of dynamic motor sequences: Learning to stand up, in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 3, pp. 1721–1726, Victoria, B.C. Canada (1998)

[Samejima 99] Samejima, K. and Omori, T.: Adaptive internal state space construction method for reinforcement learning of a real-world agent, *Neural Networks*, Vol. 12, No. 7-8, pp. 1143–1155 (1999)

[Sato 00] Sato, M. and Ishii, S.: On-line EM algorithm for the normalized gaussian network, *Neural Computation*, Vol. 12, No. 2, pp. 407–432 (2000)

[吉本 03] 吉本 潤一郎, 石井 信, 佐藤 雅昭：連続力学システムの自動制御のためのオンライン EM 強化学習法, システム制御情報学会論文誌, Vol. 16, No. 5, pp. 209–217 (2003)

[近藤 03] 近藤 敏之, 伊藤 宏司：進化的 recruitment 戦略を用いた強化学習による自律移動ロボットの制御器設計, 計測自動制御学会論文集, Vol. 39, No. 9, pp. 857–864 (2003)