

リカレントニューラルネットワークの学習と力学構造

慶應義塾大学 理工学部, Sony CSL

伊藤 真人, 谷 淳

E-mail: masato@mt.cs.keio.ac.jp, tani@cs.sony.co.jp

概要

人間や生物は、その認知の根底においては何らかの非記号的記述の上で、認知あるいは学習を行っていると考えられる。本研究では、その非記号的記述として力学系による記述を考える。力学系としてリカレントニューラルネットワークを用いた学習の実験を行い、そこで見られる力学構造から力学系の記述に基づく学習について考察する。

1 はじめに

人間社会および生態系は、無数の自律的な知的主体(エージェント)が複雑な相互作用を絶えず行っている結果として形成される系である。これらの系は外部との明確な境界を持たず、エネルギーの外部から内部への流入あるいは内部から外部への流出が絶えず起こる開放系である。この無限に広がる開放系に対して、系を構成するエージェントは有限の能力しか持たない。個々のエージェントが系全体の状態を把握することは不可能であり、常に最適な行動を取ることはできない。しかしエージェントはその存在を維持していくために、致命的な過ちを犯すことなく活動を続けている。有限の経験から、エージェントはその内部に何らかの形で系の適切なモデルを構築し、それに基づいて行動を決定しているのである。

計算機上にこのエージェントを創り出そうとするとき、エージェントが行っている認知について考えなければならない。その際には、まずモデルが必要となる。人工知能の分野では、知識を完全に記号で表現し、その記号を操作するという手法で認知のモデルを構築してきた。つまり、エージェントはその内部に置かれた環境のモデルを記号を用いて表現しているのである。しかし人間および生物が認知の根底において記号的記述の用いているとは考えにくい。もちろん、認知過程の上位レベルでは記号的記述に基づくものもあるが、そこでも、その記号自体はどのように表現されているのかといった問題が残されている。

一方、知性はカオスを含む力学系によるものであるという解釈の下に、認知のモデルを力学系の枠組の上に構築しようとする試みがなされている [4, 1, 5]。エージェントが行う認知を力学系の時間発展の上で考えようとしているのである。エージェントは置かれた環境を1つの力学系として捉えて、それに自分が持つ内部力学系を同調させることにより環境のモデルを表現しているのである。

本研究では、力学系としてリカレントニューラルネットワークを取り上げる。そのリカレントニューラルネットワークの学習の実験を通して、力学系による記述における学習について考察する。

2 実験

リカレントニューラルネットワークは、ニューラルネットワークの中でも時系列情報を扱うことのできるモデルである。ネットワークにフィードバック結合されたコンテキストユニットを持つことにより、そこに状態を保持し時系列情報を扱っている。そのネットワークの出力は、現時点での入力だけでなく過去の入力の履歴により決定される。

このリカレントニューラルネットワークの動作は、コンテキストユニットに保持された内部状態を、ユニット間の結合の重みで決定される非線形写像により、更新していく過程である。これは力学系が (1) 式のような差分方程式により時間発展していくことと同様である。その系の性質を示す力学構造は、学習により形成された学習対象のモデルとして考えることができる。

$$x_{t+1} = f(x_t) \quad (1)$$

本研究ではリカレントニューラルネットワークの学習についていくつかの実験を行い、それらを力学系の視点から解析することにより、力学系における学習について考察する。

2.1 有限状態機械の学習

まずはじめにリカレントニューラルネットワークに有限状態機械を学習させる。本研究における学習の実験は、リカレントニューラルネットワークのエージェントが有限状態機械で表現される環境を学習するという想定の下で実験を行う (図1)。エージェントは環境に関する情報をセンサ入力として得て、そのサンプルを基に学習を行う。エージェントはモータコマンドとして行動を選択し、その行動の選択により得られるセンサ入力決定されるものとする。ここでエージェントにとっての学習の目的は、現時点でのセンサ入力においてエージェントの選択した行動に対して環境から得られるであろうセンサ入力を順次予測していくことである。

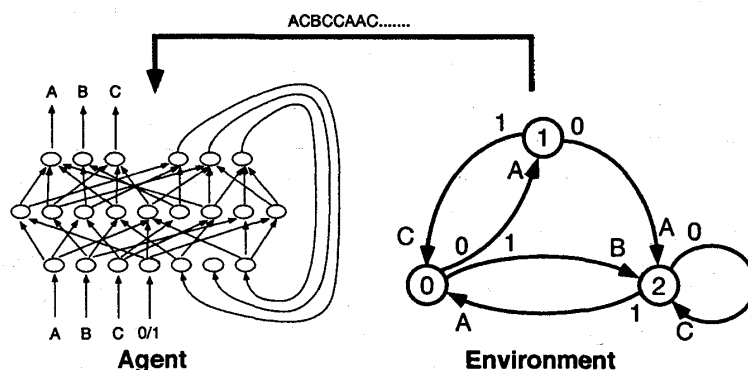


図1: 有限状態機械の学習の実験

リカレントニューラルネットワークの構造は図1に示す通りである。各ユニットの数は、入力ユニット4つ、隠れユニット8つ、出力ユニット3つ、コンテキストユニット3つとした。学習アルゴリズムは Back Propagation Through Time (BPTT) を用いた。

学習対象として、図1に示す有限状態機械を学習させる。この有限状態機械の状態数は3である。エージェントは各状態において、2種類のモータコマンド {0, 1} から行動を選択することができる。ある状態においていずれかの行動を選択したときに、3種類のセンサ入力 {A, B, C} から1つを得て、次の状態へと移動する。こうしてエージェントは環境中を動作する。エージェ

ントが得られる情報は、ある行動を選択したときに得られるセンサ入力だけであり、エージェント自身が現在どの状態にいるかということは知ることはできない。また、センサ入力は状態ごとにユニークなものではないので、現時点のセンサ入力だけから現在の状態を知ることはできない。次に実験の手順を以下に示す。

1. 学習サンプルの収集

エージェントを環境中で動作させ、センサ入力の列を集める。このとき、エージェントはランダムにモータコマンド (行動) を決定する。

2. 環境の学習

モータコマンドの列に対するセンサ入力の列を教師信号として、BPTT を用いて学習を行う。

3. 環境中の動作

エージェントを再び環境の中で動作させる。

学習後に環境中を動作させたときのリカレントニューラルネットワークの相空間を図 2 (a) に示す。コンテキストユニットにランダムな値を与えて、環境中を 10000 ステップ動作させて、そのトラジェクトリをプロットとしたものである。ただし、1000 ステップ以降のトラジェクトリだけをプロットしている。

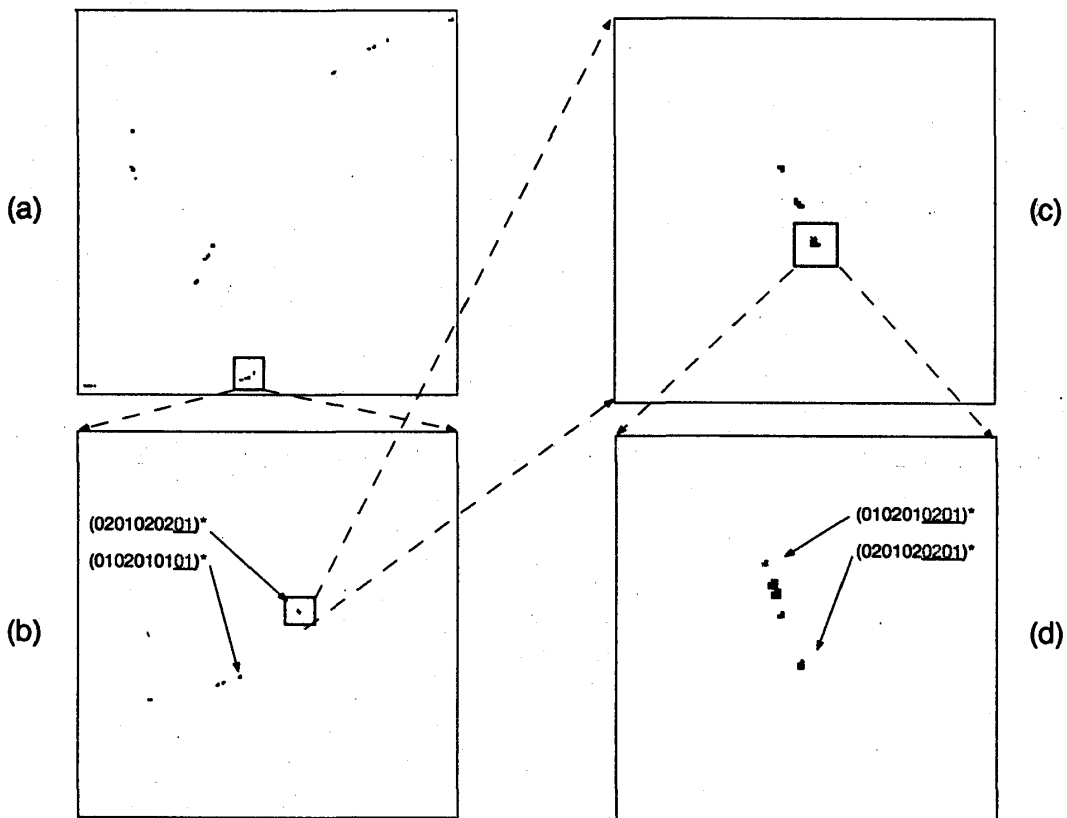


図 2: 有限状態機械学習後の動作

これを見ると、有限状態機械を学習させたときのリカレントニューラルネットワークの相空間のトラジェクトリは、いくつかのセグメントになっていることが分かる。このセグメントはコ

ンテキストユニットの初期状態を変えて動作させても同様な結果となる。したがって、このセグメントはリカレントニューラルネットワークの力学系としてのアトラクタなのである。各セグメントにあるトラジェクトリを、有限状態機械の状態と対応させてみると、1つのセグメント内のトラジェクトリは全て同一状態である。この各セグメントは学習対象の有限状態機械の各状態に対応していると考えることができる。

リカレントニューラルネットワークにリミットサイクルを学習させた場合、その力学構造はリミットサイクルのアトラクタになる。リミットサイクルの各状態に対応するものが点で表現されている。このセグメントととの違いが何から生じているのかを考えてみると、リミットサイクルにおいてエージェントが現在の状態を知るためには、周期分の履歴が必要である。一方、有限状態機械は文法的な束縛はあるものの特定の周期が存在しないので、無限周期分の履歴を保持しなければならない。

そこで、有限状態機械の学習により形成された力学構造についてさらに解析を行った。(b)、(c)、(d)は、それぞれ(a)、(b)、(c)の四角で囲まれた部分を拡大したものである。

各図の1つのセグメントを小さなセグメントの集まりと見ると、どのセグメントも似た形状をしている。相空間上の全てのセグメントは自己相似構造を持つフラクタル構造を形成しているのである。有限状態機械を学習させた場合にその力学構造がフラクタル構造になることを、KolenはリカレントニューラルネットワークをIterated Function Systems[2]としてみることにより説明している[3]。そこで、リカレントニューラルネットワークが、このフラクタル構造に有限状態機械の学習に必要な履歴をどのように埋め込んでいるかを解析するために、この記号力学系を解析した。

トラジェクトリの中で、有限状態機械における状態の履歴が(0201020201)*と(0102010101)*と点を探すと、どちらも(b)に存在している。この2つの点は過去1状態の履歴が同じで、それ以前の履歴が異なる。次に状態の履歴が(0201020201)*と(0102010201)*となる点を探すと、どちらも(d)に存在している。この2つの点は過去3状態の履歴が同じで、それ以前の履歴が異なる。解析の結果をまとめると、過去1状態の履歴が同じ点が一番大きなセグメント(a)で分けると同じセグメントにあるが、そのセグメントの中(b)では別のセグメントにある。次に過去1状態の履歴が同じ点が一番大きなセグメントから2回拡大したセグメント(c)までは同じセグメントにあるが、そのセグメントの中(d)では別のセグメントにある。つまり、フラクタル構造における階層と履歴が一致している。フラクタル構造は部分に全体の縮図を持つので、その縮図の中に前状態が埋め込まれている。フラクタル構造全体で考えると全履歴が埋め込まれていることになる。

2.2 未学習の環境

次に、未学習の環境に対するリカレントニューラルネットワークの力学構造について解析する(図3)。

学習対象として、図3に示す有限状態機械を学習させる。ここでは、実線で示されているパスのみから成る環境を学習させる。学習後、エージェントを環境中で動作させるときには、状態2に点線で示されるパスを新たに付け加えた環境で動作させる。エージェントにとってはこのパスが未学習の環境となる。

実験の手順を以下に示す。

1. 学習サンプルの収集

エージェントを環境(点線のパスなし)中で動作させ、センサ入力 of 列を集める。このとき、エージェントはランダムにモータコマンド(行動)を決定する。

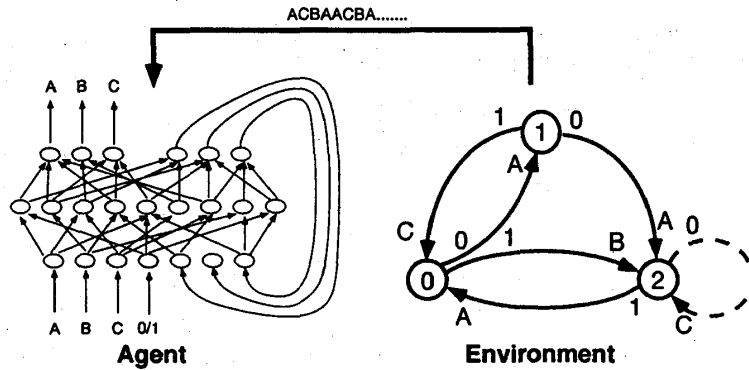


図 3: 未学習の環境の実験

2. 環境の学習

モータコマンドの列に対するセンサ入力の列を教師信号として、BPTTを用いて学習を行う。

3. 環境中の動作

エージェントを未学習環境 (点線のパスあり) 中で動作させる。

学習後のリカレントニューラルネットワークの相空間を図4に示す。(a)はリカレントニューラルネットワークを学習させた環境中で、(b)は未学習の環境中で動作させたものである。(a)と(b)を比較してみると、全体的に大きな構造上の違いはない。ただし(b)の方がセグメントの数が多く、個々のセグメントの大きさも大きい。新たなセグメントは、エージェントが未学習の環境 (点線のパス) を通った直後の状態に対応している。この状態において、エージェントは未学習の環境に対する適切な出力を出せず、エラーを生じる。しかし、エージェントは力学系の時間発展にあたる動作を繰り返すうちに再び元のセグメントに戻る。ここでは、エージェントが学習した環境なので適切な出力を出すことができる。

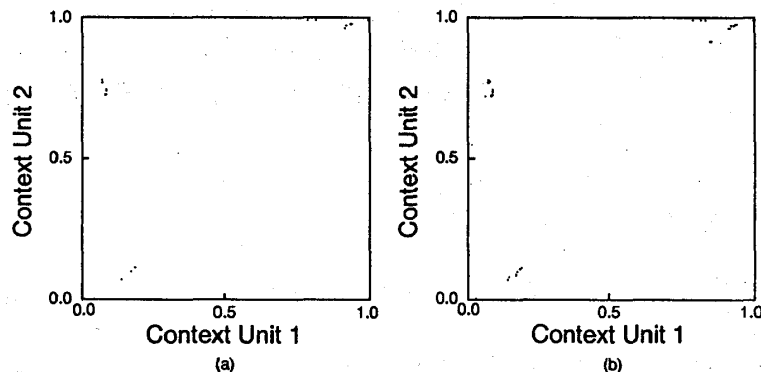


図 4: 学習後の動作の比較

未学習の環境に対する力学構造についてさらに解析を行った。図5(a)は、リカレントニューラルネットワークを未学習の環境中を動作させたときのトラジェクトリをプロットした後、学習した環境中を動作させたときのトラジェクトリをプロットとしてものである。未学習の環境中に学習した環境が含まれているので、学習したトラジェクトリの下には必ず未学習のトラジェクトリがある。これを見ると、未学習の環境中を動作させたときのトラジェクトリと学習した環境中だ

けを動作させたときのトラジェクトリが相空間上に混在している。(b)は(a)の中の四角で囲まれた部分を拡大したものである。これを見ると、2つのトラジェクトリが混在している様子をはっきり見ることができる。

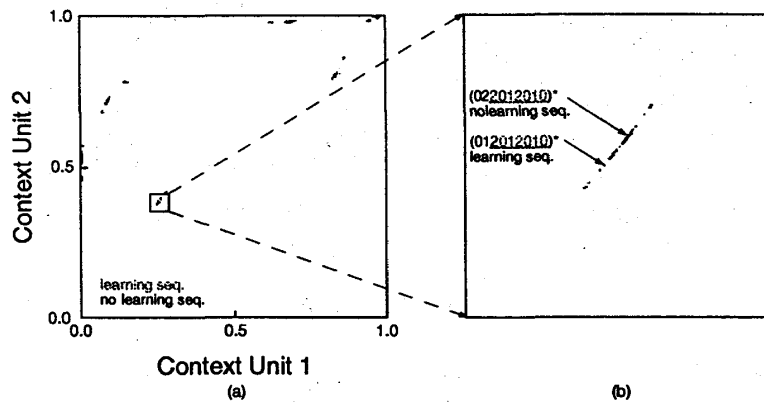


図 5: 学習後の動作

ここでも、有限状態機械の学習における力学構造の解析と同様、この記号力学系を解析した。トラジェクトリの中で、状態の履歴が $(012012010)^*$ と $(022012010)^*$ の点を探す。この2つの点の履歴を比較すると、現在の状態(0)から過去6状態の履歴が同じである。ただし、 $(022012010)^*$ はその前状態が2であるので、2の状態において未学習の環境を通過している。しかし、そのどちらの点も(b)の中に存在している。未学習の環境を通過したトラジェクトリは良く似た履歴を持つ学習した環境だけを通ったトラジェクトリのすぐ近くに存在している。学習した環境だけを通ったどの履歴についても、その1つ前に未学習の環境を通過した履歴を考えることができる。したがって、学習した環境だけを通った履歴を持つどの点の近傍にも、必ず未学習の環境を通過した点が存在する。

リカレントニューラルネットワークは、学習したものと未学習のものを明確な線を引いて区別しているわけでない。一見混在しているように見えるが、それらをフラクタル構造を持つ力学構造の中で区別していると考えられる。

2.3 追加学習

最後に、ある環境を学習した後に環境を追加して再び学習させた場合にリカレントニューラルネットワークの力学構造がどう変化するかを解析する(図6)。

学習対象として、図6に示す有限状態機械を学習させる。まずエージェントは実線で示されるパスのみの環境を学習し、学習エラーが十分に減少したところで、状態2に点線で示されるパスを付け加えた環境を学習させる。

実験の手順を以下に示す。

1. 学習サンプルの収集(1)

エージェントを環境(点線のパスなし)中で動作させ、センサ入力の列を集める。このとき、エージェントはランダムにモータコマンド(行動)を決定する。

2. 環境の学習(1)

モータコマンドの列に対するセンサ入力の列を教師信号として、BPTTを用いて学習を行う。

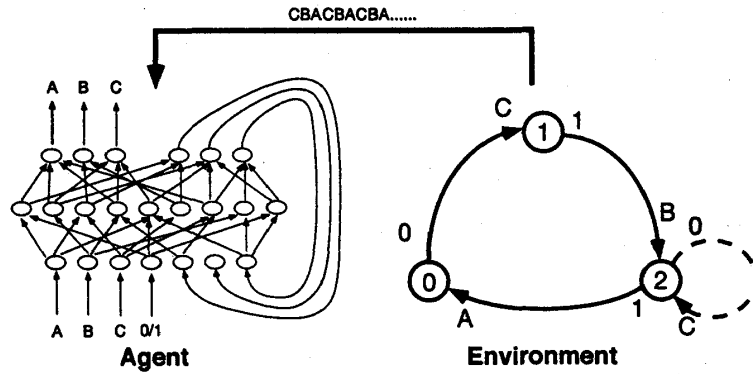


図 6: 追加学習

3. 学習サンプルの収集 (2)

学習エラーが十分に減少したところで、エージェントを変化した環境(点線のパスあり)に入れて動作させ、センサ入力の列を集める。このとき、エージェントはランダムにモータコマンド(行動)を決定する。

4. 環境の学習 (2)

モータコマンドの列に対するセンサ入力の列を教師信号として、BPTTを用いて学習を行う。

学習中のリカレントニューラルネットワークの相空間を図7に示す。(a), (b), (c)はそれぞれ学習環境追加後、学習ステップが0, 1000, 6000ステップのリカレントニューラルネットワークを、追加した環境中で動作させたものである。(a)では追加した環境を学習していないので、状態2にある点線で示されるパスを通る場合、リカレントニューラルネットワークはうまく対応することができない。状態2であるトラジェクトリが相空間の広い範囲に散在している。(b), (c)と学習が進むにしたがって広い範囲に存在していたトラジェクトリは、徐々に1箇所に集まり最終的に有限状態機械の学習で見られたセグメントになっている。

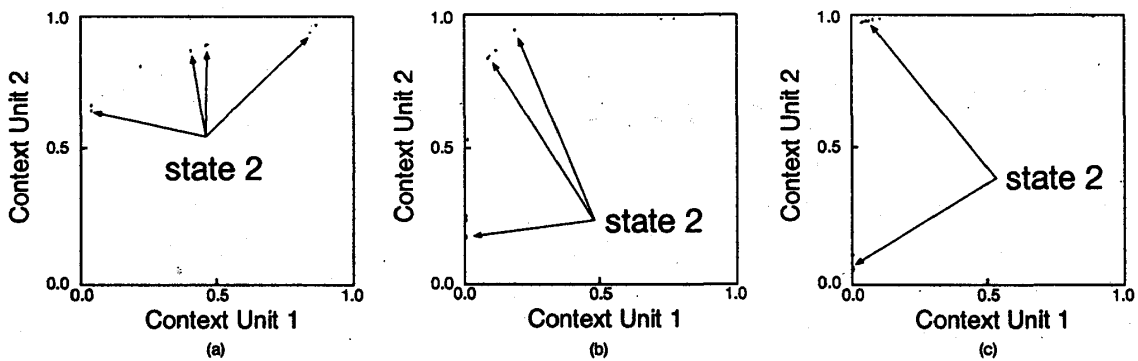


図 7: 学集中の力学構造の変化

3 まとめ

リカレントニューラルネットワークにおける学習は、非線形力学系において系のパラメータを変化させることと同じである。非線形力学系においては、系のパラメータの変化がその系の力

学構造の変化を生む。リカレントニューラルネットワークは学習対象に応じて適切に力学構造を変化させて、学習対象のモデルをその内部力学系のなかに表現することにより学習を行っているのである。有限状態機械の学習においては、フラクタル構造のアトラクタを形成することにより学習することを示した。そのフラクタルの階層に過去の履歴を埋め込んでいることが分かった。このフラクタル構造のアトラクタは、未学習の環境においても、大きくその構造を崩すことはなかった。追加学習の実験で見られたように、リミットサイクルのアトラクタにはその周期分以上の情報を保持する構造はなく、未学習の環境に対応するトラジェクトリは相空間の広い範囲に散在していた。一方、フラクタル構造は有限状態機械における履歴情報を保持するために、多量の情報をしまい込むことのできる構造になっている。そこには、未学習の環境を区別するための情報も埋め込まれていると考えられる。

本研究で行った有限状態機械の学習は帰納学習である。これを記号的記述による学習の場合、学習結果が学習対象と全く等価なものとなる。一方、本研究で用いた力学系による記述における学習の場合、有限状態機械を全履歴を保持することにより近似的に学習している。ただし、これをセグメントの単位で見れば学習対象と等価であると考えることができる。

参考文献

- [1] 谷 淳. ロボットにおける認知と自律性の構造：力学系の見地から. 日本ロボット学会誌 *Vol.14 No.4*, 1996.
- [2] Michael Barnsley. *FRACTALS EVERYWHERE*. ACADEMIC PRESS, 1988.
- [3] John F. Kolen. *Exploring the Computational Capabilities of Recurrent Neural Networks*. PhD thesis, The Ohio State University, 1994.
- [4] Jordan B. Pollack. The induction of dynamical recognizers. *Machine Learning*, 1991.
- [5] Jun Tani and Naohiro Fukumura. Embedding a grammatical description in deterministic chaos: an experiment in recurrent neural learning. *Biological Cybernetics*, 1995.