

## 研究速報

記憶容量制限下において手続き的知識を獲得する  
連想記憶モデル

郷古 学<sup>†a)</sup>(学生員) 菅谷 至寛<sup>†</sup>(正員)  
阿曾 弘具<sup>†</sup>(正員:フェロー)

Associative Memory Model to Acquire Procedural Knowledge  
under Restriction of Memory Capacity

Manabu GOUKO<sup>†a)</sup>, Student Member,  
Yoshihiro SUGAYA<sup>†</sup>, Member, and Hirotomoto ASO<sup>†</sup>, Fellow

<sup>†</sup> 東北大学大学院工学研究科, 仙台市

Tohoku University, Sendai-shi, 980-8579 Japan

a) E-mail: gouko@aso.ecei.tohoku.ac.jp

あらまし 本論文では, 記憶容量制限下において, 周囲の環境から適応的に手続き的知識を獲得・更新することが可能な連想記憶モデルを提案する. 計算機実験により, 記憶容量制限下において環境に適応した手続き的知識を効率的に獲得することを検証した.

キーワード 手続き的知識, 連想記憶モデル, ニューラルネットワーク, 記憶容量

### 1. まえがき

人間のもつ知識の一つである手続き的知識は, 物事の実行手順に関する知識である. これまでに Newell のプロダクションシステムをはじめとして, 手続き的知識を利用した多くの知的システムが提案されてきた [1]. ニューラルネットワークにより手続き的知識を表現する場合, 連想記憶モデルを用いて現在実行中の手続き  $A$  と, その次に実行すべき手続き  $B$  の間の連合を形成することで表現することが可能である [2].

従来の手続き的知識を表現する連想記憶モデルの多くは, あらかじめ知識を学習により記憶しておく必要がある (事前知識) [2]. しかし, このような学習を行ったモデルは, 事前知識を逸脱する環境下においては機能せず, 環境の変化に対して適応することができない. また, すべての環境下で必要とされる膨大な数の事前知識を網羅的に学習するということが現実的ではない. 記憶容量の制限下において, モデルが表現可能な知識の数は限られているため, 自らが置かれた環境において重要な知識を優先的に学習し, 環境の変化に合わせて自律的に自らのもつ知識を変化させていくモデルが必要となる. 適応性に関しては, 入力情報を逐次的に学習するモデル [3] が提案されているが記憶容量の制限については考慮されていない.

そこで本研究では, 記憶容量制限下において環境に適応的に手続き的知識を獲得する連想記憶モデルを提

案する. 提案モデルは, 芳澤ら [4] の提案した自己連想記憶モデルの競合層 (記憶層) を 2 層構造にし, 各層を連携させる機構をもつ. 各記憶層には現在実行中の手続きと, 次に実行すべき手続きが記憶される. 更にモデルは環境の変化に応じて自らのもつ記憶を変化させる. また, より重要な手続き的知識を優先的に獲得するという, 記憶容量制限下において効率的な知識獲得が可能なモデルとなっている.

### 2. 手続き的知識について

モデルへの入力情報は手続きの構成要素であるパターンとする. 「 $A$  の次に  $B$ 」というように二つのパターンから構成される手順を手続き的知識の 1step と考え, 「 $A \rightarrow B$ 」と表す. このときパターン  $A, B$  は因果関係をもつと考える. 提案モデルは次々と入力されるパターンの入力間隔をもとに因果関係を決定する. つまりモデルに対してパターン  $A$  の入力後,  $B$  が入力されるまでの時間が短いほど,  $A, B$  の因果関係は強いと考え, 逆にそれぞれのパターンが入力される時間間隔が長いほど, 二つのパターンの因果関係は弱いと考える. 本研究では因果関係が強いほど, 重要な手続き的知識を意味していると考え. また入力頻度が高いパターンをその環境における重要な手続きと考え. 提案モデルは記憶容量を有効利用するため, 重要度の高い情報を優先的に記憶する.

### 3. 提案モデル

提案モデルの構造を図 1 (a) に示す. モデルは  $m_0$  個のニューロンからなる入力層, それぞれ  $m_1$  個のニューロンからなる二つの記憶層をもち, 入力層の各ニューロンは各記憶層のニューロンと全結合している. 時刻  $t$  における入力層の  $i$  番目のニューロンと, 記憶層  $n$  ( $= 1, 2$ ) の  $j$  番目のニューロンとの結合重みを  $W_{ji}^n(t)$  と表す. 記憶層  $n, j$  番目のニューロンへの結合重みをまとめてベクトル表現し,  $W_j^n(t) = (W_{j1}^n(t), W_{j2}^n(t), \dots, W_{jm_0}^n(t))$  とおく. ま

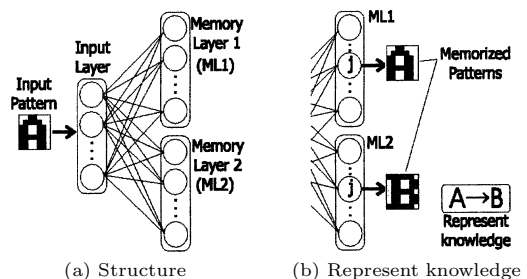


図 1 提案モデル

Fig. 1 Proposed model.

た、各記憶層のすべてのニューロンは、記憶パターンの信頼度  $R_j^n(t)$  と角度しきい値  $\theta_{max,j}^n(t)$  の二つのパラメータをもつ。更に各記憶層の  $j$  番目のニューロンは、共通の記憶パラメータ  $p_j(t)$  をもつ。記憶層 1 から 2 へはパラメータ  $p_j(t)$  を通じて影響を与える。

記憶層はモデルに対して入力されるパターンを自らの結合重みに蓄える。一つのニューロンは、自らの重みベクトルにより一つの記憶パターンを記憶する。記憶層 1, 2 のある  $j$  番目のニューロンがそれぞれパターン  $A, B$  を記憶したとき、モデルが「 $A \rightarrow B$ 」という知識を獲得したと考える (図 1 (b))。

提案モデルはパターンが入力されると、既に記憶したパターンの想起 (想起処理) と入力パターンの記憶 (記憶処理) という二つの処理を同時に行う。

### 3.1 想起処理

入力パターンはベクトルとして扱う。時刻  $t$  における入力パターンベクトルを  $h^0(t) = (h_1^0(t), h_2^0(t), \dots, h_{m_0}^0(t))$  とし、入力層では正規化処理を行う。すなわち、入力層のニューロン  $i$  の出力  $V_i^0(t)$  を  $V_i^0(t) = h_i^0(t) / \|h^0(t)\|$  とする。記憶層  $n$  では、各ニューロン  $j$  の出力  $V_j^n(t)$  を次式で決める。

$$V_j^n(t) = \sum_{i=1}^{m_0} W_{ji}^n(t) V_i^0(t) \quad (1)$$

重みベクトル  $W_j^n(t)$  の大きさ  $\|W_j^n(t)\|$  をニューロン  $j$  の記憶パターンの信頼度  $R_j^n(t)$  と呼ぶ。入力層の出力をベクトル表現し、 $V^0(t) = (V_1^0(t), V_2^0(t), \dots, V_{m_0}^0(t))$  とおく。  $V^0(t)$  と  $W_j^n(t)$  のなす角度  $\theta_j^n(t)$  は  $\theta_j^n(t) = \cos^{-1}(V_j^n(t)/R_j^n(t))$  と表される。想起処理では各記憶層のニューロン  $j_{recall}$  (想起ニューロン) のもつ記憶パターンを想起し出力する。  $j_{recall}$  は次の手順 (i) ~ (iv) により決定する (図 2 (a) 参照)。 (i) 式 (1) において  $V_j^1(t)$  を最大とする  $j$  を選択する。 (ii)

$\theta_j^1(t) \leq \theta_{max,j}^1(t)$  ならば、  $j_{recall} = j$  とする。 (iii) そうでない場合は  $j$  を除いて、 (i), (ii) の操作を繰り返す、想起ニューロンを決定する。 (iv) すべての  $j$  で  $\theta_j^1(t) > \theta_{max,j}^1(t)$  ならば、記憶パターンの想起を行わず、後述の重みベクトルの更新も行わない。図 2 (a) の例では、  $j_{recall} = 1$  となる。

記憶層  $n$  の想起パターン  $y_i^n(t)$  は、想起ニューロン  $j_{recall}$  の重みベクトルとする。すなわち、

$$y_i^n(t) = W_{j_{recall} i}^n(t) \quad (2)$$

### 3.2 記憶処理

記憶処理では、入力パターンを記憶するために、各記憶層のニューロンの重みの更新、忘却処理、角度しきい値と記憶パラメータ  $p_j(t)$  の更新が行われる。

記憶層  $n$  のニューロン  $j^n$  (更新ニューロン) の結合重みを、式 (3) に示す Oja 則 [5] を用いて更新する。

$$\begin{aligned} W_{j^n i}^n(t+1) \\ = W_{j^n i}^n(t) + \eta \alpha_n V_{j^n}^n(t) (V_i^0(t) - V_{j^n}^n(t) W_{j^n i}^n(t)) \end{aligned} \quad (3)$$

ここで、  $\eta$  はある定数で、学習係数である。また、  $\alpha_1 = 1$ 、  $\alpha_2 = p_j(t)$  である。重みベクトルの大きさは更新により 1 に収束する [5]。

記憶層 1 の更新ニューロンは  $j^1 = j_{recall}$  とする。記憶層 2 の更新ニューロン  $j^2$  は次の手順 (i) ~ (iii) により決定する。 (i)  $j = j_{recall}$  を除いて、記憶パラメータ  $p_j(t)$  が最大である  $j$  を選択する。 (ii)  $\theta_j^2(t) \leq \theta_{max,j}^2(t)$  ならば、  $j^2 = j$  とする。 (iii) そうでない場合  $j$  を除いて (i) (ii) の操作を繰り返す。更新ニューロン  $j^2$  は、  $p_j(t)$  が角度条件付きで最大となるニューロンであり、一時刻前に入力されたパターンを記憶層 1 で記憶したニューロン番号に一致することが期待されている。

提案モデルは記憶容量の有効利用を目的とした忘却処理を行う。すなわち、上記記憶処理の後にすべての重みベクトルの大きさを次式に従い一定量だけ減ずる。

$$\begin{aligned} W_j^n(t+1) \\ = \begin{cases} \frac{R_j^n(t) - R_{red}}{R_j^n(t)} W_j^n(t) & \text{if } R_j^n(t) > R_{min} \\ \frac{R_{min}}{R_j^n(t)} W_j^n(t) & \text{otherwise} \end{cases} \end{aligned} \quad (4)$$

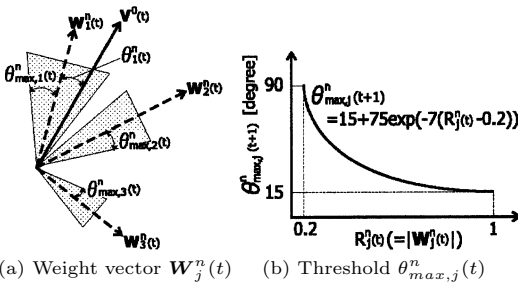


図 2 重みベクトルと角度しきい値  
Fig.2 Weight vector  $W_j^n(t)$  and threshold  $\theta_{max,j}^n(t)$ .

ただし  $j = j^1, j^2$  に対しては、式 (4) の右辺の  $W_{j^1}^1(t)$ ,  $W_{j^2}^2(t)$  は、既に更新した  $W_{j^1}^1(t+1)$ ,  $W_{j^2}^2(t+1)$  を意味し、忘却処理により更に更新するものとする。ここで  $R_{red}$  は入力パターンが 1 回提示されたときの重みベクトルの減少量であり、 $R_{min}$  は重みベクトルの大きさの下限である。この処理により、ある環境下で一度記憶したパターンを環境の変化により忘却し、新たな環境下で重要なパターンを記憶することが可能となる。すなわち記憶容量制限下において、効率の良い記憶を行うことが可能となる。 $R_{red}$ ,  $R_{min}$  は適切な値を設定する必要があるが、本研究では文献 [4] において述べられているこれらの値の設定指針を参考にし、 $R_{red} = 0.0003$ ,  $R_{min} = 0.2$  とした。

次に、各記憶層のすべてのニューロンのもつ角度しきい値を  $\theta_{max,j}^n(t+1) = 15 + 75 \exp(-7(R_j^n(t) - 0.2))$  により更新する。この右辺は図 2 (b) に示すもので、信頼度に応じて角度しきい値を決めている。これにより、信頼度が高い記憶パターンは入力パターンに対して影響を受けにくく頑健になる [4]。

続いて、記憶パラメータ  $p_j(t)$  を更新する。

$$p_j(t+1) = \begin{cases} R_j^1(t) & \text{if } j = j_{recall} \\ p_j(t) - e & \text{otherwise} \end{cases} \quad (5)$$

ただし、 $p_j(t+1) < 0$  の場合は、0 に再設定する。また  $p_j(0) = 0$  とする。 $0 \leq p_j(t) \leq 1$  である。

初期状態のモデルに対して、二つの異なるパターンが続いて入力される場合を例にモデルの動作について説明する。まず最初に入力されたパターンに対して各記憶層で更新ニューロンを決定し、重みの更新と忘却処理・各パラメータの更新がなされる。モデルに対して入力が終了し、次の入力がなされるまでの間、式 (5) により記憶パラメータ  $p_j(t)$  の更新のみを行う。次に別のパターンが入力された際、先に入力されたパターンにより記憶パラメータ  $p_j(t)$  が更新され、 $0 < p_j(t)$  を保っている記憶層 2 のニューロンのもつ重みベクトルが、後から入力されたパターンに更新される。このように提案モデルでは、過去の入力パターンに対応するニューロンの位置に、現在の入力パターンを記憶層 2 に記憶する。つまりモデルは手続き的知識の 1step を獲得していく。また式 (3) において  $\alpha_2 = p_j(t)$  とすることで、前のパターンが入力されてからの時間が短いほど記憶層 2 の記憶が速やかに形成される。

#### 4. 実験結果及び考察

提案モデルが周囲の環境から入力される情報の順

番、入力間隔及び頻度に依存して、手続き的知識を獲得することを数値実験により検証する。実験で用いたネットワークは  $m_0 = 49$ ,  $m_1 = 5$ , 学習係数と減衰係数は  $\eta = 0.01$ ,  $e = 0.1$  とし、実験では  $7 \times 7$  の 6 種類のドットパターン ( $A, B, C, D, E, F$ ) による二つの異なる手続き的知識を表す入力セット  $A \xrightarrow{0} B \xrightarrow{0} C \xrightarrow{0} D \xrightarrow{1} E \xrightarrow{2}$  (入力セット 1) と、 $A \xrightarrow{0} B \xrightarrow{0} F \xrightarrow{0} D \xrightarrow{1} E \xrightarrow{2}$  (入力セット 2) を用いた。「 $\rightarrow$ 」の上の数字は次の入力までの時間間隔を表し、数字のステップ分だけ入力を行わない。このとき記憶パラメータだけが更新される。各パターンの入力では記憶処理を 1 回行う。したがって 8 ステップで一つの入力セットの入力を終了する。次に続く入力セットは確率  $s_1, s_2$  で入力セット 1, 2 を選択し入力する。入力セットにおいて入力時間間隔の多少は、入力パターン同士の因果関係の強さの違いを表現している。実験は  $t = 0$  から 50000 まで行った。実験では  $t = 0$  から 25000 (以下 env1 と呼ぶ) では、 $(s_1, s_2) = (0.85, 0.15)$  とし、 $t = 25001$  から 50000 (以下 env2 と呼ぶ) では、 $(s_1, s_2) = (0.15, 0.85)$  とした。 $s_1, s_2$  はその環境における知識の重要度を意味し、env1 においては入力セット 1, env2 においては入力セット 2 で表される手続き的知識を獲得することが期待される。

実験結果を図 3 に示す。この図は重みベクトルの各成分の大きさを、信頼度で正規化して得られた記憶パターンを表したものである。各パターンの下に記した数字は、その記憶パターンの信頼度である。各時刻の結果において上段、下段がそれぞれ記憶層 1, 2 の記憶パターンである。上下の各パターンは同じ番号のニューロンを表しており、図中の括弧内の数字がニューロンの番号である。モデルは上段のパターンが入力として与えられた場合、下段の記憶パターンは次に入力されるはずのパターンを記憶できるようになることが期待されている。

$t = 7000$  において、記憶層 1 の各ニューロンの記憶パターンの信頼度を見ると、入力頻度の高いパターンを優先的に記憶しており、最も入力頻度が小さい  $F$  が記憶されない。またニューロン 2 に着目すると、記憶層 2 の記憶パターンとして期待される  $C$  ではなく、 $D$  を記憶している。これは記憶層 1 におけるパターン  $C$  の記憶の信頼度が低いために、パターン  $D$  入力時に  $p_2(t) > p_3(t)$  となり、記憶層 2 ではニューロン 3 ではなくニューロン 2 が更新ニューロンになるため

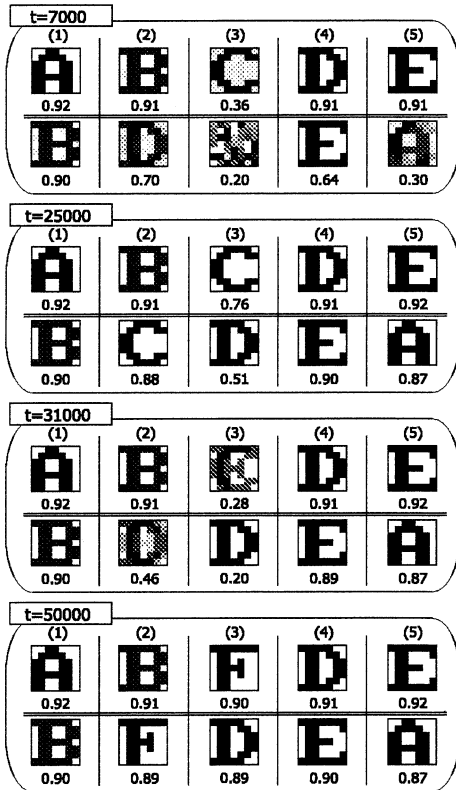


図3 実験結果  
Fig. 3 Memorized patterns.

である。このように、モデルは重要なパターンを優先的に用いて手続き的知識を表現する。また記憶層2においてニューロン1, 4, 5の記憶パターンの信頼度が  $R_1^2(t) > R_4^2(t) > R_5^2(t)$  となっていることが確認できる。これは、 $A \rightarrow B, D \rightarrow E, E \rightarrow A$ の順に因果関係が強い重要な知識であることを反映している。提案モデルはこのように、重要な知識ほど速やかに記憶を形成する。

$t = 25000$  を見ると、記憶層1のパターンCの信頼度の増加により、env1において重要な手続き的知識(入力セット1)を獲得している。

$t = 31000$  を見ると、記憶層1のニューロン3の記憶パターンがFに変化していく様子が確認できる。

これはパターンを記憶するための容量が5であることから、env1でCを記憶したニューロン3において、env2でCの頻度が小さいため忘却が進み、env2で新たに重要なパターンFを記憶していくためである。この影響により  $t = 7000$  の結果と同様の理由で、記憶層2のニューロン2の記憶パターンとして、Dを記憶していく。 $t = 50000$  を見ると、env2において重要な手続き的知識(入力セット2)を獲得している。

以上の実験より、提案モデルは入力情報の順番、入力間隔及び頻度に応じて適応的に手続き的知識を獲得することが分かった。更に知識の獲得過程において、信頼度が高いパターンを優先的に用いて手続き的知識を獲得すること、また記憶容量制限下において、効率的に自らの記憶内容を変化させるという機能も確認できた。

## 5. むすび

本論文では、記憶容量制限下において環境に適応的に手続き的知識を獲得する連想記憶モデルを提案した。提案モデルが周囲の環境からの入力情報をもとに、その環境下において重要な手続き的知識を優先的に獲得していく様子を数値実験により示した。また環境の変化に応じて、記憶容量制限下で効率的に自らの記憶内容を変化させることを確認した。

## 文 献

- [1] 計測自動制御学会編, ニューロ・ファジィ・AIハンドブック, オーム社, 東京, 1994.
- [2] 平井有三, “連想処理モデル HASP による手続き的知識の表現,” 信学論(D), vol.J69-D, no.11, pp.1743-1753, Nov. 1986.
- [3] G.A. Barreto and A.F.R. Araujo, “Unsupervised learning and recall of temporal sequences: An application to robotics,” International Journal of Neural Systems, vol.9, no.3, pp.235-242, June 1999.
- [4] 芳澤伸一, 道木慎二, 大熊 繁, “記憶容量の制約下での追加学習を目的とする新しい連想記憶システムの提案,” 信学論(D-II), vol.J82-D-II, no.6, pp.1072-1081, June 1999.
- [5] J. Hertz, A. Krogh, and R.G. Palmer (著), 笹川辰弥, 吳 勇 (訳), ニューラルコンピュータ: 統計物理学からのアプローチ, トッパン, 東京, 1994.

(平成16年8月9日受付)