

THE TRANSACTIONS OF THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS D-II

EIC | **電子情報通信学会**
D-II | **論文誌** VOL.J78-D-II NO.5 **MAY**
情報・システムⅡ—情報処理 ————1995

情報・システムソサイエティ

社団法人 **電子情報通信学会**

THE INFORMATION AND SYSTEMS SOCIETY

THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS

GA を用いたリカレントニューラルネットワーク共進化モデルの学習

准員 佐藤 裕二[†] 准員 古谷 立美^{††}

Co-Evolution in Recurrent Neural Networks Using Genetic Algorithms

Yuji SATO[†] and Tatsumi FURUYA^{††}, Associate Members

あらまし 本論文では、リカレント型ニューラルネットを基本とした先読みモデルを提案し、その有効性を検討する。先読みモデルは、行動を決定するネットワークと環境の内部モデルからなり、それぞれリカレント型ニューラルネットで構成する。環境の内部モデルを学習により構築しながら先読みを行い、投機的な行動を実際に環境に対して行う。また、先読み結果を、行動を決定するネットワークの学習に利用する。すなわち、二つのネットワークは、共進化を行う。ニューラルネットワークの構造およびウエイトの学習には、遺伝的アルゴリズムを適用した。tic-tac-toe ゲームを用いて有効性の評価を行い、以下の結論を得た。(1)環境の内部モデルを用いた先読みを行うことにより、実際の環境を介した学習回数の削減につながる可能性をもつ。(2)ある程度学習が進んでから、内部モデルを用いた先読みを行うべきである。(3)内部モデルの学習方式として、ニューラルネットワークを用いてアルゴリズムを学習する方式の方が、入出力の対応表を Classifier System を用いて学習する方式よりも、先読み効果が高い可能性をもつ。

キーワード 進化的計算, 人工生命, 遺伝的アルゴリズム, リカレントニューラルネットワーク, 共進化

1. まえがき

今日、コンピュータは、解法のためのアルゴリズムが存在しプログラム言語で明確に記述可能な良設定問題を解く能力では、人間をはるかに上回る。一方、解くべき多くの問題が不良設定でありアルゴリズムとして記述困難な実世界における情報処理能力に関しては、人間と比べて依然として柔軟ではない。この問題を解決する一つの可能性として、適応システムや機械学習の研究が考えられる。しかしながら、実世界における多くの問題では、適応システムや機械学習で必要となる試行回数や学習時間が十分与えられるとは限らない。十分な試行回数を得る前に何らかの行動を必要とされる場合が考えられる。すなわち、学習過程における部分的な情報をもとに有効な行動を行う必要がある。

この問題の解決策の一つとして、Holland⁽¹⁾ や Riolo⁽²⁾ は Classifier System (CS)⁽³⁾ を用いた先読みの

利用を提案している。しかしながら、CS では、入力と出力の対応表という形で内部モデルを構成するために、静的環境における先読みが基本となり、学習過程における深い先読み(数手先まで読むこと)や動的に変化する問題を扱うことには適さない。なぜなら、CS では、未学習の入力データに対する適応能力(すなわち、出力の信頼度)が学習過程で向上しないからである。

この問題を解決するために、本論文では、リカレント型のニューラルネットワークを基本とした先読みモデルを提案し、その有効性を検討する。すなわち、行動を決定するネットワークと環境の内部モデルをそれぞれリカレント型のニューラルネットワークで構成し、学習過程における先読みを有効に活用することを考えた。リカレント型ニューラルネットワークは、非線形関数の学習を行う、規則的な処理の学習からカオスダイナミックスの学習⁽⁴⁾まで幅広い適応能力をもつ、外界を介したフィードバックループを用いた学習が可能である等、適応・進化システムに有効と思われる性質を有するからである。以下、2. では、提案する先読みモデルの概要を示す。3. では、遺伝的アルゴリズム(GA)⁽⁵⁾を用いたニューラルネットワークの構造および

[†] 新情報処理開発機構, つくば市

Real World Computing Partnership, Tsukuba-shi, 305 Japan

^{††} 東邦大学理学部情報科学科, 船橋市

Faculty of Science, Toho University, Funabashi-shi, 274 Japan

結合荷重の学習方法を示す。4.において、tic tac toe ゲームを用いて、このモデルの有効性の評価を行い⁽⁶⁾、最後に考察とまとめを行う。

2. 先読みモデルの提案

2.1 基本とするリカレントネットワークの特徴

本論文で扱うリカレント型ニューラルネットワークは、自己帰還ループのほかに、環境(外界)を介した帰還ループをもつことを特徴とする。図1は、例えばフィードフォワード型のニューラルネットワークに時間遅れのある自己帰還ループを加えた Jordan 型の時系列発生モデル⁽⁷⁾(図中の点線で囲んだ部分)に、環境を介した帰還ループを付加した場合の例である。図1のニューラルネットワーク部が N 個のニューロンをもつ単一層から構成される場合には、離散時間モデルの動作方程式は以下の式で表すことができる⁽⁸⁾。

$$X_i(t+1) = f \left[\sum_{j=1}^n w_{ij} Y_j(t) + \sum_{j=n+1}^{n+N} w_{ij} X_{(j-n)}(t) \right] \quad (1)$$

$$Y_j(t) = F[X_1(t), X_2(t), \dots, X_N(t)] \equiv F[X(t)] \quad (2)$$

ここで、 $X_i(t)$ 、 $Y_j(t)$ はそれぞれ、 i 番目のユニットの時刻 t での出力、外部入力を表し、 w_{ij} は j 番目のユニットから i 番目のユニットへの結合荷重を表す。また、 f はシグモイド関数、しきい関数などの適当な関数、 F は環境からの影響を表す未知の関数である。すなわち、時刻 $t+1$ の出力 $X(t+1)$ は、時刻 t の出力 $X(t)$ および時刻 t の出力に対する環境(外界)からの応答 $Y(t) = F[X(t)]$ から決定される。

2.2 行動を決定するネットワークと内部モデルの共進化

本論文で提案するモデルは、先読みを実現するために、行動を決定するネットワーク(Action Network)のほかに、環境の内部モデル(Internal Model)をもつ。いずれも、前記リカレント型のニューラルネットワークを基本とする。内部モデルも含めた基本的構成を図2に示す。行動を決定するネットワークと内部モデルは相互結合関係にある。内部モデルも N' 個のニューロンをもつ単一層から構成されると仮定すると、先読みモデルの動作方程式は以下の式で表すことができる。

$$X_i(t+1) = f \left[\sum_{j=1}^n w_{ij} Y_j(t) + \sum_{j=n+1}^{n+N'} w_{ij} Y'_{(j-n)}(t) + \sum_{j=n+N'+1}^{n+N'+N} w_{ij} X_{(j-n-N')}(t) \right] \quad (3)$$

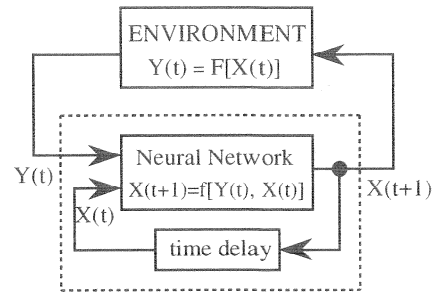


図1 基本とするリカレント型ニューラルネットワークの例
Fig. 1 The basic recurrent neural network.

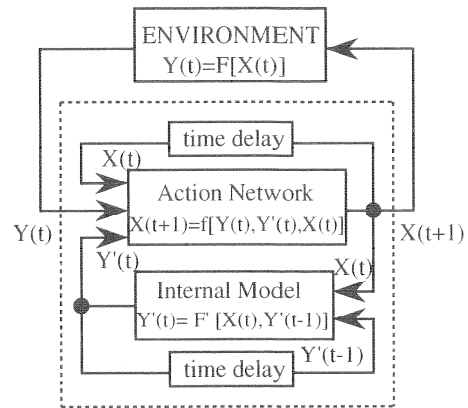


図2 提案する先読みモデル
Fig. 2 The basic lookahead model of the recurrent neural network.

$$Y'_j(t) = F' \left[\sum_{k=1}^N w'_{jk} X_k(t) + \sum_{k=N+1}^{N+N'} w'_{jk} Y'_{(k-N)}(t-1) \right] \quad (4)$$

ここで、 $Y'_j(t)$ 、 $Y_j(t)$ はそれぞれ、内部モデルを構成する j 番目のユニットの時刻 t での出力、および式(2)で定義した外部入力である。 F' はシグモイド関数、しきい関数などの適当な関数を表す。提案した先読みモデルでは、環境との応答を通して、環境の内部モデルを自己学習しながら先読みを利用する。また、先読み結果に従い、投機的な行動を(実際に外界に対して)行う。先読み結果は、行動を決定するネットワークの学習にも利用する。すなわち、本論文で提案する先読みモデルは、行動を決定するネットワークと内部モデルが相互作用をもちながら成長するモデルとなっている。

3. GA によるニューラルネットワークの構造および結合荷重の決定

ここでは、GA によるニューラルネットワークの構造および結合荷重の決定を内部モデルの合成に適用する。

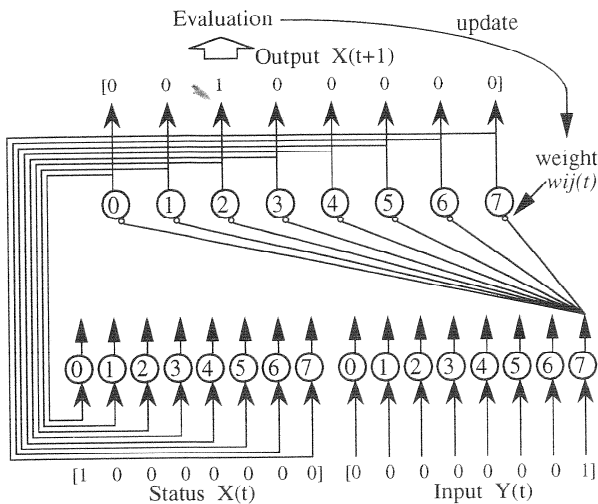


図3 行動決定ネットワークの学習方式の例
Fig. 3 Example of learning method in action network.

これは、GA がもつグローバルサンプリングの特徴を生かして、局所解に落ち込むことを防ぐことをねらったためである。GA を用いたニューラルネットワークの学習に関しては、既にいくつかの研究報告がされている^{(9)~(11)}。以下、行動を決定するネットワークの学習方式および内部モデルの学習方式について示す。

3.1 行動を決定するネットワークの学習

行動を決定するネットワークでは、時刻 t の出力 (状態) $X(t)$ 、時刻 t の出力に対する環境からの応答 (入力) $Y(t)$ と時刻 $t+1$ の出力 (行動) $X(t+1)$ の関係を学習する。今回実験に使用したネットワーク構造を図3に示す。

図3では、前章で説明したりカレントネットワークを空間的に展開した、2層構造を用いて学習を行っている。入力層16ニューロン、出力層8ニューロンの合計24ニューロンから構成される。入力層の中で、8ニューロンは状態記憶に用い、残り8ニューロンは入力信号の記憶に用いる。出力層のニューロンはしきい関数により入出力変換を行う。出力 $X(t+1)$ は、基本的には、前章の式(1)に従う。出力信号が“1”となるニューロンの番号が行動に対応する。但し、出力層を構成するニューロンの出力が複数個“1”となる場合には、以下の式により、あるニューロン番号を1個だけ選択する。例えば、8個のニューロンのすべての出力が“1”の場合、 i 番目のニューロン番号が選択される確率 P_i は、

$$P_i = W_i / \sum_{i=0}^7 W_i \quad (5a)$$

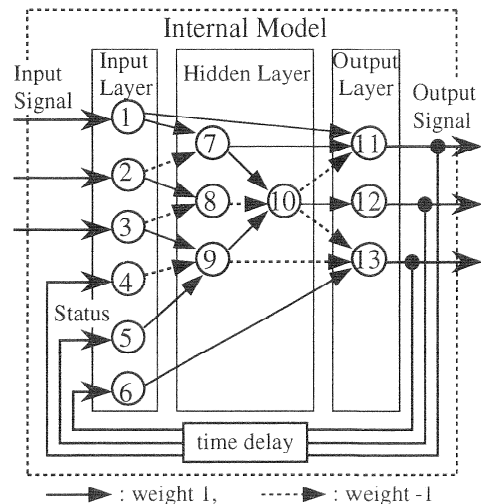


図4 ニューラルネットワークを用いた内部モデルの例
Fig. 4 Example of the internal model which is incorporated into the recurrent neural network.

$$W_i = \left[\sum_{j=0}^7 w_{ij} Y_j(t) + \sum_{k=0}^7 w_{ik} X_k(t) \right] \quad (5b)$$

と定義する。ネットワークの初期状態は、4.2.1で述べるように、明示的に与える。

学習は2種類の方法で行う。第1は、内部モデルを用いた先読み結果から予測される勝敗による評価を行い、選択したニューロン番号に対応する重み値 w_{ij} の増減を行う。第2は、あるニューロン番号を選択したことにより実際にゲームに勝ったか、負けたかという評価を行い、選択したニューロンに対応した重み値 w_{ij} の増減を行う。すなわち、実際のゲームの勝敗および先読み結果の勝敗を強化信号とする、強化学習を行う。

3.2 内部モデルの学習

ここでは、2種類の方式に関して検討を行った。いずれも、前章で説明したりカレントネットワークを空間的に展開した多層構造を用いて学習を行う。

3.2.1 学習方式1：ニューラルネットワーク方式

まず、学習方式1で扱うニューラルネットワークの条件を示す。一例を図4に示す。○印中を通し番号を付けたシンボルがニューロンを表す。

(i) 各ニューロンは、自分自身の状態として0(非励起状態)または1(励起状態)のいずれかをとる。ある時点における入力信号の総和が0よりも大きい場合は自分自身の次の状態を1にし、それ以外の場合は次の状態を0にする。

(ii) 回路網は、入力層、隠れ層群、出力層から構成される層状結合型のニューラルネットワークである。入力層は、環境からの入力信号および自己帰還信号が

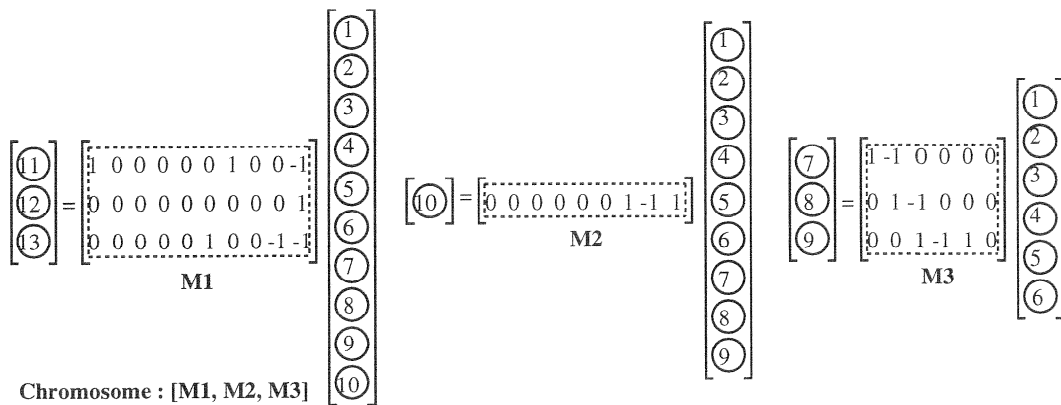


図5 図4のネットワークに対応した遷移行列
 Fig. 5 Transition matrices for the connections shown in the network of Fig. 4. (Learning method 1)

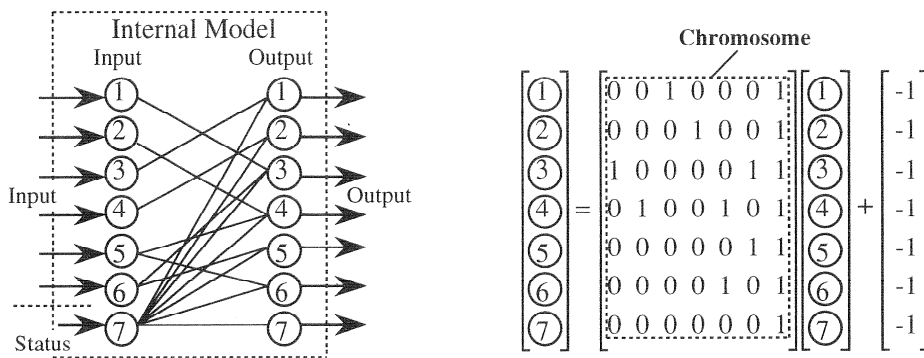


図6 入出力対応表を学習する内部モデルの例と対応する遷移行列
 Fig. 6 Example of the internal model which is memorizing input-output correspondences. (Learning method 2)

ら構成される。

(iii) ニューロン間の結合荷重は1, -1または0のいずれかをとる。0のときは、ニューロン間に結合がないことを表す。

次に、GAを適用するために、回路網の構造と結合荷重を表す染色体を定義する。ここでは、図5に示すように、各ニューロン間の遷移状態を表す行列 $M1$, $M2$, $M3$ を各層ごとに考え、これらの遷移行列から構成される列 $[M1, M2, M3]$ を染色体とする。行列の各要素は、あるニューロンから別のニューロンへの結合荷重 (1, -1または0) を表している。一般に、 m 層からなるネットワークでは、遷移行列は全部で $m-1$ 個存在する。この $m-1$ 個の遷移行列から構成される列を一つの染色体と考える。GAでは、通常、1次元の染色体を定義し遺伝オペレータの操作を行う。しかし、1次元の染色体では、ニューラルネットワーク上のある2次元部分構造が非常に有効だとしても、その構造が染色体上では

分散されてしまい、遺伝オペレータにより壊されてしまう確率が高い。そこで、各層ごとの2次元遷移行列から構成される列を染色体として定義した。

3.2.2 学習方式2：入出力の対応表を学習する方式

学習方式2では、入力から出力への2次元遷移行列を染色体と考える。入力または出力としてとり得る状態をそれぞれ異なるユニットに割り当てることにより、CSと同値になる。図6に一例を示す。図6は、時刻 t のときの出力“7”に対する“1”~“6”の入力(応答)に対して、それぞれ“1”~“7”のいずれを出力するかの対応表を学習する場合の例である。

4. 実験

4.1 実験方法

4.1.1 ゲームの設定

tic-tac-toeゲームを用いてモデルの評価を行った。ルールは以下のとおりである。

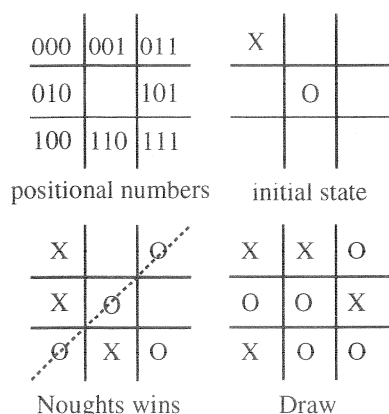


図7 tic-tac-toe ゲームの説明図
Fig. 7 Illustration of tic-tac-toe game.

(ルール) 2人が交互に自分のシンボル、すなわち○とX、を3行3列のマトリックスの中に置いていく。縦、横、斜めのいずれかの方向に、先に自分のシンボルを三つ連続して置いた方が勝ちである。

遺伝オペレータを適用するために、図7に示すように、中央を除く3行3列のマトリックス上の各位置に、“0”から“7”(すなわち、“000”から“111”)までの位置番号を設定した上で、解析を簡単化するために、以下の仮定のもとに評価を行った。

- (1) 先手は、図7に示すように、自分のシンボル○を最初に中央に置く。後手は、自分のシンボルXを、最初に位置“0”(“000”)に置く。
- (2) 先手は、熟練者であり、負けることはない。
- (3) 後手(シンボルX)は、先手のアルゴリズムを知らない。
- (4) 後手は、初期状態として、上記ゲームのルールから以下の知識をあらかじめ得ている。

- (知識1) 自分の番のときに、自分のシンボルを3個続けて一直線上に置ければ置く。
- (知識2) (知識1)が適用できないとき、もし次に相手が、相手のシンボルを3個続けて一直線上に置けるようなら阻止する。
- (知識3) (知識1), (知識2)ともに適用できない場合は、空いている場所にランダムに打つ戦略をとる。

以上の仮定のもとに、先手を環境と考え、後手に対して図2に示した先読みモデルを適用した。すなわち、先手には負けることのないアルゴリズムを知識ベースとして与える。後手は、(知識1)~(知識3)を行動を決定するネットワークの初期状態として、ゲームの試行を通して、先手のアルゴリズムを内部モデルとして学習しながら、行動を決定するネットワークのうちの(知

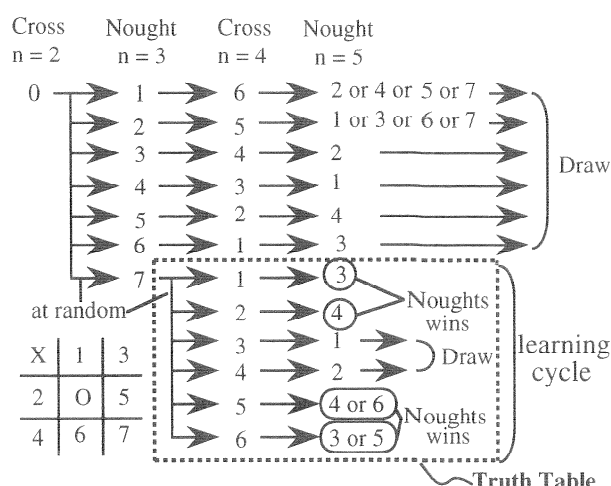


図8 (知識1)~(知識3)に基づくtic-tac-toeゲームの流れ
Fig. 8 Tic-tac-toe game flow chart based on knowledge of rules 1 to 3.

識3)の戦略のみ更新する。

4.1.2 ゲームの流れと教師データ

上記仮定に基づくゲームの流れを図8に示す。第3手目に先手は、空いている場所の中から、ランダムに選択する設定とした。第3手目に先手が“1”~“6”の位置に自分のシンボルを置いた場合、ゲームの結果は引き分けとなる。すなわち、後手の行動を決定するネットワークの更新は行われない。第3手目に先手が“7”の位置に自分のシンボルを置いた場合のみ勝敗が存在し、行動を決定するネットワークの中の(知識3)の更新が行われる。すなわち、行動を決定するネットワークと内部モデルは、それぞれ以下の学習を行う。行動を決定するネットワークは、3.1で示した学習方法に従い、“7”の手に対して“3”または“4”の位置に自分のシンボルを打つことを学習する。内部モデルは、先手の“7”の手に対する自分(後手)の“1”~“6”の応答に対して、先手が次に返してくる手(すなわち、点線内の真理値表)を学習する。ここで、“5”と“6”に対する先手の次の手はそれぞれ2種類存在し、ゲームごとにランダムに変化するアルゴリズムに設定した。すなわち、内部モデルの学習ターゲットとなる先手のアルゴリズムには不確実性をもたせる設定とした。

4.2 ネットワークの初期設定とGAの適用方法

4.2.1 行動を決定するネットワークの初期設定

図3に示した、入力層16ニューロン、出力層8ニューロンから構成される2層構造を用いて実験を行った。初期状態は、(知識1), (知識2)および(知識3)

の初期戦略に対応した構造を明示的に与えた。すなわち、状態“0”(2手目)のとき、入力“1”~“6”(3手目)に対して出力“6”~“1”(4手目)、入力“7”(3手目)に対して出力(4手目)はランダムに選択するという対応(図8に記述)を初期状態として、状態“0”，入力“7”と出力“1”~“6”の間の重み値 w_{ij} を、3.1に示す学習方式で更新した。入力“1”~“6”と出力の間の重み値情報((知識1)と(知識2)に対応)はゲームの間更新されない。

4.2.2 内部モデル獲得のためのGAの適用方法

(a) 学習方式1の場合

まず適応度について示す。適応度は、内部モデルを構成するニューラルネットワークが与えられた問題にどの程度適しているかを示す必要がある。すなわち、ある入力に対してどの程度正しい出力が得られるかを示す必要がある。ある程度十分な試行回数を得られることを前提とした学習の場合は、数十~数百回の試行のうち何回出力値が期待値と一致したかを適応度と定義することも考えられる。しかし、ここでは十分な試行回数を得られない場合の学習問題を対象としているため、1回のゲームの試行において、予測した手(すなわち、内部モデルを構成するニューラルネットワークの出力)と実際の相手の手(すなわち、先手の応答)が何ビット一致していたかを適応度とした。一致していたビット数を n_c とすると、個体 I_i の適応度 $f(I_i)$ は次式で定義される。

$$f(I_i) = n_c \quad (6)$$

次に遺伝オペレータの設定について示す。ある世代の個体群から次世代の個体群を生成するための方法として、適応度の下位の一定の割合を淘汰し、ルーレット選択による子孫を補って個体総数を一定に保った。すなわち、淘汰される個体の数を M 個とすると今の世代の N 個の個体 $I_1 \sim I_N$ から二つの個体のペアを M 組だけ重複を許してランダムに選択し、それぞれに対して交差と突然変異を施して不足を補うための M 個の個体を決定した。ここで、ある個体 I_i が次の世代の個体を作るための親として選択される確率 $P(I_i)$ は、次式によって定義した。

$$P(I_i) = f(I_i) / \left\{ \sum_{j=1}^N f(I_j) / N \right\} \quad (7)$$

交差は、染色体を構成する遷移行列中のランダムに選んだある行を、ペアとして選んだ個体の遷移行列の対応する行と置き換える方式とした。行の置換えを中心とした交差を行った理由は、学習途中でできたニュー

ラルネットワーク上の有効な2次元部分構造を交差により破壊する確率を抑えるためである⁽¹²⁾。学習方式1では、6層、22ニューロン構造(入力層から順に、6-7-2-2-2-3ニューロン)を初期状態とした。状態、入力、出力をそれぞれ3ビットで表現し、 $f(I_i)=1$ 以下の個体を淘汰した。個体数 N は100、突然変異率は0.01に設定した。上記遺伝的操作は、すべての個体の適応度が2以上になるまで、繰り返し行った。

(b) 学習方式2の場合

学習方式2における適応度および遺伝オペレータの設定は基本的に上記学習方式1と同一に設定した。但し、図6に示すように、学習方式2では入力7ユニット、出力7ユニットで構成した上で、とり得る手をそれぞれ異なるユニットに対応させている。従って、状態1ビット、入力6ビット、出力7ビットで表現し、 $f(I_i)=5$ 以下の個体を淘汰した。交差は、学習方式1との対応から、遷移行列中のランダムに選んだある行をペアとして選んだ個体の遷移行列の対応する行と置き換える方式とした。上記遺伝的操作は、すべての個体の適応度が6以上になるまで、繰り返し行った。

4.3 実験結果

以下の四つのケースに関して実験を行った。

(実験1) (知識3)の学習による更新を、先読みなしで、行う場合。内部モデルは使用しない。

(実験2) (知識3)の学習による更新を、先読みも利用して、行う場合。但し、内部モデルの学習は前章で述べた、学習方式1を用いる。

(実験3) (知識3)の学習による更新を、先読みも利用して、行う場合。但し、内部モデルの学習は前章で述べた、学習方式2を用いる。

(実験4) (知識3)の学習による更新を、先読みも利用して、行う場合。但し、内部モデルとして、先手(シンボル○)の完全なアルゴリズムを最初から使用する。

ゲームの試行回数と後手がそれまでに負けた試合総数の割合との関係を、いくつか乱数の初期値を変えて求めた。結果を、図9から図11に示す。

5. 考察

実験結果から、以下のことが考えられる。(実験1)と他の三つの実験結果の比較から、今回扱った問題に対しては、内部モデルを用いた先読みが勝率向上に有効に作用していたと考えられる。内部モデルを用いた先読み結果を学習に利用することにより、学習に必要な試行回数を削減できる可能性をもつ。(実験2)と(実験

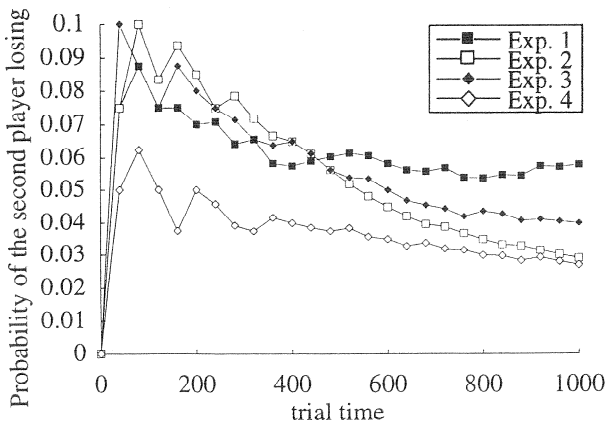


図9 後手が負ける確率(累積平均)と試合数の関係
Fig. 9 Probability of second player losing as a function of the number of trial games played. (seed=7301)

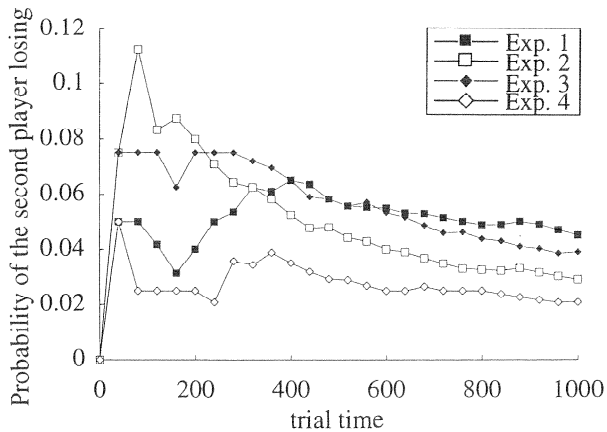


図10 後手が負ける確率(累積平均)と試合数の関係
Fig. 10 Probability of second player losing as a function of the number of trial games played. (seed=94417)

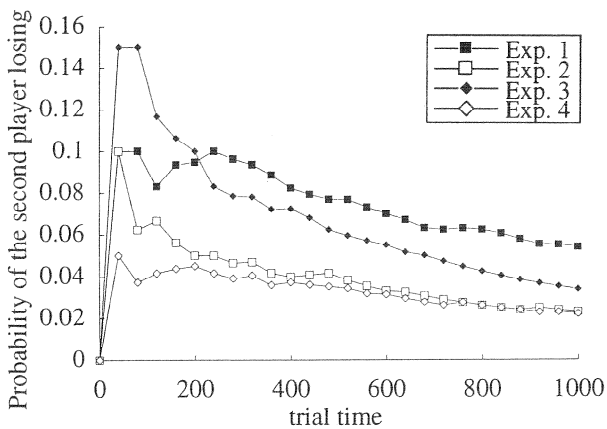


図11 後手が負ける確率(累積平均)と試合数の関係
Fig. 11 Probability of second player losing as a function of the number of trial games played. (seed=1994629)

3) では、学習開始直後は先読み効果にばらつきがあるが、試行回数が進むにつれて安定して高い効果を維持する。一方、(実験4)では学習開始直後すぐに高い先読み効果を示している。これは、内部モデルの学習が進み、先読みの信頼性が向上して初めて安定した効果が得られるためと推測される。従って、工学的な応用を考える場合は、内部モデルの学習進行状況をチェックし、途中から先読み結果を学習に利用するように切り換える方式が有効である。(実験2)と(実験3)では、(実験2)の方が高い先読み効果を示す傾向にあった。すなわち、多層型ニューラルネットワークを用いてアルゴリズムを学習するタイプの学習方式1が、CSを用いて入出力の対応表を学習するタイプの学習方式2よりも、高い先読み効果を示す傾向にあった。この原因を調査するために、内部モデルの学習ターゲットの詳細な調査、および、(実験2)と(実験3)で得られた内部モデルの解析を行った。表1は、学習ターゲットとなる真理値表の入/出力をそれぞれ3ビットで表した例である。表1から、内部モデルは以下のブール式を学習することになる。

(1) “5”→“4”、“6”→“3” のとき

$$y_1 = \overline{x_1}x_2x_3 + x_1\overline{x_2}x_3, \quad y_2 = \overline{x_1}x_2x_3 + x_1x_3$$

$$y_3 = \overline{x_1}x_3 + x_2(x_1 \oplus x_3)$$

(2) “5”→“4”、“6”→“5” のとき

$$y_1 = x_2x_3 + x_1(x_2 \oplus x_3), \quad y_2 = \overline{x_2}(x_1 \oplus x_3)$$

$$y_3 = \overline{x_1}x_3 + x_2(x_1 \oplus x_3)$$

(3) “5”→“6”、“6”→“3” のとき

$$y_1 = \overline{x_1}x_2x_3 + x_1\overline{x_2}x_3, \quad y_2 = \overline{x_1}x_2 + \overline{x_2}x_3 + x_1x_3$$

$$y_3 = \overline{x_1}x_3 + x_2(x_1 \oplus x_3)$$

(4) “5”→“6”、“6”→“5” のとき

$$y_1 = x_2x_3 + x_1(x_2 \oplus x_3), \quad y_2 = \overline{x_2}(x_1 + x_3)$$

表1 内部モデルの真理値表
Table 1 Truth table of the internal model.

x ₁	x ₂	x ₃	y ₁	y ₂	y ₃
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	0	0	1
1	0	0	0	1	0
1	0	1	1	0	0
			1	1	0
1	1	0	0	1	1
			1	0	1

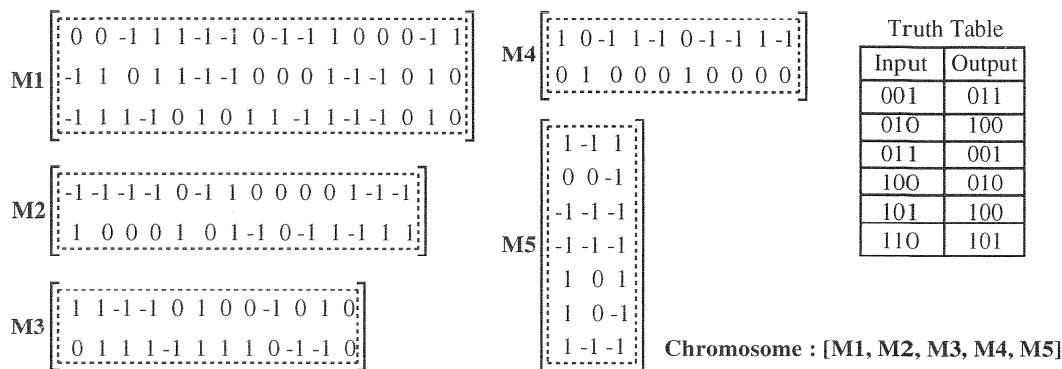


図12 ニューラルネットワークを用いた内部モデルで得られた遷移行列の例(簡略化のため、自己帰還ループの情報は省略している)
 Fig. 12 Example of the chromosome carried out using learning method 1. Networks connected with a self-feedback loop were omitted.

$$y_3 = \overline{x_1 x_3} + x_2(x_1 \oplus x_3)$$

すなわち、内部モデルの学習ターゲットは排他的論理和(XOR)を部分的に含んでおり、内部モデルは、ターゲットとなる真理値表のランダムな変化に伴い、XOR論理の生成と消滅を動的に繰り返す必要があったことがわかる。図12に(実験2)で得られた内部モデルの一例を示す。学習方式1で得られた内部モデルは、XOR論理(すなわち、GA-deceptiveな論理⁽⁵⁾)を確実に獲得していることがわかる。このことから、動的に変化する非線形論理の学習には、入出力の対応表をCSで学習する方式よりも、ニューラルネットを用いて学習する方式の方が有効であると推測される。

6. むすび

本論文では、リカレント型ニューラルネットワークを基本とした先読みモデルを提案し、その有効性を検討した。行動を決定するネットワークと環境の内部モデルをそれぞれリカレント型ニューラルネットワークで構成し、内部モデルを学習により構築しながら先読みに利用する。先読み結果は、行動を決定するネットワークの学習回数の削減に活用する。tic-tac-toeゲームを用いて有効性の評価を行い、以下の結論を得た。(1)環境の内部モデルを用いた先読みを行うことにより、実際の環境を介した学習回数の削減につながる可能性をもつ。(2)ある程度学習が進んでから、内部モデルを用いた先読みを行う方が効率的である。(3)今回の実験結果では、内部モデルの学習方式として、ニューラルネットワークを用いてアルゴリズムを学習する場合の方が、CSにより入出力の対応表を学習する場合よりも、

先読みの効果が高かった。今後、より探索空間が広く、深い先読みを必要とする大規模問題での調査が必要である。

文 献

- (1) Holland J. H.: "Concerning the Emergence of Tag-Mediated Lookahead in Classifier Systems", Emergent Computation, pp. 188-201, MIT Press (1991).
- (2) Riolo R. L.: "Lookahead Planning and Latent Learning in a Classifier System", From animals to animats, pp. 316-326, MIT Press (1991).
- (3) Holland J. H.: "Adaptation in Natural and Artificial Systems", pp. 172-181, MIT Press (1992).
- (4) Sato M., Murakami Y. and Joe K.: "Learning chaotic dynamics by recurrent neural networks", Proc. Inter. Conf. on Fuzzy Logic & N. N. pp. 601-605 (1990).
- (5) Goldberg D. E.: "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley (1989).
- (6) Sato Y., Hatano S., Hatano H. and Furuya T.: "Lookahead Planning and Co-Evolution in Recurrent Neural Networks", Proc. of ICEC '94-Orlando, pp. 764-769 (1994).
- (7) Jordan M. I.: "Serial order: A parallel distributed processing approach", Advances in Connectionist Theory: Speech, Erlbaum (1989).
- (8) Hecht-Nielsen R.: "ニューロコンピューティング", pp. 234-247, トッパン (1992).
- (9) Miller G. F., Todd P. and Hegde S.: "Designing Neural Networks using Genetic Algorithms", Proc. of ICGA '89, pp. 379-384 (1989).
- (10) Kitano H.: "Designing Neural Networks using Genetic Algorithms with Graph Generation System", Complex System, 4, 4 (1990).
- (11) Nagao T., Agui T. and Nagahashi H.: "Structural evolution of neural networks having arbitrary

connections by a genetic method”, IEICE Trans. INF & SYST., **E76-D**, 6, pp. 689-697 (1993).

- (12) Sato Y., Ochiai T. and Hatano S.: “Lookahead Planning Model based on Recurrent Neural Networks and 2-D Genetic Algorithm Operation”, Proc. of ICONIP '94, pp. 798-803 (1994).

(平成6年8月2日受付, 12月7日再受付)



佐藤 裕二

昭56 東大・工・物理工学卒。同年(株)日立製作所入所。可変構造論理LSIの研究・開発, ニューラルネットワークのハードウェアの研究・開発に従事。平4年10月より新情報処理開発機構へ出向。情報処理学会会員。



古谷 立美

昭48 成蹊大学院工学研究科修士課程了。同年電子技術総合研究所入所。平5 新情報処理開発機構へ出向。平6年4月東邦大学教授。人間に近い柔軟な情報処理の実現に興味をもつ。情報処理学会, IEEE 各会員。