複数勝者 KFM 連想メモリを用いた強化学習の実現

池谷孝裕　長名優子

東京工科大学大学院 バイオ・情報メディア研究科コンピュータサイエンス専攻

1 はじめに

近年，自律分散型ロボットやマルチエージェントの研究が盛んになるにつれ，学習主体者が環境と相互作用し情報の獲得と行動を選択する学習方法として強化学習の研究も盛んになっている．強化学習では Q-learning や Actor-Critic[1] など多くの手法が提案されている．また，ニューラルネットワークを用いて強化学習を実現するようなモデルも提案されているが，これらのモデルでは追加 (逐次) 学習ができないため，環境が変化すると学習を再度行わなくては適切な行動をとれないという問題がある．

本研究では，逐次学習が可能な複数勝者 KFM (Kohonen Feature Map) 連想メモリを用いて Actor-Critic 手法による強化学習を実現する．

2 複数勝者 KFM 連想メモリを用いた強化学習

提案する複数勝者 KFM 連想メモリは，逐次学習が可能な KFM 連想メモリ [2] を入力に類似した重みを持つ複数のニューロンが発火できるように拡張したモデルである．複数のマップ層ニューロンの重みから出力を決定することで，未学習の入力に対しても類似した複数の既学習のデータをもとに適切な出力を生成することが期待できる．複数勝者 KFM 連想メモリは図 1 に示すように入出力層とマップ層の 2 層から構成されており，入出力層は複数のパターンを表す部分に分けられる．このモデルを強化学習に用いる場合には，入出力層は状態と行動を表す 2 つの部分に分けられる．

提案モデルにおいて強化学習は以下のような流れで行う．

(1) 複数勝者 KFM 連想メモリの重みを小さなランダムな値で初期化する．

(2) エージェントが環境 $\boldsymbol{s}(t)$ を観測し，複数勝者 KFM 連想メモリにより，行動 $\boldsymbol{a}$ を決定する．行動の決定は以下のような手順で行う．

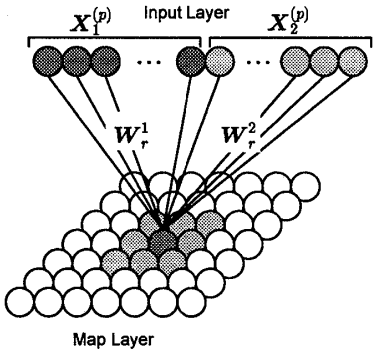


図 1: 複数勝者 KFM 連想メモリの構造

(a) マップ層の各ニューロンについて内部状態を計算する．内部状態 u_i^{map} は以下のように与えられる．

$$u_i^{map} = \frac{1}{N^{in}} \sum_{k=1}^{N^{in}} g(X_k(t) - W_{ik}) \quad (1)$$

$$g(b) = \begin{cases} 1, & |b| < \theta^d \\ 0, & \text{それ以外} \end{cases} \quad (2)$$

ここで，$\boldsymbol{X}(t)$ は，$\boldsymbol{X}(t) = ((\boldsymbol{s}(t))^T, \boldsymbol{0}^T)^T$ で与えられる入力ベクトルである．

(b) 内部状態が閾値以上になるマップ層のニューロンを勝ちニューロンとする．内部状態が閾値以上のニューロンが存在しない場合には，内部状態が最大値をとるニューロンを勝ちニューロンとする．

(c) 各勝ちニューロンの内部状態の大きさに基づく影響力を考慮して，入出力層の出力を以下のように決定する．

$$x_k^{in} = \sum_{i \in R} \frac{u_i^{map}}{\sum_{j \in R} u_j^{map}} W_{ik} \quad (3)$$

ここで，R は勝ちニューロンの集合を表す．

(3) エージェントが行動 $\boldsymbol{a}$ を実行することにより，状態が $\boldsymbol{s}(t+1)$ に遷移する．

(4) クリティックは，環境の状態 $\boldsymbol{s}(t+1)$ から報酬 $r(t+1)$ を受け取り，TD 誤差 δ を出力する．

$$\delta = r(s(t)) + \gamma V(s(t)) - V(s(t-1)) \quad (4)$$

Realization of Reinforcement Learning using Multi-Winners KFM Associative Memory
Takahiro Ikeya and Yuko Osana (Tokyo University of Technology, ikeya@osn.cs.teu.ac.jp, osana@cc.teu.ac.jp)

(5) TD 誤差 δ に基づき状態価値関数 $V(s)$ を次のように更新する.

$$V(s) \leftarrow V(s) + \gamma\delta \tag{5}$$

ここで, γ $(0 \leq \gamma \leq 1)$ は学習率である.

(6) TD 誤差に基づいて重みの更新を行う. TD 誤差が正の場合には, 直前に観測された状態と行動が学習される. TD 誤差が負の場合には, 直前に観測された状態と行動を忘れる方向に重みを更新する.

(a) 学習ベクトル $\boldsymbol{X}(t)$ と重みベクトル $\boldsymbol{W}_i$ のユークリッド距離 $d(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i)$ を計算し, ユークリッド距離が最小となるマップ層のニューロンを勝ちニューロン r とする.

(b) $d(\boldsymbol{X}^{(p)}, \boldsymbol{W}_r) > \theta^l$ のとき, 重みの値が固定されていないニューロンに結合する重みを以下の更新式に基づいて更新する.

$$\Delta \boldsymbol{W}_i(t) = \delta H(d_i)\alpha(t)h_{ri}(\boldsymbol{X}^{(p)} - \boldsymbol{W}_i(t)) \tag{6}$$

ここで, h_{ri} は近傍関数であり,

$$h_{ri} = \exp\left(\frac{-||\boldsymbol{r} - \boldsymbol{i}||^2}{2\sigma(t)^2}\right) \tag{7}$$

で与えられる. ここで, $\sigma(t)$ は単調減少関数であり,

$$\sigma(t) = \sigma_i \left(\frac{\sigma_f}{\sigma_i}\right)^{t/T} \tag{8}$$

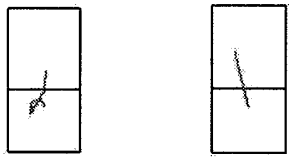
で表される. $\sigma(t)$ は学習開始時には σ_i であり, 終了時には σ_f(ただし $\sigma_i > \sigma_f$) となる. また, T は最大の学習回数を表す. 式 (6) において $\alpha(t)$ は学習係数であり,

$$\alpha(t) = \frac{-\alpha_0(t - T)}{T} \tag{9}$$

のような単調減少関数で与えられる. ここで, α_0 は $\alpha(t)$ の初期値である. また, $H(d_i)$ は

$$H(d_i) = \tanh(d_i/\varepsilon). \tag{10}$$

で与えられる. d_i はニューロンとそのニューロンから最も近い位置にある重みが固定されているニューロンとの距離, ε は $H(d_i)$ の傾きを決める係数である. $H(d_i)$ の値は d_i の値が小さいほど, つまり重みが固定されたニューロンとの距離が近いほど小さくなるようになっており, それによって準固定を実現している.



(a) 学習前の軌跡　　(b) 学習後の軌跡

図 2: 強化学習の結果

(d) $\delta > 0$ の場合には $d(\boldsymbol{X}^{(p)}, \boldsymbol{W}_i) \leq \theta^l$ になるまで (a)〜(c) を繰り返す. $\delta < 0$ の場合にはあらかじめ決めておいた回数 (T) だけ (a)〜(c) を繰り返す.

(e) $\delta > 0$ の場合には, 勝ちニューロン r に結合する重み $\boldsymbol{W}_r$ を固定する.

(7) エージェントが目的を達成するまで (2)〜(6) を繰り返す.

3 計算機実験

提案モデルを用いて車の車庫入れを例に実験を行った. 車は前後に動くことができ, ハンドルは一定の範囲の角度できれるものとする. また, 環境としては, 車と駐車スペースの頂点の座標との距離, 直前の動作, ハンドルの角度を用いる.

図 2 に提案モデルでの強化学習の結果を示す. この図では上のスペースが駐車スペースとなっており, 図中の赤い線は車が移動した軌跡を示している. この結果より学習後車が駐車スペースにスムーズに向かっていることから提案モデルにおいて強化学習が行えることが確認できる.

参考文献

[1] I. H. Witten : "An adaptive optimal controller for discrete-time Markov environments," Information and Control, Vol.34, pp. 286–295, 1977.

[2] T. Yamada, M. Hattori, M. Morisawa and H. Ito: "Sequential learning for associative memory using Kohonen feature map," Proceedings. of IEEE and INNS International Joint Conference on Neural Networks, paper no.555, Washington D.C., 1999.