

概周期的な時系列情報に含まれる潜在的規則を 発見・記憶・忘却する連想記憶型神経回路モデル

和久屋 寛* ・ 福江 慈孝*

An Associative Memory Model Which Discovers, Memorizes and Forgets an Underlying Rule in Quasi-Cyclic Temporal Sequence Data

By

Hiroshi WAKUYA and Yoshitaka FUKUE

Abstract : Generally speaking, a serial order from the *present* to the *next* is quite important, when we memorize temporal sequence data. Also, we can find out an underlying rule in quasi-cyclic temporal sequence data and extract it easily, even if the conventional engineering systems are hard to deal with. In this paper, therefore, an associative memory model to complete these kinds of functions is considered. As a result of computer simulations, it is found that the proposed model shows good performance to discover and memorize some cyclic sequence successfully from the given quasi-cyclic temporal sequence data. Furthermore, when a new underlying rule is presented, the proposed model can memorize the new one and forget the old one successfully.

Key words : associative memory, temporal sequence, quasi-cyclic data, underlying rule, forgetting

1 はじめに

我々がいろいろな事項を記憶する場合には、いわゆる前後関係に着目していることが多い。例えば電話番号を考えてみよう。一般に、先頭から順番に述べることは容易であっても、最後から逆に唱えることは意外と難しい。この事実は、単に前後関係と言っても、「前→後」と「後→前」とでは、異なる手続きに基づいて独立に記憶されることを示唆している。また我々の日常生活において、前者（正順）を経験することが圧倒的に多いのに対して、後者（逆順）を経験する場合は極めて少なく、学習する機会の多寡が記憶としての残りやすさに対応しているようである。

このような記憶方法は生体に見られる固有なもので、ある情報を他の情報と関連付けて保持するというこ

から連想記憶方式と呼ばれている。生体はこれを実現するため、その構成要素である多数の神経細胞（ニューロン）に種々の情報を分散させて記録しており、そのため、毎日数万個の神経細胞が死んでいるという過酷な状況下でも、正常な情報を保持し続けることが可能であると言われている。ちなみに、現在の電子計算機（コンピュータ）は、記憶装置（メモリ）に割り振られた番地（アドレス）を参照して、その1ヵ所へ情報を集約して記録するため、理論的には故障に弱いなどの性質がある。このように、生体における記憶方式は電子計算機と根本的に原理が異なっている。

ところで、この世の中には、実に多種多様な時系列情報（時間とともに変化する情報）が溢れている。その中には、同じ状態を一定間隔で繰り返す周期的なものもあれば、これとは正反対に、決して同一状態の繰り返しを含むことのない非周期的なものがある。また、その中間的な性質を帯びたものとして、カオス系列のような概周期的な時系列情報もある。なお以下では“概

周期的”という表現を,

- 短期的には, 繰り返しを含まない非周期的な信号と見えるもの。
- 長期的には, 何らかの規則が潜んでいて周期性の存在を感じるもの。

として用いることとする。我々の有している感覚は非常に優れており, 上述の規則を見出すことを, ごく自然に行っている。

これに対して, 我々の脳内における情報処理では, あまり使われない情報は忘れ去られるという傾向もある。電子計算機では, 電源を落としてしまうと揮発性メモリに保存されていた情報が一様に消し去られるのに対して, 上述のように, 我々の記憶は, 選択的に失われるという意味で興味深い特性を有している。実際には, 忘れたと思っていた事柄を, ある日, 突然に思い出す場合もあり, 記憶していた内容が“失われる”のではなく, 単に“呼び出せない”という状況にあるだけとの解釈もあるが定かではないようである。これらは, やはり記憶に関する根本原理が, 生体と電子計算機では異なっていると見るべきであろう。

そこで本研究では, このような記憶機構を実現する神経回路モデルについて検討を行った。以下では, **2.**でいくつかの連想記憶モデルについて紹介し, **3.**で時系列データに潜む規則の記憶と忘却に関する計算機シミュレーション⁽¹⁾⁽²⁾を行う。また**4.**では工学的な応用と生体脳モデルとの類似性という観点から提案モデルについて考察し, 最後に**5.**で結論を述べる。

2 連想記憶を実現する神経回路モデル

2.1 基本的な構成

まず N 個のニューロンから成り, お互いに結合のある相互結合型の神経回路網を考える。このとき, 入力 \mathbf{x} と出力 \mathbf{y} の関係は, ニューロン j からニューロン i への結合荷重を w_{ij} とおくと,

$$y_i = f \left(\sum_{j=1}^N w_{ij} x_j \right) \quad (1)$$

$$f(z) = \begin{cases} +1, & z \geq 0 \\ -1, & z < 0 \end{cases} \quad (2)$$

で表される。

次に, P 個の入出力状態の組 $\{(\mathbf{x}^{[p]}, \mathbf{y}^{[p]}) \mid p = 1, 2, 3, \dots, P\}$ を用意し, これらを上の神経回路網に記憶させることを考える。具体的には, 上述の結合荷重 w_{ij} を,

$$w_{ij} = \frac{1}{P} \sum_{p=1}^P y_i^{[p]} x_j^{[p]} \quad (3)$$

という関係式によって定めればよい⁽³⁾⁽⁴⁾。またここで,

$$\mathbf{y}^{[p]} = \mathbf{x}^{[p+1]} \quad (4)$$

とおいて, ある時刻 t の出力 $\mathbf{y}(t)$ を次の時刻 $t+1$ の入力 $\mathbf{x}(t+1)$ としてフィードバックしてやれば, この神経回路網は, $\mathbf{x}^{[1]} \rightarrow \mathbf{x}^{[2]} \rightarrow \mathbf{x}^{[3]} \rightarrow \dots \rightarrow \mathbf{x}^{[P]}$ という時系列を生成することが可能となる。

2.2 学習機能の付与

ここまでで, 与えられた時系列を生成する神経回路網を作ることができた。ただし, これは式(3)によって事前に定められた固定型結合荷重を用いているため, 学習機能を全く備えていない。そこで, 次に学習によって与えられた順序で時系列を生成することを考える。 n 回目の学習のときに $\mathbf{x}(n)$ が与えられ, その直前が $\mathbf{x}(n-1)$ であった場合, 結合荷重 $w_{ij}(n)$ を,

$$w_{ij}(n) = w_{ij}(n-1) + x_i(n)x_j(n-1) \quad (5)$$

$$w_{ij}(0) = 0 \quad (6)$$

と定めれば, 与えられた時系列情報 $\{\mathbf{x}(n) \mid n = 1, 2, 3, \dots, n_{max}\}$ を順序まで含めて学習することができる⁽⁵⁾。また, 式(1)を少し改変したものに, この学習後の結合荷重 $w_{ij}(n_{max})$ を用いて,

$$x_i(t) = f \left(\sum_{j=1}^N w_{ij}(n_{max}) x_j(t-1) \right) \quad (7)$$

とすれば, 適切な初期値を $\mathbf{x}(0)$ として与えることで, 先に学習した時系列を正しく読み出すこと(想起)が可能となる。

2.3 忘却機構の導入

次に, 忘却について考える。忘却とは, 簡単に言えば過去の出来事を忘れていくことであり, 日常生活においては, 概してない方がよいものと受け取られているようである。しかしながら, 忘却がなければ, 過去の出来事に関する情報をすべて保持し続けるということであり, 自らの特性を臨機応変とすることで環境の変化に対応するという考え方からは, 忘却は欠くべからざるものであり, むしろ歓迎すべきものと位置づけることが出来る。

さて, このような性質も有している忘却機構の導入についてであるが, 具体的には, 先に学習の際に示した式(5)の右辺各項に係数 k_1, k_2 を乗じて,

$$w_{ij}(n) = k_1 w_{ij}(n-1) + k_2 x_i(n) x_j(n-1) \quad (8)$$

と改めればよい。定性的には、この式が過去の結合荷重（第1項）に新しい関連付け（第2項）を順次加算していくものとみなせるため、

$$k_1 = 1 - k, \quad k_2 = k \quad (9)$$

のように各項の重み付けを定め、係数 k を0から1.0までの範囲で指定すれば、時間的に過去のものほど $(1-k)^n$ のオーダーで指数関数的に減小していく重み付けとなる。そのため、以下ではこの係数 k を忘却係数と呼ぶことにする。

ところで、この忘却機構は $k_1 + k_2 = 1.0$ としているため、数学的に至極明解で「忘れた程度に応じて新しい内容を覚える」と解釈できる。しかし換言すると、「忘れなければ覚えられない」あるいは「覚えるためには忘れなければならない」ということを意味し、現実には、思うように機能しない場合がある。そこで、式(9)を

$$k_1 = 1 - k, \quad k_2 = 1 \quad (10)$$

と改める。忘却係数 k は、上と同様に0から1.0までの範囲で指定する。この場合は「覚える内容は常に一定で、忘れる程度を調節する」となり、忘れる度合いとは独立に、新しく内容を覚える度合いを指定することが可能となる。また、忘却機構を有しないモデルと比較しても、結合荷重を一方的に増大させることなく、ある一定の有限の範囲内に収められることを保証している点で優れている⁽⁶⁾。

以下本研究では、この古城らの提案した忘却機構を導入した連想記憶型神経回路モデルを用いることにする。

3 計算機シミュレーション

ここで、本研究の目指すものを再度整理しておく。我々の身のまわりには、概周期的な振る舞いをする時系列情報があり、その中に潜む規則性を見出すことが可能である。このとき、最近の情報に含まれる系列をごく短期間にうちに記憶し、想起することが可能である。その一方で、過去の情報に含まれた系列は忘却によって消去することも可能である。

このような状況を模擬するにあたり、ここでは種類の異なる2つ時系列情報（#1, #2）を用意し、まず一方（#1）を一定期間呈示し、その後、他方（#2）を呈示する。このとき、まず最初の系列#1を記憶し、しかる後にこれを忘れて、次の系列#2を新たに記憶できるかについて検討する。なお、各系列は5つの状態

Table 1 学習用データを生成させる際の潜在的な規則

(a) 系列#1

		present					total
		$x^{[1]}$	$x^{[2]}$	$x^{[3]}$	$x^{[4]}$	$x^{[5]}$	
next	$x^{[1]}$	0	3	2	1	14	20
	$x^{[2]}$	14	0	3	2	1	20
	$x^{[3]}$	1	14	0	3	2	20
	$x^{[4]}$	2	1	14	0	3	20
	$x^{[5]}$	3	2	1	14	0	20
	total	20	20	20	20	20	100

(b) 系列#2

		present					total
		$x^{[1]}$	$x^{[2]}$	$x^{[3]}$	$x^{[4]}$	$x^{[5]}$	
next	$x^{[1]}$	0	2	1	14	3	20
	$x^{[2]}$	3	0	2	1	14	20
	$x^{[3]}$	14	3	0	2	1	20
	$x^{[4]}$	1	14	3	0	2	20
	$x^{[5]}$	2	1	14	3	0	20
	total	20	20	20	20	20	100

($P = 5$) から成るものとし、各状態は100個のニューロン ($N = 100$) から構成される相互結合型神経回路網における各ニューロンの発火 (+1) と静止 (-1) の組み合わせで定めるものとする。

以下では、種類の異なる2つ時系列情報（#1, #2）を用意する予備実験の過程と、実際の学習を通して記憶能力を検討する本実験の過程に分けて議論する。

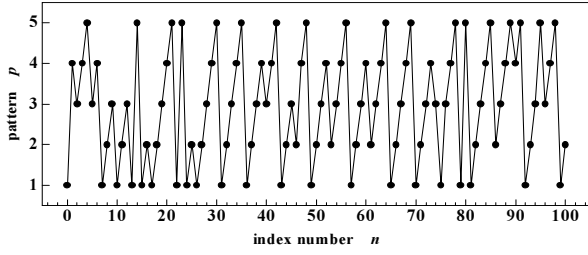
3.1 予備実験：学習用時系列データの作成

3.1.1 時系列データの生成方法

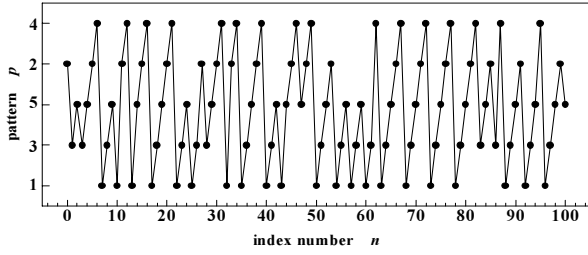
ここでは、学習用データとして潜在的な規則を含んだ時系列情報を2つ用意する。一口に“概周期的”な時系列と言っても、試行錯誤では潜在的な規則を有するものを作ることはできない。そこで、表1に示すような関係を状態遷移表という形で事前に定義し、これに記した確率に応じたデータ作成を行った。ちなみに、ここでは以下のような規則を想定している。

- 系列#1 : 1 → 2 → 3 → 4 → 5 → 1 → …
- 系列#2 : 1 → 3 → 5 → 2 → 4 → 1 → …

なお系列#2については、①系列#1の単なる逆順ではない、②系列#1と同一の状態遷移（例えば1 → 2）を含まない という2つの観点に留意して決定した。



(a) 系列#1



(b) 系列#2

Fig.1 生成された学習用時系列データ

実際に生成された時系列データを図1に示す。ここでは、縦軸方向に生成される系列の状態を順番[†]に割り振っているため、右上がりの三角状になればよい。また、表2が生成された時系列データの状態遷移表である。これを表1と見比べるとほぼ同一であり、大きな流れとしては上に示した傾向が認められるものの必ずしも周期的でなく、その中に揺らぎの存在する所望の性質を帯びた時系列情報となっていることが確認できる。

3.1.2 時系列データの学習可能性

さて次に問題となるのが、これら2つの時系列情報が実際に学習可能であるかの確認である。そこで、ここで用いるものと同様の構造、すなわち100個のニューロンから構成される相互結合型神経回路網を用意して、独立に学習させた場合について検討してみた。

このときの結果をまとめたものが表3である。これは、上で生成した各々の時系列データのうち最初の100点を取り上げて、10回の状態遷移（隣接する11状態の間の関連付け）、すなわち10回の学習ごとに系列想起が正しくできるかを確認したものである。ここでは、この学習回数に相当する情報を縦方向（列方向）に並べ、忘却係数をいろいろと変化させたときの傾向の移り変わりを横方向（行方向）に示している。それぞれの時刻 t でいずれの状態 p が実現されているかについては、オーバーラップ

[†]系列#2については、状態の番号を入れ替えて、下から順番に1, 3, 5, 2, 4としている。

Table 2 生成された学習用時系列データの状態遷移表

(a) 系列#1

		present					total
		$x^{[1]}$	$x^{[2]}$	$x^{[3]}$	$x^{[4]}$	$x^{[5]}$	
next	$x^{[1]}$	0	2	3	1	14	20
	$x^{[2]}$	16	0	1	2	1	20
	$x^{[3]}$	1	16	0	3	2	22
	$x^{[4]}$	1	1	17	0	1	20
	$x^{[5]}$	3	0	1	14	0	18
total		21	19	22	20	18	100

(b) 系列#2

		present					total
		$x^{[1]}$	$x^{[2]}$	$x^{[3]}$	$x^{[4]}$	$x^{[5]}$	
next	$x^{[1]}$	0	2	0	14	5	21
	$x^{[2]}$	2	0	1	0	15	18
	$x^{[3]}$	17	3	0	1	1	22
	$x^{[4]}$	0	13	2	0	1	16
	$x^{[5]}$	2	1	19	1	0	23
total		21	19	22	16	22	100

$$S_p(t) = \frac{1}{N} \sum_{i=1}^N x_i(t)x_i^{[p]} \quad (11)$$

を算出し、その最大値をとっている状態

$$p^*(t) = \arg \max_p S_p(t) \quad (12)$$

で決定した。なお、上述のオーバーラップ S_p とは、事前に記憶させた5つのパターン $\{x^{[p]} \mid p = 1, 2, 3, 4, 5\}$ との類似度を表している。ここでは簡単のため、結果については3つに大別して記述している。すなわち、

- 時間的な変化を伴わず、一定値を示したもの
【図2(a)参照】
- 時間的な変化を伴うものの、特定の状態が欠落した周期的なもの
【図2(b)参照】
- 正しく生成したもの（系列#1と系列#2に分けて表記）
【図2(c)参照】

である。このうち、3つ目の正しく生成したものについては、当該箇所影を付けて強調している。

さてこれらによれば、忘却係数 k の小さな方が、それぞれの系列を正しく記憶し、これを正しく想起できて

Table 3 生成された学習用時系列データを用いて学習したときの結果

(a) 系列#1

iteration t	forgetting coefficient k																								
	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.12	0.14	0.16	0.18	0.20	0.25	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
10	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
20	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
30	const	const	const	const	const	const	const	const	suc-1	const	const	const	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	repeat	const	const	const	const	const	const
40	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
50	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	const
60	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	repeat	repeat	repeat	repeat	repeat	repeat
70	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	repeat	repeat	repeat	repeat	repeat
80	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	suc-1	repeat	repeat	repeat	repeat
90	suc-1	suc-1	suc-1	suc-1	const	const	repeat	const	suc-1	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const
100	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	const	const	const	const	const

(b) 系列#2

iteration t	forgetting coefficient k																								
	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.12	0.14	0.16	0.18	0.20	0.25	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
10	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const
20	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	const	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	const	const	const	const
30	const	const	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const
40	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	repeat
50	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	
60	suc-2	suc-2	const	const	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	
70	suc-2	suc-2	suc-2	suc-2	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	
80	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	const	const	const	const	
90	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	
100	suc-2	suc-2	suc-2	suc-2	suc-2	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	

- const : 生成された時系列は時間的な変化を伴わず、一定値を示した。
- repeat : 生成された時系列は時間的な変化を伴うものの、特定の状態が欠落した周期的なものであった。
- suc-1 : 「系列#1」を正しく生成した。
- suc-2 : 「系列#2」を正しく生成した。

いる割合が大きい。これに対して、忘却係数 k が大きくなると、正しく想起できない割合が増してくる。これは忘却の度合いと密接に関わっており、いわゆる“忘れにくいもの”については、時系列データへ事前に埋め込んだ規則からの揺らぎがあっても十分に吸収できているのに対して、“忘れやすいもの”については、その影響を大きく受けてしまうということであろう。ちなみに、今回の実験結果を一瞥すると、 $k=0.04\sim 0.06$ のあたりが1つの境界となっているようである。

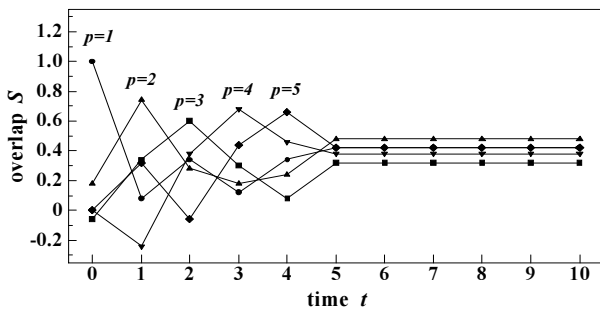
また、正しく想起できている影付きの部分が、“岬”のように左側から右側へ向かって数カ所伸びている。これは時系列データに含まれる状態遷移の中で、正しいものが直前に集中して学習が急速に進行したものの、正しくないものが直後に出現して学習が退行したためと考えられる。もちろん、これらのことは上述の忘却係数との関係で決まるのであるが、図1と比較すれば、この説明がほぼ妥当であることが確認できよう。

このように、ここでは異なる規則に従って2つの時系列データを生成し、それらがともに単独で学習が可能であることを確認した。以下の実験は、これら2つ

の時系列データを用いて行うこととする。

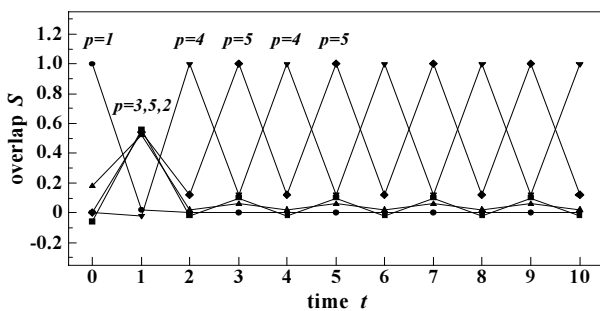
3.2 本実験：時系列データに潜む規則の記憶と忘却

これまでに、2種類の時系列データを用意し、単独ではあるものの、それらが相互結合型神経回路網で学習可能であることを確認した。そこで次に、本研究の最終的な目標でもある時系列データに潜む規則の記憶と忘却について検討する。まず最初に、3.1と同様にして、100個のニューロンから構成される相互結合型神経回路網に、系列#1の最初の100点を学習させる ($t=0\sim 100$)。そして、この後に系列#2を学習させる。系列#1の影響を消し去るには、少なくとも同等以上の学習回数が必要と考えられるため、ここでは200点の学習を行った ($t=100\sim 300$)。なお、これまでは特に断っていなかったが、図1からも明らかのように、系列#1については、 $p=1$ の状態からパターン生成を開始していたのに対して、系列#2については、 $p=2$ の状態からパターン生成を開始していた。これは、ここでの本実験において、状態遷移の際に不必要な変動を与えることを避けるため、系列#1の100番目の状態が「2」であることを見越して、系列#2の最初を定めておいた



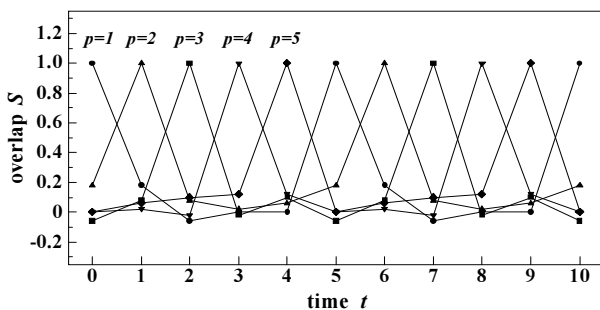
(a) 一定値を示したもの

— 系列#1, $k = 0.07, t = 90$ のとき —



(b) 特定の状態が欠落した周期的なもの

— 系列#1, $k = 0.20, t = 90$ のとき —



(c) 正しく生成したもの

— 系列#1, $k = 0.03, t = 90$ のとき —

Fig.2 生成した時系列データを学習した後の連想記憶モデルの代表的な応答例

のである。

このときの実験結果を表4に示す。上述の説明からも明らかなように、系列#1を学習している部分については表3(a)と同一である。また、その後に系列#2を学習する部分についても、概して表3(b)とほぼ同一の結果となっている。おもな相違点としては、忘却係数 k が小さい部分であるが、ここは「忘れにくい」という特徴を有している領域に相当し、系列#1の学習後、系列#2の数十回の学習によって系列#1の記憶は消失して

いるものの、なかなか系列#2に相当する内容を記憶することが出来なかったものと考えられる。なお事前の予想では、忘却係数 k が小さな部分は「覚えやすく忘れにくい」という特徴があると思っていたが、 $t = 200$ 以降においても新しい系列を記憶できないということは、新旧の学習用時系列データの単なる影響の度合いの大小だけでなく、忘却機構の有無そのものが大きな意味を持っているのかもしれない。現段階では単なる憶測に過ぎないが、2.3でも言及したように、忘却機構がなければ結合荷重は増大する一方であり、このことが何らかの障害となっているとも考えられる。

また、忘却係数 k が大きな部分は「覚えやすいが忘れやすくもある」という特徴を有していると思っていた。確かに、極端に大きな値のときは与えられた時系列を記憶することができず、忘却に関する議論がそもそも成り立たないが、それにしても、 $k = 0.25$ 付近でも条件付きではあるが、時系列の記憶と想起を実現できている場合のあることは意外であった。

今回の計算機シミュレーションに関して言えば、学習に用いた時系列データやそれを構成する状態数、あるいは埋め込んだ潜在的な規則など、かなりの制約条件があつて極めて限定的な議論となっているが、総じて忘却係数は $k=0.03\sim 0.04$ のあたりがよいように見受けられる。

4 考察

我々の身のまわりを見渡すと、過去の行動パターンを分析し、そこに含まれる傾向を自動的に発見して、将来の振る舞いを“察してくれる”ようなシステムがあればよいと感じる場面がある。例えば、音楽プレーヤにおいて、過去の選曲履歴から利用者の好みに応じた曲順の自動再生をしたり、ワープロにおいて、過去のキー入力から利用者のよく用いる単語候補を列挙するなどである。現在のところ、上述のシミュレーション結果からも明らかなように、①最初の系列を記憶する部分、②最初の系列を忘れて次の系列を記憶するまでの部分の2点について、それらに要する時間の長いことが最大の課題である。もし何らかの方法でこれを短縮することが可能であれば、この種の潜在的需要も大きいように思えるため、工学的な応用が図れるのではないかと期待を抱いている。

また、今回のように概周期的な時系列情報の中から潜在的な規則を発見して記憶し、もしその規則が変わったと判断したら、新しいものを覚えるように努め、古いものは徐々に忘れるようにするという一連の流れは、

Table 4 計算機シミュレーションの結果

iteration <i>t</i>	forgetting coefficient <i>k</i>																								
	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.12	0.14	0.16	0.18	0.20	0.25	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
10	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
20	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
30	const	const	const	const	const	const	const	suc-1	const	const	const	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	repeat	const	const	const	const	const	const	const
40	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
50	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	const
60	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	repeat	repeat	repeat	repeat	repeat
70	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	repeat	repeat	repeat	repeat
80	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	suc-1	repeat	repeat	repeat	repeat
90	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	suc-1	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat
100	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	suc-1	const	const	const	const	const	const
110	suc-1	suc-1	suc-1	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const
120	suc-1	const	const	const	const	const	const	const	const	const	const	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	const	const	const	const	
130	suc-1	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	
140	const	const	repeat	const	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	
150	const	const	const	const	const	const	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	
160	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	
170	const	repeat	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	
180	const	const	repeat	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	const	const	const	const	const	
190	const	repeat	repeat	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	const	const	const	const	
200	const	const	suc-2	suc-2	suc-2	suc-2	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	
210	const	const	suc-2	suc-2	suc-2	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	
220	const	const	suc-2	const	suc-2	suc-2	suc-2	suc-2	const	const	const	const	const	const	const	const	const	const	repeat	const	const	const	const	const	
230	const	const	suc-2	suc-2	suc-2	const	const	const	const	const	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	
240	const	suc-2	suc-2	suc-2	suc-2	const	const	const	suc-2	const	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	const	const	const	
250	const	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	
260	repeat	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	
270	repeat	const	const	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	
280	repeat	const	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	const	
290	const	suc-2	repeat	repeat	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	const	
300	const	suc-2	suc-2	suc-2	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	repeat	const	

const : 生成された時系列は時間的な変化を伴わず、一定値を示した。
 repeat : 生成された時系列は時間的な変化を伴うものの、特定の状態が欠落した周期的なものであった。
 suc-1 : 「系列#1」を正しく生成した。
 suc-2 : 「系列#2」を正しく生成した。

我々自身が日々体験していることとよく似ている。例えば、当初の規則を破棄して別の規則を採用するまでには、我々も逡巡する期間があるのは事実であり、今回の計算機シミュレーションの結果は、このような状態を再現しているようにも受け取れる。ヒトの記憶メカニズムについては未だ不明なところも多く、このような実験結果を参考にして、脳の仕組みに迫ればとも考えている。

5 おわりに

今回は、概周期的な時系列情報の中に潜む何らかの規則を、学習によって見出す連想記憶型神経回路モデルについて検討した。その結果、最近の情報に含まれる系列をごく短期間にうちに記憶し、過去の情報に含まれた系列は忘却によって消去できる可能性を、計算機シミュレーションを通して明らかにした。今後は、ここで得られた示唆を詳細に検討するとともに、更なる能力向上を目指す予定である。

参考文献

- [1] 和久屋 寛, 福江慈孝: “時系列情報に含まれる潜在的規則を抽出する連想記憶型神経回路モデル”, 第7回日本知能情報ファジィ学会九州支部学術講演会予稿集, B303, pp.97-98, 2005
- [2] 福江慈孝, 和久屋 寛: “忘却機構を有する連想記憶モデルによる時系列中の潜在的規則の抽出と自動削除”, 第2回人工頭脳工学シンポジウム講演論文集, O-2, pp.16-17, 2006
- [3] K.Nakano: “Associatron —A model of associative memory”, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.SMC-2, pp.380-388, 1972
- [4] T.Kohonen: “Correlation matrix memories”, *IEEE Transactions on Computers*, Vol.C-21, pp.353-359, 1972
- [5] S.Amari: “Learning patterns and pattern sequences by self-organizing nets of threshold elements”, *IEEE Transactions on Computers*, Vol.C-21, pp.1197-1206, 1972
- [6] 古城 玄, 和久屋 寛, 信太克規: “忘却機構を有する連想記憶モデル”, 第9回電子情報通信学会九州支部学生会講演会講演論文集, D-37, p.98, 2001