2008年2月

自律型ロボットにおける制御回路構造の学習手法
～オンライン・リアルタイムな
ネットワーク構造の強化学習～

Learning Method for Control Circuit in Autonomous Robot
～Online Real-Time Structural Reinforcement Learning～

早稲田大学 理工学研究科

金　天海

Abstract

In recent years, practical use of autonomous robot is expected in several fields such as communication, space search, and replacement of sevier human works. On the other hand, robot engneer have some difficulty or inability to design effective control rule for such autonomous robots, because of the unpredictability of environments.

So, in this paper, we investigated a learning system based on such robot's requirements, emergence of effective behavior, adaptation to various emvironments, and immediacy of learning. In conventional reinforcement learning system, it is difficult to fullfill these three requirements, because they require division of state space, translation of input signal, or determination of network topology by every environment or every task. On the other hand, for conventional systems based on genetic algorithms, online realtime calculation is difficult. Therefore, in this paper, we proposed a novel framework Self-Organizing Network Elements (SONE) as a solution of these problems. This SONE is a online realtime network structural learning system, which is based on reinforcement signal propagation method.

We introduced tracking problem and two-spiral problem so as to examine the ability of generalization, incremental learning, temporal sequence learning, and etc. In these examinations, we could confirmed all of these ablities. From these experiments and collision avoidance experiment with a mobile robot, effectiveness of SONE was confirmed against initial three robot's requirements. Also, more effective composition of SONE was revailed by the enhancement of elemnt's noise resistance.

This SONE realizes general purose learning system by its composition. So, the possibility of it's application is open and not limited to the field of robotics. Also, SONE is related to many other fields (i.e. reinforcement learning, genetic algorithms, neural networks, boosting, multiagent).

摘　要

現在，コミュニケーション，宇宙探索，過酷環境下での労働の代替などの多くの技術分野において自律型ロボットの応用への期待が高まっている．一方で，そのような環境は予測不可能な側面を持っており，ロボット開発者がロボットに効果的な制御則を実装することが困難な場合が多い．

そこで本論文では，自律型ロボットに必要となる，効果的な振る舞いの創発（創発性），多様な環境への適応（適応性），即時的な対応（即時性）に着目して学習制御器の開発を行った．従来提案されている強化学習システムでは，環境・タスク毎の状態空間分割や，入力信号の変換，またはネットワークトポロジーの決定が必要となり，多様な環境への適応を行うことが困難である．一方で，遺伝的アルゴリズムに基礎をおいたシステムでは，即時的に学習を行うことが困難である．そこで本論文では新しい学習システムの枠組みとして，自己組織化回路素子 Self-Organizing Network Elements (SONE) を提案する．この SONE は，オンライン・リアルタイムなネットワーク構造の強化学習法である．

SONE に対し，軌道学習や二重螺旋問題による基本特性試験を行うことで，汎化能力，ノイズ耐性，追加学習能力，時系列学習能力などについての検証と他の学習制御器との比較を行った．それと併せて，移動ロボットにおける衝突回避実験を行うことで，先に示したロボットの要求機能に対する SONE の有効性が確認できた．さらに，SONE を構成する各素子に対しノイズ耐性を持たせることで，より効果的な強化学習が実現できることも明らかとなった．

SONE は環境やタスクへの依存性を低減した構成をとることによって，汎用的な学習システムを実現しており，将来的には広い分野での活用が期待できる．また学術的には，強化学習や遺伝的アルゴリズムのみならず，ニューラルネットワーク，ブースティング，マルチエージェント等の多くの分野と関連があると考えられる．

目 次

用語説明

ここでは用語の解釈を統一するため，本論文における解釈について説明を行う．

創発	明示的には観測・予期できなかった機能や振る舞いが，系に関わる要素間の相互作用の結果から立ち現れること．
汎化	サンプルデータを，ドメインが持つ特徴の推定によって効率的に近似すること．
抽象化	データを圧縮性の高いシンボルとして表現すること．
リアルタイム性	制御対象の持つサイクルと同期できる時間間隔によって演算結果が得られること．
漸次性	追加学習をした際の既学習データ損失が少ない状態を漸次性が高い状態とする．
学習器・学習制御器	本論文では，数値計算を用いて学習システムを構成する手法のうち，機械制御を行う目的で使用可能な手法を学習制御器，それ以外を学習器として区別して表記する．
教師あり学習	学習制御器が学習するべき入出力写像のサンプルを，学習制御器の外部から学習信号として受け取り，受け取った学習信号を表現するための関数を獲得する手法の総称．この学習法に基いた学習制御器は，圧縮された情報表現や汎化の点で有効な写像の獲得を目的として用いられる場合が多い．誤差逆伝播法 Error back propagation を用いたニューラルネットワークの大部分はこの方式をとっている [1–9].
教師なし学習	入力，出力という二種類の信号をセットとして学習信号とする教師あり学習とは対称的に，教師なし学習におけるサンプルは入力のみで構成する．この学習法に基いた学習器は，学習信号として与えられたデータの想起やクラスター化を目的として用いられる場合が多い．自己組織化マップ Self-Organizing Map (SOM) などの学習器はこの方式をとっている [10, 11].

強化学習	強化学習は「入力に対して出力を返す学習システムを考えた場合，強化信号（スカラーの価値信号）を伴う入力を数多く得るような行動出力を獲得するための入出力の写像を学習するメカニズムを指す」と定義されており [12]，広義には遺伝的アルゴリズムの多くもこの枠組みに含まれる．この学習法は機械学習における行動創発を実現するために広く用いられている．
遺伝的アルゴリズム	進化論に従った方法で，仮想的な遺伝子を交叉・突然変異させることによって，効果的なルールを探索・獲得していくアルゴリズムの総称．
状態空間	状態空間は系の取り得る状態に対応する空間であり，一般には状態を記述するために必要となるだけの次元を持つ．機械制御では，センサ入力数の次元を持った空間として用いられることが多い．
行動空間	行動空間は系の取り得る行動に対応する空間であり，一般には行動を記述するために必要となるだけの次元を持つ．機械制御では，モータ出力数の次元を持った空間として用いられることが多い．
ϵ-Greedy 法	探索を伴った学習制御器において行動選択を行うための基本的な手法．学習制御器が獲得した行動選択の中で最も効果的な方法以外の方法を，一定確率 ϵ でランダムに選択する [13]．
次元の呪い	状態空間を行列等で分割する場合，状態空間の次元が広がるに従って分割数が指数関数的に増大し，様々な弊害を生み出す．この，次元にまつわる不具合を次元の呪い Curse of Dimension と言う [13]．

第1章　序論

1.1　自律型ロボットの制御則

現在，ロボットは，人間にとって身近な存在として社会に浸透しつつある．ホンダの2足ロボット [14]，ソニーのペットロボット [15] や，早稲田大学ヒューマノイドプロジェクト [16] の成果はマスメディアを通じて紹介され注目を集めており，今後の産業発展に大きな影響を与えると考えられる．そして近年では，宇宙・海底の未知環境を自律的に探査するロボットや，対人コミュニケーションロボット等の自律型ロボットの応用に対する期待も高まっている．

一方で，このような自律型ロボットにおける効果的な制御系の構築法は未だ完成されたとは言い難い．そして，自律型ロボットを社会において広く効果的に活用できる枠組みを作るには，その制御系の構築技術を高める必要がある．特に自律型ロボットでは，設計段階で効果的な制御則を実装することが原理的に難しい場合があり，応用を目指したロボット開発の障害となっている．そこで，本論文ではこの問題に着目した制御則の構築を考える．

以下では，未知環境探査やコミュニケーションを行うロボットを例として，設計段階において効果的な制御則を実装することの難しさと学習制御の必要性を述べる．

1.1.1　未知環境探査ロボット

まずは，Mars Exploration Rover [17] に代表されるような，宇宙・海底の未知環境を探査するロボットを例として説明を行う．現在の探査用ロボットの多くは遠隔操縦によって操作されることが多いが，遠隔操縦による操作には時間遅れの問題や通信回線確保に関する問題があるため，人による操作量を軽減できる自律型ロボットの実用化

への期待は高い.

事実，Mars Exploration Rover Project においても回線が途切れるトラブルがあり，その解決は今後の重要課題だといえる [17].

また，遠隔操縦を行う場合には通信距離に伴った時間遅れの問題がある．火星はロボットの操作が可能な圏内であるが，将来さらに広範囲の宇宙を探査する場合には距離的な限界がある．例えば，双腕宇宙ロボットを用いた遠隔操作の研究がある [18]．この研究は，宇宙遠隔操作実験システム（ARS/A）を用いて，地上から軌道上の宇宙ロボットを遠隔操作する場合に近い環境を再現可能としている．そのうえで，双腕宇宙ロボットの遠隔操作システム（DARTS）が構築されており，単腕のスレーブアームと比べ複雑な船外活動を地上から実現することが期待される．しかしながら，このような研究の多くは宇宙の中でも比較的地球に近い領域においての使用を目標としており，さらに遠くの領域での使用は困難である.

このような背景から宇宙・海底の未知環境を探査するために自律型ロボットを用いることが期待されているが，未知環境の探査に対して効果的な自律型ロボットの開発は原理的に難しい現状がある．通常ロボット開発者は，ロボットが使用されるタスク・環境を織り込んでロボット制御則を設計する．しかしながら未知環境探査に使用するロボットでは，ロボットが使用されるタスク・環境を，ロボット開発時に正確に限定することが不可能である．よって，開発者が自律型ロボットの制御則を設計することが困難であるため，現在用いられているロボットの多くが遠隔操縦によるものとなっている.

1.1.2 コミュニケーションロボット

次に，ソニーのペットロボットや WAMOEBA [19] 等のコミュニケーションを行うロボットを例として説明を行う．現在，コミュニケーションロボットには遠隔操作型や自律型のロボット等の多くの種類がある．中でも自律型ロボットによるコミュニケーションシステムは，今後ロボットが社会において人と活動する際に不可欠なシステムであり，発展が期待されている．このような自律型ロボットが行うコミュニケーション

は，エンターテイメントや情緒交流などが考えられるが，いずれにおいてもユーザの嗜好に合わせた制御則や，ユーザの飽きを回避するための状況に応じた制御則が必要となる．

現在のコミュニケーションロボットの多くは，ロボット開発者がコミュニケーションの場面を限定し，その場面において適切な制御則をロボットに組み込むことで作成されている．しかしながら，このような「作り込み」による制御では，ユーザの嗜好を考慮したコミュニケーションや，状況に応じた飽きの回避を実現することは難しい．この場合，各ユーザが求めるコミュニケーションの様相を充分に想定することができず，ロボットの設計段階において効果的な制御則の設計ができないという問題がある．このようなコミュニケーションにおいて，ロボットの動作を作り込むためには，ユーザの嗜好やコミュニケーションの現場で起こり得る状況の変化を，予め考慮に入れて設計する必要がある．しかしながら，ユーザの嗜好はユーザが未知であるという制約上設計者が知り得ない情報であるし，コミュニケーションの現場で起こり得る状況の変化も無数に存在するため，そのそれぞれの状況に対してコミュニケーションのあり方を逐一設計することは難しい．

一方で，遠隔操作型のコミュニケーションロボットではこのような問題をある程度解決できる．例えば，ぬいぐるみロボット Keepon [20] は，乳児期から幼児期の子どもたちと，安全なインタラクションができるようにデザインされており，高さ 120mm・直径 80mm のシリコンゴムでできたダンゴ型の身体を備えている．この身体を使って，

1. 注意の表出：顔（つまり視線）を人物や対象物に向けること
2. 情動の表出：身体を左右あるいは上下に揺すり，楽しさや興奮といった心の状態を表現する

という二つの動作を遠隔操作によって行うことで，乳幼児とのコミュニケーションを行っている．このシステムでは，ロボットと乳幼児の間で非常に長期間のコミュニケーションが実現できており，乳幼児の嗜好，状況の変化に対してロボット操作者が適切にロボットを制御することで対応が可能であった例であるといえる．

また，剣道ロボットシステムを用いたエントレイメントに関する研究では [21]，二人の人間がリズムコントローラを介して模式的な剣道対戦を行うための剣道システムが用いられており，ロボットはアバターとして遠隔操作によって操作できる．そして，このようなコミュニケーションシステムを通じた間合い形成時にコントローラ操作リズムにコヒーレンスの生成，崩壊が繰り返し起きることが発見されており，タイミングや間合い，リズムの形成に関する新しい知見が得られている．そして，コヒーレントな状態（ロボットユーザ間でのある種の情報共有が認められる状態）とインコヒーレントな状態（ロボットユーザ間でのある種の情報共有が認められない状態）とが繰り返されることより，ユーザ間で常に新しいルールを生成しながらコミュニケーションが進行していることがわかる．ここでも，状況の変化に応じて作り出されるコミュニケーションに対する適切なロボット制御則を随時操作者が提供しているといえる．

このように，遠隔操作型のコミュニケーションロボットでは，ユーザの嗜好や飽きをある程度回避できると考えられる．一方で，自律的に活動するロボットがコミュニケーションを創出するような場合 [19] には，多彩なコミュニケーションの様相に対して，ロボット自身の判断での対応が必要となるため，先に示した通り，これらの対応の仕方を設計段階で作成することは非常に困難である．

1.1.3 学習制御

このように，ロボットが設計者の想定し得ない状況へ対処しなければならない場合においては，設計段階における効果的な制御則の実装が原理的に不可能か，またはできたとしても非常に困難であることが多い．そして，効果的な制御則の実装に関する問題は自律型ロボットの設計において特に顕著に現れる．一般に，このような場面において使用する自律型ロボットの効果的な制御則を実現するには，学習制御が有効であると考えられている．

1.2 学習制御による制御則の獲得

前節で述べた，自律型ロボットの応用に対する期待と，自律型ロボットの制御則構築に関する問題に対し，本節では学習制御を用いることの有効性を述べる．前節で述べたように，自律型ロボットにおける効果的な制御則をロボット製作段階で作成することが，原理的に不可能な場合，または困難である場合が存在する．学習制御によるアプローチでは，ロボット製作段階において設計者が製作困難となる制御則を，ロボットが活動する環境の中で得られたデータを基に構築することが期待できる．この場合には制御則の構築は，設計段階で得られないデータが得られた段階で行われるため，環境やタスクに対してより効果的な規則を構築することが期待できる．

例えば，強化学習を用いて制御則を作成する場合には，設計者は環境やタスクに対する詳細な知識を持っている必要は無い [13]．強化学習の枠組みでは，ロボットは報酬を与えられる目標状態へ到達するための制御則を環境，タスクの中で学習することができる．よって，設計者は制御則を学習する学習制御器のパラメータ設計と，ロボットに対する報酬値の与え方の設計をするだけでよい．

ただし，強化学習による学習制御器を使うアプローチは，設計者が全ての制御則を作り込むというアプローチに較べれば効果的であるが，自律型ロボットに即した効果的な学習制御器の設計法としては完成されたとは言いがたい．よって，本論文では自律型ロボットに求められる学習制御器について考察し，その効果的な設計法を導出する．

1.3 自律型ロボットに求められる学習制御器

1.3.1 自律型ロボットの要求機能

設計者にとって想定が困難な状況で活動する自律型ロボットに求められる機能として以下の機能がある．

1. 効果的な振る舞いの創発（創発性）

2. 多様な環境への適応（適応性）

3. 即時的な対応（即時性）

効果的な振る舞いの創発（創発性）は，自律型ロボットが設計者にとって想定困難な状況・環境に対応するために必要となる機能であり，ロボットが新しい状況・環境へ遭遇した際に自身の制御即を新たに獲得するために必要となる．

また，そのような環境下でロボットが自律的であるためには，多様な環境への適応（適応性）が必要である．つまり，人によるロボットの調整を極力省くことができるシステム，単純な外部パラメータによって調整可能（または，調整を必要とせず，外部パラメータを全く持たないシステム）であることが要求される．

さらに，ロボットの活動する環境は動的に変化するため，その動的な変化に追随して学習を行うために，ロボットの学習システムには即時的な対応（即時性）が求められる．

1.3.2 学習制御器の要求機能

以上に示したロボット側の要求機能を学習制御器側の仕様へ置き直した場合，次の五要素との対応が議論される必要がある．

1. 行動創発
2. 汎化・抽象化
3. 柔軟性
4. オンライン性
5. 漸次性

1.3.2.1 行動創発

行動創発は創発性を支えるための要素であり，新たなロボットの行動（出力）を学習制御器が創発的に生成するために必要となる．

1.3.2.2 汎化・抽象化

汎化・抽象化は適応性・即時性を支えるための要素であり，ロボットの獲得した知識を効率良く保持するために必要となる．

1.3.2.3 柔軟性

柔軟性は創発性・適応性を支えるための要素であり，ロボットが置かれた環境・タスクによらず使用可能な学習制御器を提供するために必要となる．柔軟性を確保するためには，学習制御器のパラメータを極力簡素化することで，タスク・環境毎の調整を極力避けることが望ましい．

1.3.2.4 オンライン性

オンライン性は即時性を支える要素であり，動的に変化する環境に対し随時学習によって対応するために必要である．

1.3.2.5 漸次性

漸次性は即時性を支える要素であり，既学習のデータを積み上げ，忘却を抑制することで，既学習の対象に対して即時的に対応するために必要となる．

1.3.3 従来手法

以上の要求機能を両立させるという観点から従来研究について考察する．

1.3.3.1 行動創発

一般に学習法は教師あり学習，教師なし学習，強化学習に大別される（遺伝的アルゴリズムは広義の意味での強化学習に区別することができる）．そして，自律型ロボットにおける学習制御器において目的に即した行動創発を実現するためには，これらの学習法のうち強化学習が有効である．教師あり学習や教師なし学習では，いずれにお

いても入出力データを学習制御器自らが探索することが無いため，それ自体では目的に即した行動創発に用いることはできない．

従来の手法では，強化学習 Reinforcement Learning (RL) に基づいた手法 [13] や，遺伝的アルゴリズム Genetic Algorithms (GA) に基づいた手法 [22] 等によって行動創発を達成することができる．

例えば強化学習では，ϵ-Greedy 法等の探索手法によってロボット自身が効果的な出力を試し，獲得することができる．ϵ-Greedy 法は，ロボットの獲得した行動選択の中で最も効果的な方法に対し，一定確率 ϵ でランダムな行動選択を行うルールを付加することで，新しい行動の探索を試みる手法である．この方法で探索した新しい行動が効果的であった場合には，新たな効果的な行動選択としてロボットの行動選択ルールへと反映する．これによってロボットは，順次自らの行動選択を改善していくことができる．

また，遺伝的アルゴリズムでは，遺伝子の交叉や突然変異によって効果的な出力ルールを探索・獲得していくことができる．遺伝的アルゴリズムでは，進化論に従った方法で個体（ロボットの制御ルール）を選択・淘汰することで，ロボットの行動を改善することができる．この手法では，遺伝子の交叉や突然変異によって生じる新しい個体（新しい制御ルール）が効果的な出力の自律的探索手法となっており，新しくできあがった効果的な個体は，進化のプロセスの中で，必要に応じて遺伝する（保持される）．

このように従来，強化学習や遺伝的アルゴリズムの分野では，ロボットにおける行動創発に関する手法が広く議論されている．

1.3.3.2　汎化・抽象化

強化学習の分野等では，データベース的な手法にまつわる次元の呪い Curse of Dimension や，それに伴った状態空間分割に関する問題を回避するためにネットワーク型の学習制御器を用いて汎化・抽象化を行うことで，高次元データへ対応するための技術がある．以下ではこれらの問題の概要を説明する．

Q-learning [23, 24] 等の一般的な強化学習では，ロボットの状態遷移をマルコフ決定過程 Markov Decision Process (MDP) [25] として記述しており，本来は連続かつ無限

に存在するロボットの状態を有限個の状態に区切らなければならない．この状態空間の分割数は状態の次元が増加する毎に指数関数的に増大するため，高次元での計算が困難となる（次元の呪い）．よって設計者は，状態空間の分割を問題・環境毎に適宜調整しなければならず，その分割法が問題となる（状態空間分割に関する問題）．

この状態空間を自動的に分割する手法として，矩形基底による自律分散型関数近似 [26] や階層型強化学習 Multi-Layered Reinforcement Learning (MLRL) [27, 28] 等が提案されている．これらの手法では状態空間の自動的な分割を実現しており，特に後者ではネットワーク状に配置されたレイヤーに Q-learning モジュール群を自動生成する手法によってこの解決を試みている．

1.3.3.3 柔軟性

先に示した自律分散型関数近似や MLRL 等のシステムでは各タスク・環境毎にセンサ入力を適切に変換して学習器へ送信する必要があるため，この柔軟性を確保することが難しい．

具体的には，浅田らはサッカーロボットにおける学習システムを構築しているが，MLRL へ送信する信号は，ロボットに備えられたカメラや，サッカーフィールド上のカメラの画像処理を行うことで取得した，ゴールの位置座標，ボールの位置座標などであった．そして，これら位置座標情報の選定には，サッカーというタスクへ特化した対象（ゴールの位置，ボールの位置）等に関する設計者の知識を必要としている．

しかしながら，例えば宇宙探査ロボットや対人コミュニケーションロボット等において，行うタスクが予め明確に規定できない場合においては，MLRL に伝達すべき入力に何を設定するべきかは明らかではないという問題がある．よって，このような場合には MLRL を用いることが難しい．

そこで，この手法をさらに発展させた学習法として，Direct-Vision-Based Reinforcement Learning (DVB-RL) [29] が提案されている．この手法では，ロボットの入出力をニューラルネットワークへ直結し，入出力の信号変換を行わずに強化学習を実現することで，各タスク・各環境毎の入力信号の変換を省略することができる．

しかし依然として，階層型強化学習やDVB-RLに用いられるネットワーク構造の決定は，設計者によってタスク・環境毎に行われる必要があるという問題が残っている．

一方で遺伝的アルゴリズムによるアプローチとして，NeuroEvolution of Augmenting Topologies (NEAT) [30] やSelf-Designing Neural Network (SDNN) [31] 等が提案されている．これらの手法ではニューラルネットワークの構造とそれぞれのリンクの重みを遺伝的アルゴリズムによって決定することで，設計者による状態空間の分割，入力信号の変換，さらにはネットワーク構造の決定を必要とせず，タスク・環境への依存の少ない学習制御器が構成できる．

これらの方法をロボットに応用する際の具体的なプロセスは以下のようになる．ロボットを制御するためのニューラルネットワークを多数用意し，各ニューラルネットワーク（表現型）に仮想的な遺伝子（遺伝子型）を対応させる．各表現型をロボットに組み込み，テストを行うことで，各表現型の評価値を算出する．得られた評価値を遺伝子型に適用し，その評価値に応じて遺伝子の交叉，突然変異，淘汰などの進化プロセスを実行する．これらのプロセスで得られた遺伝子から，次の世代の表現型を作成するというものである．

この方法では，制御器としてニューラルネットワークを用いているため，DVB-RLと同様に，状態空間の分割や入力信号の変換を必要としない学習が実現できている．また，ネットワークトポロジーに関しては，遺伝子の進化プロセスを通じて獲得することができるため，タスク・環境毎に設計者が決めなければならないパラメータを大きく削減しており，高い柔軟性を備えていると考えられる．

1.3.3.4 オンライン性

前節で，行動創発，汎化・抽象化，柔軟性を兼ね備えたNEATやSDNNを紹介したが，これらの学習制御手法では各遺伝子型に対応する表現型をテスト・評価するための評価時間の設定を必要とするという問題があり，オンライン性の確保は難しい．

この評価時間は十分に長く設定しておけば，タスク・環境毎の再設定を回避できるため，単純な外部変数での実装と両立できる．しかしながら，その分学習時間は長くなり，オンライン・リアルタイムな学習が困難となる．一方で，評価時間を短く設定

すると学習時間は短縮できるが，どこまで短く設定できるかはタスク・環境により異なるため，やはりタスク・環境に応じた再設定を必要としてしまう．そして結果的に，自律型ロボットの場合，タスク・環境毎の再設定を回避して単純な外部変数の仕様を満たすという観点から，評価時間を長く取る必要が生じてしまう．

よって遺伝的アルゴリズムによるアプローチでは，自律型ロボットにおけるオンラインかつリアルタイムな学習を実現し，動的に移り変わる環境に対しての素早い学習を行うことは難しいといえる．

NEAT を用いてオンライン, リアルタイムに学習を行う手法として，real-time NEAT (rtNEAT) [32] や NEAT+Q [33] が提案されているが，やはり，これらの手法においても自律型ロボットへの応用は難しい．rtNEAT はビデオゲームにおけるマルチエージェントの進化において，エージェント郡全体のオンラインかつリアルタイムな学習を実現している．しかしながら，個々のエージェントの学習は NEAT により行われるため，単体のロボットにおけるオンラインかつリアルタイムな学習は実現できていない．また，NEAT ＋ Q では NEAT と Q-learning を併用することで，NEAT のリアルタイム性・オンライン性を Q-learning により補っている．しかしながら，この手法では Q-learning を用いたために，状態空間の分割問題が再浮上している．

一方で，先に示した強化学習の手法である DVB-RL はオンライン学習を実現しており，柔軟性，オンライン性に関して NEAT と DVB-RL は一長一短の関係にあるといえる．

1.3.3.5 漸次性

漸次性を確保する最も容易な手段としては，学習データをデータベースに保存して参照するという方法がある．しかしながら，他の項目と同時に漸次性を考える場合には多くの技術課題が存在する．

例えば先に挙げた DVB-RL の例ではロボットの学習にニューラルネットワークを用いている．しかし，一般にニューラルネットワークは漸次的学習には不向きであり，新たな学習データを追加学習した再には致命的な忘却 Catastrophic Forgetting [34] によって既学習データの忘却が起こる．この問題はニューラルネットワークを学習させる代

表的な手法である，誤差逆伝播法の学習方式と密接な関連があり，根本的な解決は難しい．

Catastrophic Forgetting を回避する手法としてコンソリデーションラーニングという学習法 [35,36] が提案されているが，この手法を用いた場合にはロボットが実時間内で学習することはできなくなってしまうという問題がある．

1.3.4 機能のまとめ

以上から，目標とすべき学習制御器の機能をまとめる．まず，行動創発，汎化・抽象化，柔軟性の 3 項目に関しては NEAT や SDNN を用いることで両立できる．また，行動創発，汎化・抽象化，オンライン性を両立できる手法として DVB-RL も知られている．そして，これらのいずれもがニューラルネットワークを用いた学習法である．

NEAT や SDNN では遺伝的アルゴリズムに基礎を置いているため，環境，タスク毎に設定が必要となるパラメータをも獲得することができる．ただし，遺伝的アルゴリズムは本来オンライン学習を不得意としており，その欠点が受け継がれている．

DVB-RL は強化学習に基礎を置いており，オンライン学習が可能である．ただし，遺伝的アルゴリズムを用いた場合のような，学習制御器のネットワーク構造や，その他のパラメータをも学習できるほどの柔軟性は持ち合わせてはいない．

そこで，行動創発，汎化・抽象化，柔軟性，オンライン性の四項目を両立するための必要条件として，オンライン・リアルタイムなネットワーク構造の強化学習を行うシステムの構築が求められるが，それができる学習制御器は未だ開発されていない．

よって，自律型ロボットに適した学習制御器を構築するためには，オンライン・リアルタイムなネットワーク構造の強化学習が行える学習制御器を開発したうえで，漸次性の確認やパラメータの簡素化を行っていく必要がある．

1.4 研究目的

本研究では，行動創発，汎化・抽象化，柔軟性，オンライン性，漸次性の五要素を確保できる学習制御手法として，オンライン・リアルタイムなネットワーク構造の強化

学習を実現することを第一の目的とする．また，提案手法に対し漸次性の確認とパラメータの簡素化を施すことで，創発性，適応性，即時性を兼ね備えた自律型ロボットの学習制御手法に関するプラットフォームを構築することを最終目的とする．

1.5 基本構想

以上の目的を達成するために，本研究では新しい学習制御手法として自己組織化回路素子 Self-Organizing Network Elements (SONE) を提案する．以下ではこの手法の根拠となった従来研究に対する考察と，SONE の概要を示す．

1.5.1 ネットワークを使用することの有効性

機械学習において，行動創発を行うためには強化学習を用いることが有効である．しかしながら，強化学習に基づいた学習制御器の構築を基礎とした場合，状態空間の分割法，入力信号の変換法といった問題から，汎化・抽象化や柔軟性を確保することが難しい．

しかしながら，DVB-RL のようにネットワーク型の学習制御器を用いた強化学習法によって，これらの問題に対する有効性が示されている [29]．

先に示したように，従来 Q-learning 等の一般的な強化学習手法では，マルコフ決定過程に従った学習を行うために，設計者が各タスク・環境に応じてロボットの状態空間を分割する必要があった（状態空間の分割問題）．

そこで浅田らは階層型強化学習を提案し，ロボカップ等 [37, 38] で活躍しているサッカーロボットに関して，ゴールまでの状態遷移と動作系列によって状態空間の自動的な分割を実現している [28]．

それに対して伊藤らは，浅田らの手法では，ボールやゴールの大きさや見える位置を，予め用意したプログラムによって視覚センサ信号から計算させているため，タスク・環境に依存した入力信号の変換を必要としていることを指摘しており（入力信号の変換に関する問題），ニューラルネットワークを用いた強化学習を実現することでその問題が解決できることを示している [29]．

そこで本手法でも同様に，ネットワーク型の学習器を用い，それをロボットの入出力と直結する手法をとることで，状態空間の分割や入力信号の変換を必要としない学習制御システムの構築を行い，汎化・抽象化や柔軟性の確保につなげる．

1.5.2 ネットワーク構造の決定法

さらに高い柔軟性を確保するためには，ネットワーク自身の外部パラメータも削減する必要が生じるため，ネットワーク構造の自己組織化を考える必要がある．

先に示した，Miikkulainen らの提案している NEAT は，伊藤らの手法と同様に状態空間の分割や入力信号の変換を必要とせず，さらには設計者によるネットワーク構造の決定すらも必要としない学習制御器が実現できるため，外部パラメータの単純化に関するアドバンテージが大きいと考えられる [30].

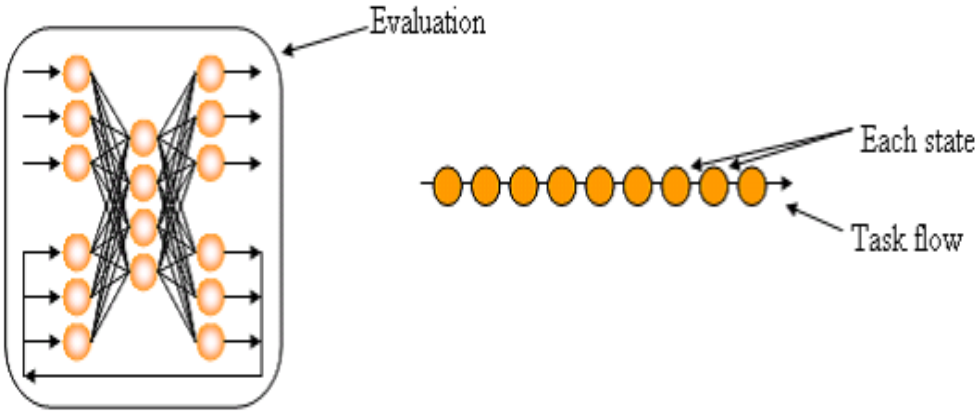
しかしながら，NEAT のような遺伝的アルゴリズムに従った学習ではネットワーク全体を一個体として評価するため，その評価にはタスク・環境に応じた評価時間が必要となる (例えば，[30, 39, 40])．DVB-RL や NEAT のようにネットワークがロボットの入出力と直結される場合，ネットワーク全体が学習するべき対象は，ロボットが学習するべきタスク全体と対応する（図 1.1(a)）．よって，ネットワーク全体の評価を行うためにタスク全体に対する評価を行うだけの評価時間を必要とするという問題が生じる．

一方でネットワーク内の個々の素子を評価する場合，タスク全体に対する評価は必ずしも必要ではない（図 1.1(b)）．一般にネットワーク内の個々の素子はロボットの特定の状態において反応し，その状態に対する適切な出力の形成に寄与する．よってそれらの素子は，素子の代表している個々の状態において評価可能であると考えられる．そこで，本論文ではネットワークの各素子に関する評価値を算出し，その評価値によってネットワーク構造を決定する手法をとり，タスク・環境に応じた評価時間の設定をも必要としない，オンライン・リアルタイムなネットワーク構造の強化学習法として SONE を提案する．

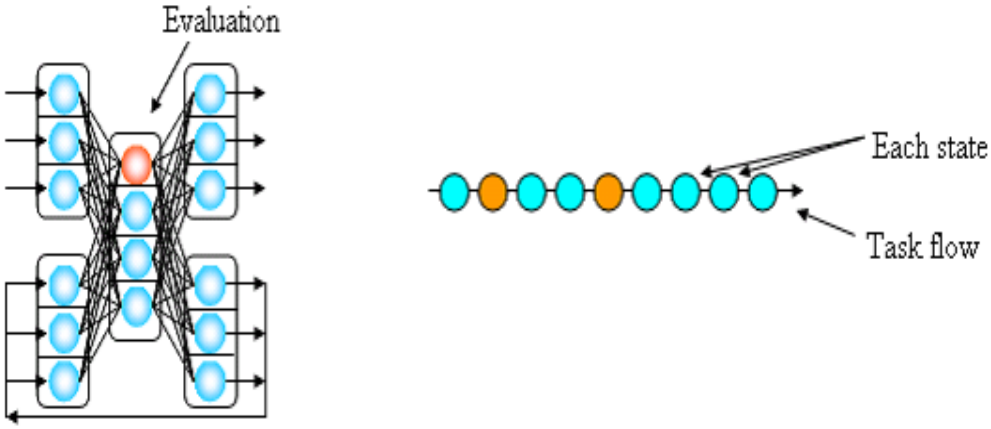
1.6 本論文の構成

以上では，自律型ロボットにおける制御則を設計する際の問題と学習制御を導入する必要性を述べ，自律型ロボットにおいて創発性，適応性，即時性を両立させるという観点から，従来の学習制御器の応用上の問題点を示し，さらにはその問題点を踏まえた新しい学習制御器の基本構想について述べた．

本論文第 2 章ではその基本構想に従い，以上の仕様を満たす学習制御器の構成法として，自己組織化回路素子 Self-Organizing Network Elements (SONE) を提案する．またこの章，提案する学習制御器構成法の具体的な実装例として，自己組織化論理回路の構成法について述べ，自己組織化論理回路の学習に関する数学的な証明，他の学習制御手法との関連について述べる．また第 3 章では，SONE の教師あり学習への応用と，SONE の基本特性解析試験について述べる．この章では，軌道学習に関する試験と二重螺旋問題を用いた試験によって，要求仕様のうち特に汎化・抽象化，柔軟性，オンライン性，漸次性の四項目を中心として試験を行う．軌道学習に関する試験では，SONE の耐ノイズ特性，追加学習特性，時系列学習特性を解析すると伴に，二重螺旋問題を用いた試験によって，汎化能力に関する試験も行う．これらの試験ではリカレントニューラルネットワーク等を用いた学習法との比較を適宜行い，SONE を用いた場合の利点についてまとめる．第 4 章では，行動創発，柔軟性，オンライン性の両立をシミュレーション上の移動ロボットを用いた衝突回避実験によって確認する．第 5 章では，SONE の内部には原理的にノイズが発生するという仮説に対し，SONE を構成する各素子に対するノイズ耐性を高めるための手法を提案する．また，ノイズ対策を施した SONE による移動ロボットの衝突回避実験を行い，その有効性を検証する．第 6 章では本論文の全体に関する考察とまとめ，さらには今後の展望について述べる (図 1.2)．



(a) Evaluation of a network



(b) Evaluation of a part of a network

図 1.1　Evaluation of network

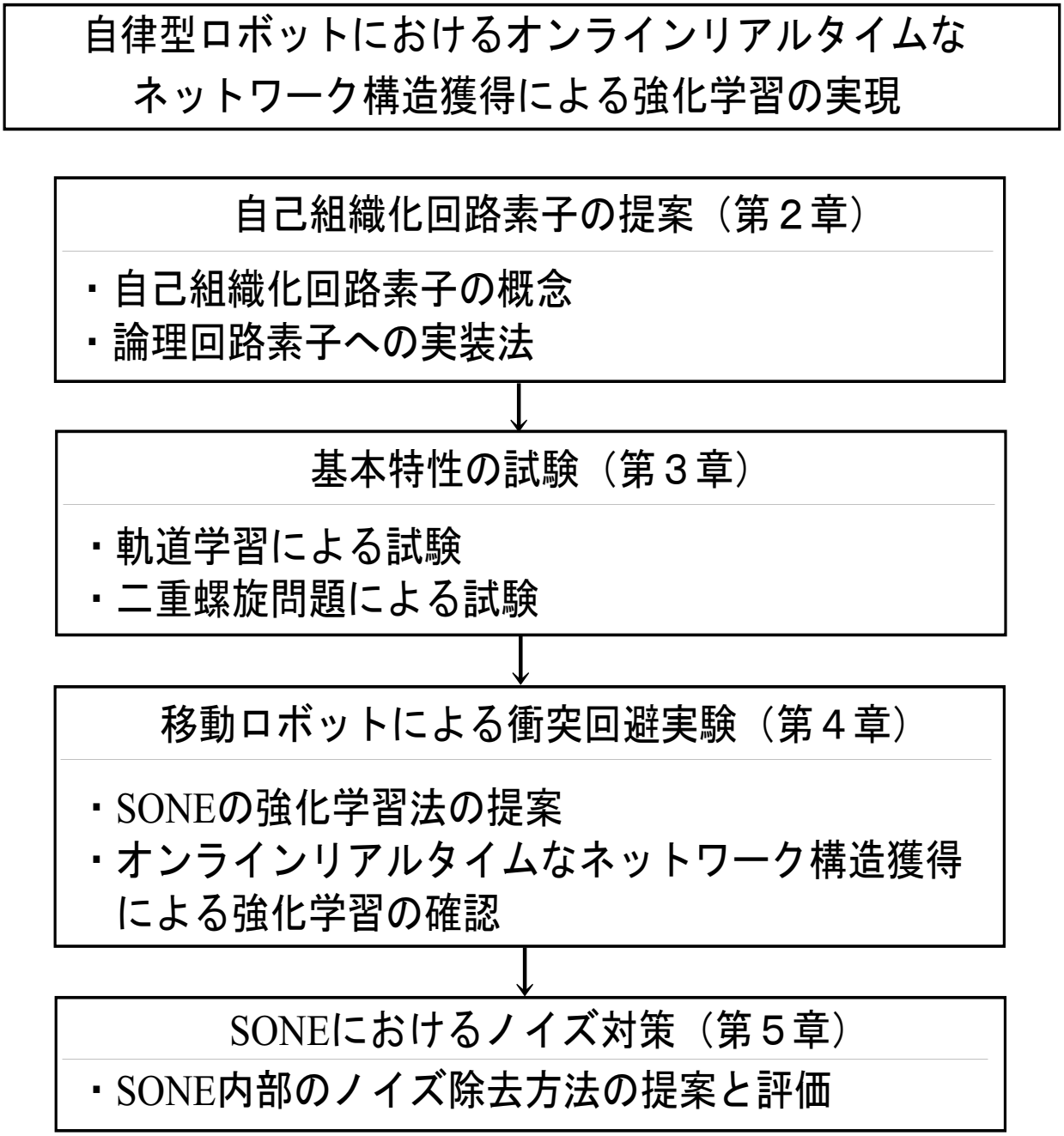


図 1.2 Structure of this thesis

第2章　自己組織化回路素子 Self-Organizing Network Elements (SONE)

ここでは，基本構想から新しい学習制御器の仕様をさらに具体化すると伴に，基本素子として論理回路素子を用いた場合の SONE の実装方法を示す.

2.1　基本概念

前章では，自律型ロボットに必要となる学習制御器の要求仕様（行動創発，汎化・抽象化，柔軟性，オンライン性，漸次性）のうち，行動創発を行うためには強化学習を用いることが有効である．また，汎化・抽象化や柔軟性を確保するためには，ネットワーク型の学習制御器を用いること，さらにはネットワーク構造の学習を行うことが有効である．そしてオンライン性に関しては，ネットワーク上の素子を部分的に評価する技術が必要であるという考えを述べた.

1. 強化学習

2. ネットワーク型の学習制御器

3. ネットワーク構造の学習

4. ネットワーク上の素子の部分評価

これらの仕様を全て満たすための枠組みとして，自己組織化回路素子の概念を提案し，論理回路素子を用いた実装法の中で漸次性の確保やパラメータの簡素化が可能であることを示していく.

本論文で提案する自己組織化回路素子では，ネットワークを構成するために必要となる素子を独立したプロセス，または独立したエージェントとして記述する．また，各素子が持つ機能として，以下の機能を含めるものとする．

1. 強化信号伝播
2. 自己解体
3. 新しい素子の生成

各素子は，強化信号（報酬値）をより多く受け取るための学習である，強化学習を行うものとする．ここでは，強化学習を行うために，各素子に強化信号伝播機能を持たせている．この強化信号伝播機能によってネットワーク上の素子は互いに評価値を算出し合い，互いの重要度を決定できる．

さらに，強化信号をより多く受け取れるようにネットワークの構造を変更する機能として，自己解体機能，新しい素子の生成機能を持たせている．これらの機能は，先の強化信号伝播規則によって決定した評価値をもとに実行され，素子単位でのネットワーク構造の生成と淘汰を実現するための機能である．

筆者らは，これらの機能によって，オンライン・リアルタイムなネットワーク構造の強化学習が実現できると考えた．

この枠組みでは，ネットワーク上の各素子を強化信号（報酬）に対して貪欲（Greedy）な学習制御器として設計することで，ネットワーク全体もまた，強化信号の増大を計ることができる学習制御器となる．このような強化学習が，ネットワーク構造の素子単位の自己組織化に基いて行われることで，遺伝的アルゴリズムに基いたアプローチでは難しかったオンライン・リアルタイム学習も実現できと考えられる．

また，SONE の設計は報酬量を安定的に増大させる回路素子の設計に帰着し，その設計がタスク・環境と独立したものとして扱えるという利点もあるため，汎用的な学習システムの構築が期待できる．

2.2 自己組織化論理回路

SONE を実装するための素子には多様な選択が考えられる．本研究では，その中でも比較的容易な対象である 2 値による論理回路素子へ SONE を実装した．SONE の概念を論理回路へ適用した自己組織化論理回路は And ノード，Or ノード，反転リンク，非反転リンクより構成される．ここでは，自己組織化論理回路を構成するうえでの基礎となる Or ノード，非反転リンクの構成法について述べた後，その他の素子に関する構成法を述べる．

2.2.1 Or ノードの構成法

図 2.1(a) に示される Or ノードは，1 本のテストリンクと 2 本の実リンクを持っており，リンクからの入力はそれぞれ X_T，$X(1)$，$X(2)$ で与えられる．テストリンクは各 Or ノードにつき必ず 1 本存在するが，実リンクの本数には制限が無く可変であり，一般に N 本の実リンクを持つことができる．

SONE を構成する素子は，ネットワーク出力を生成するための出力フェイズ，強化信号を伝播し，各素子の評価値を算出するための伝播フェイズ，そして評価値に従ってネットワーク構造を変更するための構造変更フェイズの三つのフェイズによってそれぞれの機能を実行する．

1. 出力フェイズ
2. 伝播フェイズ
3. 構造変更フェイズ

2.2.1.1 出力フェイズ

出力生成時（出力フェイズ）には各 Or ノードは実リンクに対し OR 演算を行うことで出力 $Y = \bigcup_{i=1}^{N} X(i)$ を計算する．

2.2.1.2 伝播フェイズ

強化信号伝播時（伝播フェイズ）には各 Or ノードは表 2.1 に示されるルールに従って各リンクとその入力側ノードに対しそれぞれ強化信号 R_1, R_2 を伝播する．表 2.1 の各 Case は次のように機能する．例えば Or ノードが 3 本の実リンクを保持しており，それらの入力が $\{X(1), X(2), X(3)\} = \{T, T, F\}$ であるとする．このとき，Or ノードの出力は $Y = T$ となる．Or ノードがこの出力を行った結果負の強化信号 $R < 0$ を受け取った場合，F の出力を行っているリンク 3 は Case1 に相当し，T の出力を行っているリンク 1, 2 は Case5 に相当する（表 2.1 において N_T は実リンクのうち T を出力するリンクの数として計算される）．各リンクは各 Case に従って算出された R_1 を受け取り，各リンクの入力側ノードに R_2 を伝達する．R_1, R_2 を伝達された素子はそれらの信号を蓄え，自らの評価値である R 値にこれを加える．また，強化信号を伝達したノードは R 値を 0 にリセットする．テストリンクには，テストリンクが昇格，実用化された場合を想定して Or ノードの出力の算出を行い，表 2.1 を適用する．ただし，テストリンクの昇格によって Or ノードの出力が反転する場合には表**??**の結果算出される R_1 に-1 を乗じる．また R_2 は常に 0 とする．

2.2.1.3 構造変更フェイズ

構造変更時（構造変更フェイズ）にはテストリンクの昇格判定が行われ，テストリンクの R 値がある閾値 $Th1$ を上回る場合，テストリンクの昇格，実用化を行う．また，Or ノードの保持する実リンク数が 1 以下である場合には Or ノードは入出力の演算を保つようにネットワークを適宜つなぎ直し，自己解体する．

2.2.2 非反転リンクの構成法

図 2.1(b) に示される非反転リンクは，一つのテストノードを持っている．このテストノードは And ノード，Or ノードの二通りの中から，非反転リンクの出力側ノードの種類と逆のノードを備えるように生成される．テストノードはさらに二本のテストリンク（TL1,TL2) を保持しており，TL1 は非反転リンクの入力側ノードに結合すること

で，非反転リンクと同様の入力を得る．また，TL2 はネットワーク内にある他のノードと結合している．

2.2.2.1 出力フェイズ

出力フェイズには各非反転リンクは入力をそのまま出力として伝える（$Y = X$）．

2.2.2.2 伝播フェイズ

伝播フェイズには各非反転リンクは表 2.2 に従って R_T を計算し，テストノードへと伝播する．テストノードは伝播された R_T を用いて自らの伝播規則（例えばテストノードが Or 素子なら Or 素子の規則）を用いて TL2 に伝播する．TL2 は伝播された強化信号を自らの R 値に加える．

2.2.2.3 構造変更フェイズ

構造変更フェイズにはテストノードの昇格判定が行われ，非反転リンクの R 値がある閾値 $Th2$ を上回りかつ TL2 の R 値がある閾値 $Th3$ を上回れば，テストノードを昇格，実用化し，不要となった非反転リンクは自己解体を行う．また，非反転リンクの R 値が 0 を下回った場合には非反転リンクは自己解体する．

2.2.3 その他の素子の構成法

以上では，Or ノードと非反転リンクに関する SONE の構成を示した．今回，自己組織化論理回路に使用している他の素子は，この二つの素子の構成法をもとにして容易に実装できる．まず，Or ノードと And ノードはド・モルガンの法則を用いることで互いに変換が可能である．例えば And ノードは，Or ノードの周囲に結合しているリンクの全てを反転処理した素子として捉えることができる．これによって，And ノードは Or ノードを基に容易に作成できる．また，反転リンクも非反転リンクを反転処理した素子として捉えることができ，同様の理由から容易に作成できる．

表 2.1 Reinforcement signal propagation rule for or-node

$Case1:$	$(Y = T) \wedge (X(k) = F)$
	$R_1(k) = 0, R_2(k) = 0$
$Case2:$	$Y = F$
	$R_1(k) = R/N, R_2(k) = R/N$
$Case3:$	$(Y = T) \wedge (N_T = 1) \wedge (X(k) = T)$
	$R_1(k) = R, R_2(k) = R$
$Case4:$	$(Y = T) \wedge (N_T \neq 1) \wedge (R \geq 0) \wedge (X(k) = T)$
	$R_1(k) = -R \times (N_T - 2)/N, R_2(k) = 0$
$Case5:$	$(Y = T) \wedge (N_T \neq 1) \wedge (R \geq 0) \wedge (X(k) = T)$
	$R_1(k) = R \times N_T/N, R_2(k) = 0$

表 2.2 Reinforcement signal propagation rule for non-inverted link

$Case1:$	$(R > 0) \wedge (Y_T = Y)$
	$R_T = 0$
$Case2:$	$(R > 0) \wedge (Y_T \neq Y)$
	Reconstructing TL2
$Case3:$	$(R \leq 0) \wedge (Y_T = Y)$
	$R_T = R_1$
$Case4:$	$(R \leq 0) \wedge (Y_T \neq Y)$
	$R_T = -R_1$

2.2.4　ネットワークの構成法

これらの素子の出力フェイズ，学習フェイズ，構造変更フェイズを用いてネットワークの自己組織化を行うことができる．まず，ネットワークの初期状態としてセンサ入力を受け付けるための入力ノードとモータ出力を行うための出力ノードを，ロボットに応じて必要な数用意する．ただし，これらのノードはOr ノードを用いて構成し，自己解体は不可とする．リスト構造を用いてこれらノードの管理を行い，リストの前方には入力ノード，後方には出力ノードを配置する．新しくできたノード（中間ノード）はその出力側に位置するノードの直前に挿入されることで，リストへ登録される．

ネットワークの出力計算の際には，このリストの前方から順に出力フェイズによってノードの起動を行い各ノードの出力を計算する．ネットワークに強化信号を伝播す

表 2.3 Calculation direction of the list

実行する演算	演算方向
出力計算	前方から後方
強化信号伝播	後方から前方
構造変更	順序を問わない

る際には，リストの後方から前方へ向かって順に伝播フェイズによってノードの起動を行い各ノードによる信号の伝播を行う．ネットワークの構造変更を行う際には，全てのノード，リンクにおける構造変更フェイズを起動するが，この際にはその順序を問わない（表 2.3）．ネットワークの計算は出力計算，強化信号伝播，構造変更の順に繰り返し行われ，ロボットの行動する全てのタイムステップに関してネットワークの出力，学習，構造変更が行える（図 2.2）．

2.3 強化信号伝播規則の作成法

ここでは，先に示した強化信号伝播規則（表 2.1,2.2）を作成するための方法について説明する．本論文で実装した強化信号伝播規則は主に，ネットワークに学習効果を与えるための設計指針とネットワークの冗長性を抑制するための設計指針より成り立っている．ただし，これらの指針は本研究での予備実験にる経験則によるものが多い．

2.3.1 予備実験

SONE に用いる素子の強化信号伝播規則を定める前に，本研究では二つの単純な規則に関する予備実験を行った．

2.3.1.1 等分配

ひとつめの規則は素子が受け取った強化信号を，出力に関与のある他の素子に対して等しく分配するという規則である．表 2.4 に Or 素子の入力が 3bit である場合の例を示す．

表 2.4 Equal distribution

$x(1)$	$x(2)$	$x(3)$	y	R	$R_1(1)$	$R_1(2)$	$R_1(3)$
T	T	T	T	1	$1/3$	$1/3$	$1/3$
T	T	F	T	1	$1/2$	$1/2$	0
T	F	T	T	1	$1/2$	0	$1/2$
T	F	F	T	1	1	0	0
F	T	T	T	1	0	$1/2$	$1/2$
F	T	F	T	1	0	1	0
F	F	T	T	1	0	0	1
F	F	F	T	1	$1/3$	$1/3$	$1/3$
T	T	T	T	-1	$-1/3$	$-1/3$	$-1/3$
T	T	F	T	-1	$-1/2$	$-1/2$	0
T	F	T	T	-1	$-1/2$	0	$-1/2$
T	F	F	T	-1	-1	0	0
F	T	T	T	-1	0	$-1/2$	$-1/2$
F	T	F	T	-1	0	-1	0
F	F	T	T	-1	0	0	-1
F	F	F	T	-1	$-1/3$	$-1/3$	$-1/3$

この強化信号伝播規則を用いてネットワークに 2bit 演算を学習させる実験を行うと（参照：第 5 章 3-bit 演算に関する実験），一応の学習効果が確認できる．しかしながら，冗長な回路構造が多数発生するうえに学習にも時間がかかる．この理由は，この規則では冗長性を抑制する機構が備わっていないことによるものである．例えば，表 2.4 に示した x(1) と x(2) が常に同じ値しかとらないとしよう．その場合，x(1) と x(2) が受け取る強化信号の総和は常に等しくなり，x(1) が生き残ることができるならば，x(2) も生き残ることになる．この種の冗長なリンクは同時にいくつでも存在し得るため，回路は無制限に冗長性な構造を増やし続けてしまう．

2.3.1.2 集中分配

次に試す規則は，等分配の規則に変更を施し，正の強化信号（$R \geq 0$) を得た場合には特定の素子にだけ伝播するというものである．等分配の場合と同様に表 2.5 にその例

表 2.5 Concentlate distribution

$x(1)$	$x(2)$	$x(3)$	y	R	$R_1(1)$	$R_1(2)$	$R_1(3)$
T	T	T	T	1	1	0	0
T	T	F	T	1	1	0	0
T	F	T	T	1	1	0	0
T	F	F	T	1	1	0	0
F	T	T	T	1	0	1	0
F	T	F	T	1	0	1	0
F	F	T	T	1	0	0	1
F	F	F	T	1	1	0	0
T	T	T	T	-1	$-1/3$	$-1/3$	$-1/3$
T	T	F	T	-1	$-1/2$	$-1/2$	0
T	F	T	T	-1	$-1/2$	0	$-1/2$
T	F	F	T	-1	-1	0	0
F	T	T	T	-1	0	$-1/2$	$-1/2$
F	T	F	T	-1	0	-1	0
F	F	T	T	-1	0	0	-1
F	F	F	T	-1	$-1/3$	$-1/3$	$-1/3$

を示す.

集中分配を用いた場合には等分配の場合のような冗長な素子は残らない．なぜならば，仮に x(1) と x(2) が同じ値しかとらないとしても，報酬が得られる場合には常に x(1) が優先するため x(2) に相当するリンクと区別できる.

ただし，この規則を用いて等分配の場合と同じ 2bit 演算に関する実験を行うと，学習できない演算が存在することがわかる.

2.3.2 学習効果を与えるための設計指針

強化学習では，学習制御器に外界から与えられる強化信号（報酬）を増大するための学習を学習制御器が行うことで，環境・タスクに即したロボット制御が実現できる．そして，SONE のようなネットワークの構造変更によって学習する学習制御器では，その構造変更が強化信号の増大を考慮して行われる必要がある.

ネットワークの構造変更によって強化信号の増大を保障するために，本論文では予備実験の結果を踏まえ，次のような指針を立てた．

1. 各素子が自らが受け取った強化信号を他の素子へ伝播する際に，受け取った強化信号以上の信号を周囲の素子へ与えることを禁止する．
2. 受け取る強化信号の期待値が 0 を上回るテスト用素子を実用化する．
3. 受け取る強化信号の期待値が 0 を下回る素子は削除する．

まず指針 1 によって，ネットワーク内部の総報酬量が自己完結的に発散することを防ぐ．これによってネットワークは，報酬を増大させるための信号をネットワーク外部に求めることになり，外部環境に対して意味のある入出力を実現して報酬を得なければならなくなる．

次に指針 1 のうえで，指針 2 に従って新たな素子を生成することで，ネットワークの外界から得られる強化信号の期待値を増大することが期待できる．さらに指針 3 に従って素子を淘汰することで，ネットワークの外界から与えられる負の強化信号（罰）の量を抑えるように学習を行うことが期待できる．

2.3.3　回路の冗長性を抑制するための設計指針

SONE のようなネットワークの構造変更によって学習する学習制御器では，回路の冗長性を抑えることでネットワークのノード数，リンク数が発散することを防ぐ必要がある．

そこで，冗長と思われる出力を行っている素子に対し罰の信号を与えることを行う．ここでは，ネットワーク内の各ノードが，その入力を照らし合わせ，冗長と思われる入力を発見した場合には，それらの入力を行っているリンクに対し罰の信号を与えることで，冗長性の解消が期待できる．

2.3.4 数学的な記述と Or ノードの強化信伝播規則

ここでは，以上の設計指針を数学的に記述すると伴に，実際の強化信号伝播規則を作成する．まずは，Or ノードに関する強化信号伝播規則を考える．

自己組織化論理回路のネットワーク内に存在するノード数を N_{Net}，ネットワークのとり得る全状態を $S := \{T, F\}^{N_{Net}}$，ネットワーク上のある着目する Or ノード α が保持する i 番目のリンクが T（真値）を出力するような，ネットワークの状態を $L_T(i) := \{l | l \in S, x(i) = T\}$，リンク i を除いた実リンク群の Or をとることにより T が得られるような，ネットワークの状態を $L'_T(i) := \cup_{j \neq i} L_T(j)$，ノード α にとって，T の出力が正解となるネットワークの状態を $A_T := \{a | a \in S, y = T, R > 0\}$ と定義し，A_T，L_T，L'_T によって区切られる S 上の状態をそれぞれ状態 $A - H$ とする (図 2.3) ．このとき，Or ノード α の学習の目標状態は以下のようになる．

$$\cup_i L_T(i) = L_T(i) \cup L'_T(i) = A_T \tag{2.1}$$

この図 2.3 を基に，ノード α が保持するリンク i と，リンク i の入力側ノードに対する強化信号伝播法を記述していく．

2.3.4.1 ネットワークの状態 B,E に関する規則

まず，状態 B，状態 E ではリンク i，リンク i の入力側ノードのいずれにも強化信号を伝播しない．この状態ではリンク i は出力として F を提示しているにも関わらず，他のリンク郡によって Or ノード α の出力は T となっている．よって，この状態ではリンク i は Or ノード α の出力決定に対して寄与していないと考え，強化信号を伝播しないこととする（図 2.3 Case1).

2.3.4.2 ネットワークの状態 D,G に関する規則

次に，状態 D，状態 G では Or ノード α の入力側リンクの中で，リンク i だけが T を示しており，その結果 Or ノード α の出力が T となっている．この状態では，Or ノード α の出力は完全にリンク i によって決定されているため，Or ノードの受け取った強

化信号をそのままの値で伝播する (図 2.3 Case3)．ここでは，リンク i の入力側ノードへの信号を R とすることで，ネットワーク内部の総報酬量が自己完結的に発散することを防ぐことができる．

2.3.4.3　ネットワークの状態 A,H に関する規則

そして，状態 A，状態 H では Or ノード α の入力側リンクは全て F を示しており，リンク i の出力もまた F である．この状態では，全てのリンクが Or ノード α の出力に対して等しく貢献していると考えられるため，全てのリンクとその入力側ノードに対し R/N を割り振る（N: Or ノード α の入力側実リンク数）．ここでも，R を Or ノード α の持つリンク数で除算することで，ネットワーク内部の総報酬量の自己完結的な発散を防いでいる (図 2.3 Case2)．

2.3.4.4　ネットワークの状態 C,F に関する規則

最後に，状態 C，状態 F では Or ノード α は入力側リンクから複数の T 信号を受け取っており，リンク i もまた T 信号を発信している．そしてこれにより Or ノード α は T の出力を行っている．しかし本来 Or ノード α が T の出力を行うためには一つの T 信号が入力されれば充分であるため，これら複数の T 信号を発信しているリンクは冗長である可能性がある．そこで，状態 C では各リンクに $-R \times (N_T - 2)/N$ として罰の信号を伝達する．また，状態 F においても各リンクに $R \times N_T/N$ としてやはり罰の信号を伝達する．この冗長性はリンクの結合に関する冗長性であると考え，リンクの入力側ノードに関しては強化信号を伝播しない (図 2.3 Case4,5)．

2.3.5　Or ノードの強化信号伝播規則に対する証明

本節では，作成した強化信号伝播規則が学習効果をもたらすこと，さらには冗長なリンク数を根号オーダに抑えることができるということに関する証明を行う．

あるネットワークの状態 x において，Or ノード α の持つリンク i が 1 ステップ間に得る強化信号の期待値を $Er(x,i)$ と定義し，$Er(X,i) := \sum_{x \in X} Er(x,i)$ $(X = \{A, B, \cdots, H\})$

とする．

このとき，リンク i が淘汰されずにネットワークに保持される条件は，リンク i が 1 ステップ間に得る強化信号の期待値 $Er(i)$ は表 2.1 を用いて次のように計算できる．

$$\begin{aligned} Er(i) = \ & \frac{1}{N}Er(A,i) + \sum_{x\in C} -\frac{N_T(x,N)-1}{N}Er(x,i) \\ & + Er(D,i) + \sum_{x\in F}\frac{N_T(x,N)}{N}Er(x,i) + Er(G,i) + \frac{1}{N}Er(H,i) \geq 0 \end{aligned} \quad (2.2)$$

ここで，図 2.3 における A_T と，現在の状態 x の関係によって $Er(x,i)$ の正負が決定できる．

$$\begin{aligned} & -\frac{1}{N}|Er(A,i)| + \sum_{x\in C}\frac{N_T(x,N)-1}{N}|Er(x,i)| \\ & + |Er(D,i)| - \sum_{x\in F}\frac{N_T(x,N)}{N}|Er(x,i)| - |Er(G,i)| + \frac{1}{N}|Er(H,i)| \geq 0 \end{aligned} \quad (2.3)$$

さらに，Or ノードのリンク数が充分に多い状態 $(N \to \infty)$ において以下の式が成り立つ．

$$\begin{aligned} & \lim_{N\to\infty}\{\sum_{x\in C} -\frac{N_T(x,N)-1}{N}|Er(x,i)| - \sum_{x\in F}\frac{N_T(x,N)}{N}|Er(x,i)|\} \\ & + |Er(D,i)| - |Er(G,i)| \geq 0 \end{aligned} \quad (2.4)$$

$N_T(x,N)$ は 2 以上の整数値であるため，この式の初項は常に負である．よって，$|Er(D,i)| \geq |Er(G,i)|$ を導くことができる．この式は状態 D によって得られる報酬量が，状態 G によって得られる報酬量を上回るという条件を示しており，図 2.3 において，各状態 $A-H$ に割り当てられた面積が得られる強化信号の期待値に比例して描かれるとするならば，D の面積が G の面積よりも広くなることを示している．

よって，生存可能なリンクは Or ノードの受け取る報酬量の増大に貢献しており，$\cup_i L_T(i)$ が A_T をより良く近似するために貢献しているということが証明できた．

さらに，ある Or ノードの全てのリンクが 1 ステップ間に受け取る強化信号の総和に対する期待値を Er_{All} とすると，全てのリンクが生存する条件は $Er_{All} \geq 0$ であり，以下のように書ける．

$$\begin{aligned} Er_{All} = & \sum_{i=0}^{N} \frac{1}{N} Er(A,i) + \sum_{i=0}^{N} Er(D,i) + \sum_{i=0}^{N} Er(G,i) + \sum_{i=0}^{N} \frac{1}{N} Er(H,i) \\ & + \sum_{x \in C} \frac{N_T(N,x)^2 - N_T(N,x)}{N} Er(x) + \sum_{x \in F} \frac{N_T(N,x)^2}{N} Er(x) \geq 0 \end{aligned} \tag{2.5}$$

この式を $Er(i)$ の絶対値を考慮して変形すると以下のようになる．

$$\begin{aligned} Er_{All} = & -\sum_{i=0}^{N} \frac{1}{N} |Er(A,i)| + \sum_{i=0}^{N} |Er(D,i)| - |\sum_{i=0}^{N} Er(G,i)| + |\sum_{i=0}^{N} \frac{1}{N} Er(H,i)| \\ & - \sum_{x \in C} \frac{N_T(N,x)^2 - N_T(N,x)}{N} |Er(x)| - \sum_{x \in F} \frac{N_T(N,x)^2}{N} |Er(x)| \geq 0 \end{aligned} \tag{2.6}$$

さらにこの式を変形させることを考える．状態 D と状態 G では強化信号を受け取る入力側リンクが 1 本のみ存在する．ここで，そのリンクのインデックスを i^* とすると，$\sum_{i=0}^{N} Er(D,i) = Er(D,i^*)$ さらには $\sum_{i=0}^{N} Er(G,i) = Er(G,i^*)$ が成り立ち，以下のように変形できる．

$$\begin{aligned} & |\sum_{x \in C} \frac{N_T(N,x)^2 - N_T(N,x)}{N} Er(x)| + |\sum_{x \in F} \frac{N_T(N,x)^2}{N} Er(x)| \\ & \leq |Er(H,i)| - |Er(A,i)| + |Er(G,i^*)| - |Er(D,i^*)| \end{aligned} \tag{2.7}$$

ここで，N に対して上式の右辺は定数である．

$$|Er(H,i)| - |Er(A,i)| + |Er(G,i^*)| - |Er(D,i^*)| = Constant \tag{2.8}$$

$$|\sum_{x \in C} \frac{N_T(N,x)^2 - N_T(N,x)}{N} Er(x)| + |\sum_{x \in F} \frac{N_T(N,x)^2}{N} Er(x)| \leq Constant \tag{2.9}$$

この式より，Or ノードが十分なリンク数を備えているとき $(N \to \infty)$ $N_T(N,x)$ のオーダは $\sqrt{N}$ 以下であるということがいえる．つまり，次のように変形ができるため，

$$|\sum_{x\in C}\frac{N_T(N,x)^2 - N_T(N,x)}{N}Er(x)| \leq Constant \tag{2.10}$$

$$|\sum_{x\in F}\frac{N_T(N,x)^2}{N}Er(x)| \leq Constant \tag{2.11}$$

以下のようにして $N_T(N,x)$ のオーダが求められる．

$$|\sum_{x\in F}\frac{N_T(N,x)^2}{N}Er(x)| \leq Constant \tag{2.12}$$

$$\sum_{x\in F}|\frac{N_T(N,x)^2}{N}Er(x)| \leq Constant \tag{2.13}$$

ここで，全ての x に対して

$$\frac{N_T(N,x)^2}{N} \leq Constant \tag{2.14}$$

$$N_T(N,x) \leq Constant\sqrt{N} \tag{2.15}$$

式 (2.10) についても同様であり，冗長なリンク数が根号オーダで抑制できることが示せた．

2.3.6 他の素子への拡張

以上では，Or ノードに関する強化信号伝播規則の構成法について述べた．本節では，Or ノード以外の素子（And ノード，反転リンク，非反転リンク）に関する強化信号伝播規則の構成法について述べる．

まず，Or ノードと And ノードはド・モルガンの法則によって同一の素子と見なすことができる．具体的には Or ノードへ結合している全てのリンクに対して反転処理（図 2.4）を施すと And ノードが作成できる．以上で作成した Or ノードの強化信号伝播規則は，入力側または出力側に反転・非反転のいずれのリンクが結合するかを問わず使用できるため，この反転処理を用いて And ノードにも適用可能である．

次に，非反転リンクに関する強化信号伝播規則の構成法を考える．非反転リンクの強化信号伝播規則は，非反転リンクが所有するテスト用ノードを評価するために使用する規則である．非反転リンクは Or ノードの入力側，出力側に 1 本の非反転結合が有る場合と同じ働きをする素子であるため，非反転リンクの強化信号伝播規則は，Or ノードの入力側，出力側に 1 本の非反転結合がある場合の，Or ノードの強化信号伝播規則より作成できる．本論文で使用している非反転リンクの強化信号伝播規則は，Or ノードの強化信号伝播規則より導き出される規則に対し，テスト用ノードへの淘汰圧をさらに高めるようにして実装している．

そして，反転リンクに対しては，非反転リンクを単に反転させただけなので，非反転リンクの強化信号伝播規則より容易に伝播規則を作成できる．

2.4　他の構造学習法との関連

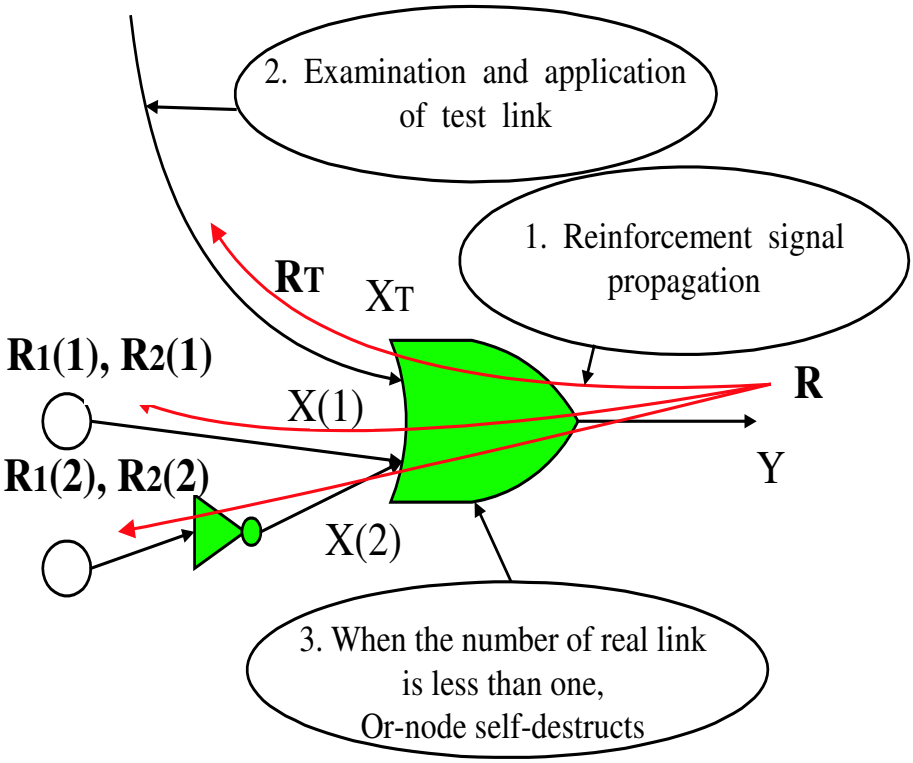
本節では，SONE に比較的関連の深い学習法と SONE との関連や違いについて述べることで，SONE の新規性に関する議論の補完を行う．

従来，ニューラルネットワークの分野においても，このようなネットワークの各素子を評価してトポロジーを決定する手法が存在する [41–47]．しかしながら，このように著者の行ってきた調査の範囲においても，トポロジー決定によるオンライン学習と強化学習の両立が実現できるシステムは発見できなかった．

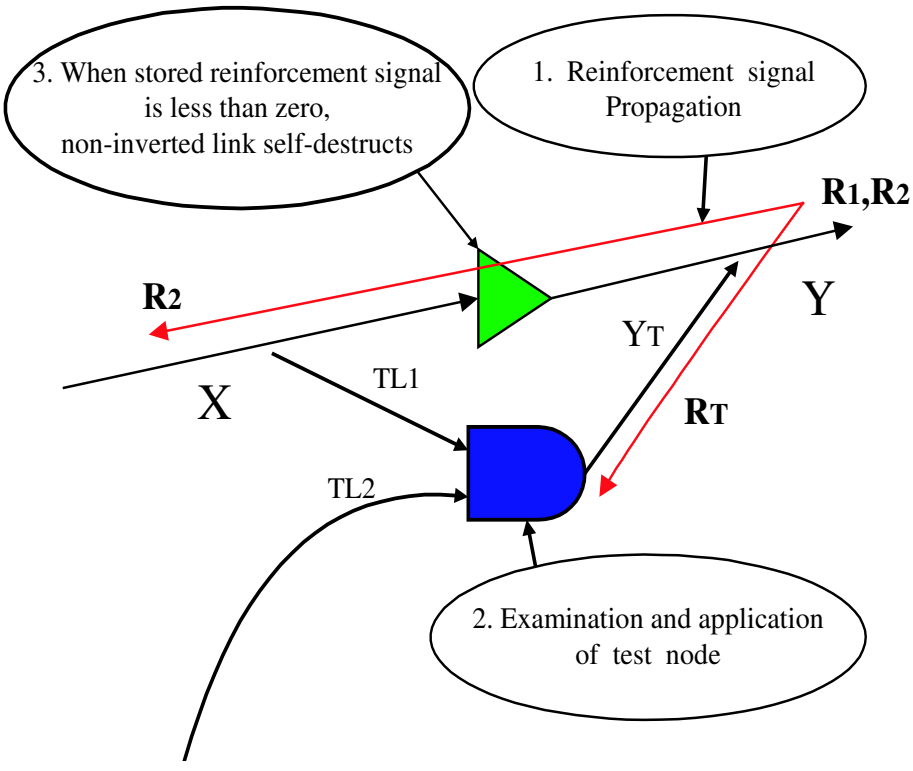
それに対して本論文で提案する手法では，外部から与えられる強化信号を各素子毎の評価値に反映するための強化信号伝播規則を導入し，ネットワークの出力層から各素子へと強化信号を伝播することで各素子の評価を行っている．これによって，オンライン・リアルタイムなネットワーク構造の強化学習を実現できる．

また福永らによって，強化学習によりネットワーク構造を決定する方法も提案されており，自律型移動ロボットの制御に用いるネットワークのリンク構造の学習ができている [48]．しかしながらこの方法では，三層型のネットワークに限定されており，ノードも含めたネットワークの構造を自己組織化するには至っていない．本論文で提案する手法では，ネットワーク上のノード，リンクのいずれについても自己組織化することができる．

そして Teuscher らは，2 値演算ネットワークである，Random Boolean Networks (RBNs) においてタスクを学習するためのローカルルールを提案している．しかしながら，ここで提案されている手法はネットワークノードが出力する値の平均を振動させる，またはある値へ近づけるというタスクへの応用に留まっており，複雑な入出力関係を学習することは難しいといえる [49].



(a) Or-node



(b) Non-inverted link

図 2.1 Elements

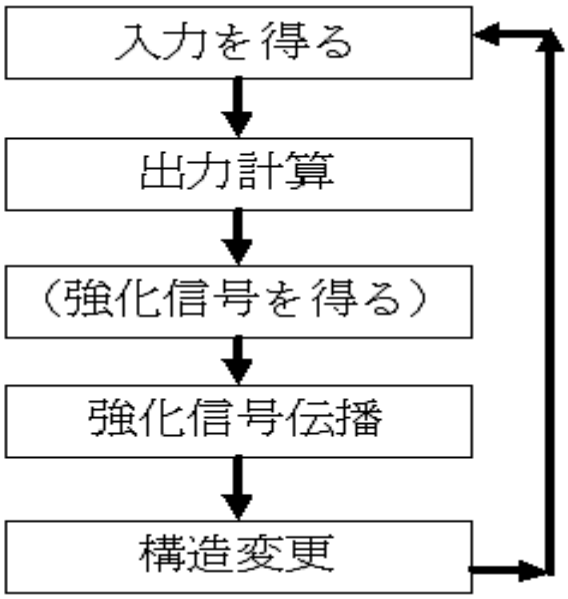


図 2.2 Basic learning cycle

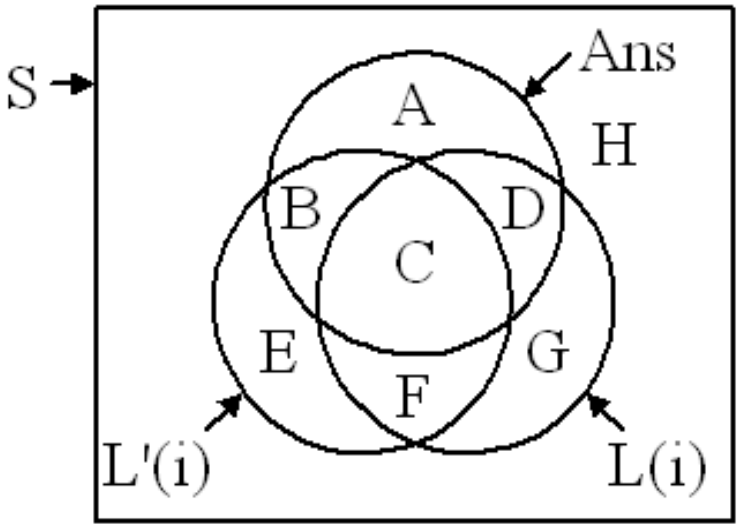


図 2.3 State space

表 2.6 Definition

記号	数学的な定義	定義
T		論理演算における真値 (True)
F		論理演算における偽値 (False)
α		Or ノードのインデックス
i		ノード α 内の入力側実リンクのインデックス
N		ノード α の持つ入力側実リンク数
y		ノード α の出力
R		ノード α の受け取った強化信号
$x(i)$		ノード α の持つリンク i の出力
$A-H$		A_T，L_T，L'_T によって区切られる S 上の状態 (図 2.3 参照)
N_{Net}		ネットワーク内に存在する全ノード数
S	$\{T,F\}^{N_{Net}}$	ネットワークのとり得る全状態
x	$\{x\|x \in S\}$	ネットワークのある状態
X	$\{A,B,\cdots,H\}$	A から H のいずれか
$L_T(i)$	$\{l\|l \in S, x(i)=T\}$	ノード α が保持するリンク i が T を出力するような，ネットワークの状態
$L'_T(i)$	$\cup_{j\neq i} L_T(j)$	ノード α において，リンク i を除いた入力側実リンク群の出力の Or をとることにより T が得られるような，ネットワークの状態
A_T	$\{a\|a \in S, y=T, R>0\}$	ノード α にとって，T の出力が正解となるネットワークの状態
$Er(x,i)$		ネットワークが状態 x をとって，ノード α の持つリンク i が 1 ステップ間に得るであろう強化信号の期待値
$Er(X,i)$	$\sum_{x\in X} Er(x,i)$	ネットワークが状態 x をとって，ノード α の持つリンク i が 1 ステップ間に得るであろう強化信号の期待値の状態 X に対する総和
$Er(i)$	$\sum_{x\in S} Er(x,i)$	リンク i が 1 ステップ間に得る強化信号の期待値
Er_{All}	$\sum_{i=0}^{N}\sum_{x\in S} Er(x,i)$	ノード α の全てのリンクが 1 ステップ間に受け取る強化信号の総和に対する期待値

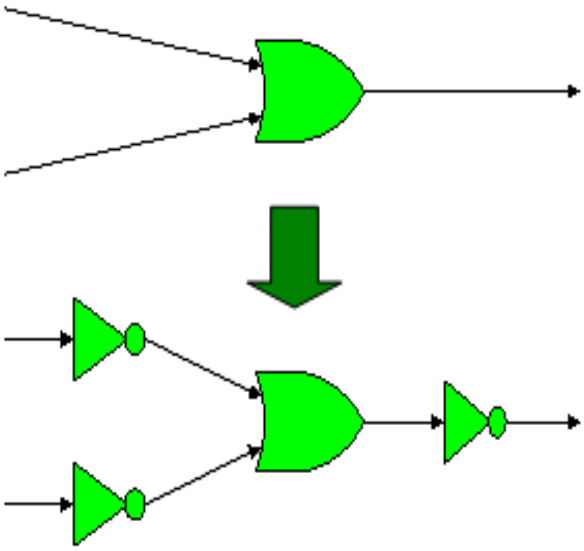


図 2.4 Reverce process

第3章　基本特性の試験

本章では，前章で構成したSONEの基本特性を明らかとするための試験を行う．SONEは，効果的な出力の自律的探索を行うために，強化学習を行うことを考慮して作成されているが，強化学習を行う場合，学習制御器の定量的な評価を行うための問題設定が難しい．そこで本章では，教師あり学習を用いて試験を行い，次章における強化学習を用いた実験の足がかりとする．具体的には，軌道学習問題やTwo-spiral問題へSONEを適用することにより，SONEの汎化・抽象化，柔軟性，オンライン性，漸次性に関する検証を行っていく．

3.1　SONEによる教師あり学習

一般に教師あり学習では，正解となる入出力は学習制御器の外部から与えられる．そして，学習制御器はその入出力信号に応じて学習を行う．教師あり学習は，バッチ学習とオンライン学習に大別され，本論文では，SONEを用いて教師あり学習をオンラインに行うこととする．

前章で示したように，SONEの各素子は報酬に対し貪欲（Greedy）に設計されており，素子の受け取る報酬量を増大させるようにローカルネットワークの構造変更を促すことができる．そこで，本研究ではこの性質を用いて，次のように教師あり学習を実現した．

図3.1に教師あり学習のフローを示す．まず，SONEが現在のネットワーク構造によって，正解の入出力（教師データ）を表現できるか否かを検証するために，SONEの入力ノードへ教師データより入力信号を与える．次に，この入力信号をもとに，ネットワークに出力を算出させ，得られたネットワークの出力信号と教師データの出力信号とを比較する．この比較の結果，出力ノードの出力信号と教師データの出力信号に差

違が無い場合にはその出力ノードへ報酬として1を差違が有る場合にはその出力ノードへ罰として-1の強化信号を付与する（図3.2)．さらに，この付与された強化信号を強化信号伝播法を用いてネットワーク内の各素子に伝播する．そして，伝播した報酬・罰に応じてネットワークの構造変更を行う．これを教師データを一巡するまで繰り返し，学習が充分であるかを確かめる．学習が不十分である場合には，最初の教師データから再び学習を行う．

このようなサイクルの中で，ネットワークの各素子はより多くの報酬を得るための構造を探索し，ネットワーク全体として受け取ることのできる報酬量の増大が見込まれる．その結果として，ネットワークの入出力を教師データに沿ったものへと改変することが期待できる．

また，以上のサイクルによって，SONEは各学習データをオンラインに学習することが期待できる．ここで，オンライン学習についての説明を行うことにする．バッチ（オフライン）学習とオンライン学習の違いは，バッチ学習の場合には，教師データの全てを用いてサイクル（ここでは，図3.1の内側のサイクル）を実行するのに対し，オンライン学習は，教師データの一部によって順次サイクルを実行できることにある．また，バッチ学習では，完全な教師データを必要とするために，ロボットが長時間学習を続けると，教師データのデータベース容量が不足する，学習に時間がかかる等の問題が発生する．これに対し，オンライン学習では完全な教師データを必要としないため，以上の問題に対し頑健である．特に今回のSONEによるオンライン学習の仕様では，1回の入出力データを単位として学習するため，教師データのデータベース容量は全くといって良いほど問題にはならない．

3.2　軌道学習に関する試験

本節では，軌道学習に関する問題をSONEに学習させるPC上のシミュレーション実験を通じて，ノイズを含んだ問題に関する学習，追加学習，時系列学習に関するSONEの基本特性を明らかとする．

本実験で対象とする軌道は，2次元平面上の円軌道 (C-track) と8の字軌道 (I-track),

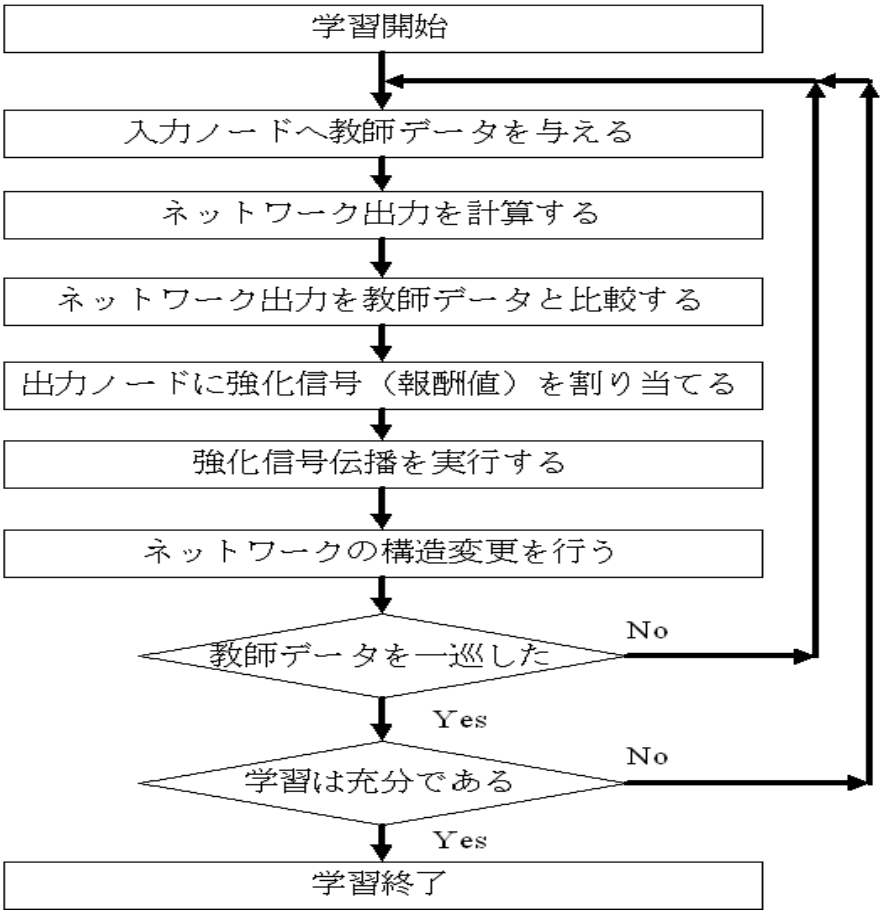


図 3.1 Supervised learning flow

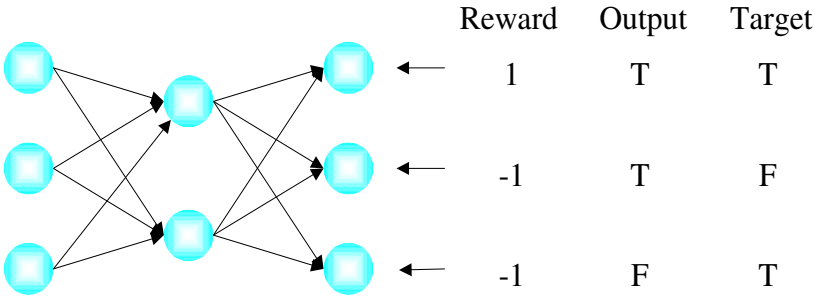


図 3.2 Reinforcement signal setting for supervised learning

そしてC-trackとI-trackを交互に周回するC& I-trackの3種類である．そして本実験では，仮想的なエージェントがこれらの軌道を周回するのに必要な制御ネットワークを，SONEによって獲得させることで，エージェントの軌道学習を実現する．

このような学習をニューラルネットワーク等の学習器で行う場合，通常はオフライン学習を用いるが，本実験ではオンライン学習により実現する．SONEを用いたオンライン軌道学習について説明する．

まず，図3.1の入力ノードへの教師データの割り当てでは，軌道上の点に関するX-Y座標にA/D変換を施し，それぞれ16bitのデジタルデータとしてSONEに入力する．また，図3.1のネットワーク出力によって得られる値にD/A変換を施すことによって，軌道上のエージェントが次に移動するべき軌道上の点と対応させる．出力ノードへの強化信号の割り当てでは，正解となる点のX-Y座標とネットワークの出力したX-Y座標を比較して割り当てを行う．

このようにして，C-track，I-track，C&I-trackをオンラインに学習させる．

3.2.1　ノイズを含んだ問題に関する試験

まず，SONEのノイズに対する基本特性を検証するために，先の学習サイクルにノイズを付与して学習を行わせた．

SONEに入力する軌道データのX-Y座標に関して，軌道全体のスケールを1としたときに最大0.05までの半径でアナログ的にノイズを付与する．こうすることで，本実験でのSONEへの入力は，軌道上の同一点においても各周回毎に異なるものとなり，単純なデータベース等では学習ができない問題となる．

C-track，I-track，そしてC&I-trackに対しノイズを付与せず学習した結果を表3.1に示し，ノイズを付与して学習した結果を表3.2に示す．表3.1，表3.2には各軌道を10回学習した際の平均値として，学習収束時までの平均ステップ数，学習終了時における出力ビットの平均誤差，学習収束時の実数値換算での平均誤差（軌道全体のスケールを1とする），学習収束時のネットワークに含まれるノード数の平均，そして学習収束時のネットワークに含まれるリンク数の平均が示してある．

まず，表 3.1，表 3.2 の平均誤差（ビット/実数）より，SONE の学習効果が確かであることがわかる．特にノイズ有りにおいて C-track，I-track では，全ての実験において誤差が 0 に収束しており，軌道が完全に学習できている．

次に，この結果はノイズの無い環境での実験結果（表 3.1）よりも良い数値である．このことから，SONE ではノイズの付与によって収束先の誤差が小さくなることが明らかとなった．一般に，ニューラルネットワークの学習においても，ネットワークをノイズ環境下におく事によって，収束先の精度が改善されることがわかっている．これは，局所解の脱出可能性が向上するためであり，SONE においても同じ理由によって誤差が改善するのだと考えている．

また，ノイズ有りでは，ノイズ無しに較べて収束までに多くのステップ数を要し，収束点でのノード数，リンク数も多い．これは，上記の仮説とも矛盾しておらず，局所解の脱出可能性が高まることにより，さらに深く学習が行われ，多くのノード，多くのリンクを生成したのだと考えられる．この効果については後の実験においてさらに検証する．

そして，与える問題ごとの違いを見ると，ノイズ有り，ノイズ無しともに C-track における収束時のノード数，リンク数が最も少なく，I-track，C&I-track となるに従って増大している．一般的に，I-track は時系列問題を内包しており（時系列問題に関しては後の試験で扱う），C-track よりも高度な問題であるとされている．また，C-track と I-track を交互に周回する C&I-track はこれらの軌道のうち最も難しい問題であると考えられる．比較的単純な問題である C-track では収束時のノード数，リンク数が最も少なく，比較的高度な問題である C&I-track では収束時のノード数，リンク数が最も多いことから，SONE は問題の難度に応じてネットワーク構造を複雑化することで，様々な問題に対処していると考えられる．

さらに，ノイズの有る環境での誤差が 0 にまで収束したことより，SONE は汎化能力を有する可能性が高い．汎化能力については Two-spiral 問題によってさらに検証する．

表 3.1 Learning result (whitout noise)

学習データ	平均ステップ数	平均誤差 (ビット)	平均誤差 (実数)	収束点のノード数	収束点のリンク数
C-track	88.3	0.3	6.25×10^{-7}	70.8	244.0
I-track	1064.9	0.3	4.93×10^{-5}	82.2	306.3
C& I-track	923.1	1.0	2.78×10^{-4}	149.6	536.7

表 3.2 Learing result (with noise)

学習データ	平均ステップ数	平均誤差 (ビット)	平均誤差 (実数)	収束点のノード数	収束点のリンク数
C-track	196.3	0	0	91.5	318.2
I-track	1068.7	0	0	153.9	537.5
C& I-track	2779.8	1.0	2.08×10^{-6}	448.1	1725.3

3.2.2 時系列学習に関する試験

SONE の時系列学習に関する基本特性に関する試験によって，SONE が時系列問題を学習する際の特性を明らかとする．本節では，ネットワーク内に発生するフィードバックループに関する考察と，SONE にメモリ機能を有する素子を追加した際の学習に関する試験を行う．

時系列学習はロボットが受け取るセンサ入力の時間的な遷移に従って現在のロボットのモータ出力を決定するための学習である．通常，静的な入出力の関係を学習する場合には，ロボットの現在のセンサ状態 $I(t)$ のみを用いてロボットのモータ出力 $O(t)$ を決定する規則を学習するのに対し，時系列学習では，$I(t)$,$I(t-1)$,$I(t-2)$…といった過去の状態の履歴を用いて $O(t)$ を出力する規則を学習する．前者の静的な学習では，隠れ状態を含むデータを学習できないのに対し，時系列学習では状態の履歴情報から，隠れ状態を考慮したデータも学習できる場合がある．先の I-track，C&I-track に関する問題はこの隠れ状態を含んでおり，時系列学習無しには学習できない問題である．

ここでの隠れ状態は，適切な $O(t)$ を決定するために $I(t)$ 以外に必要となるロボットの内部状態のことである．例えば 8 の字軌道 I-track における軌道中央の重複点上では，次に移動するべき適切な点 $(O(t))$ は重複点の座標 $(I(t))$ からは一意に決定できない．ここでは，右上から左下に抜けるシーケンス，左上から右下に抜けるシーケンスのいずれを実行中であるかに関する情報が隠れ状態となっている．この隠れ状態は $I(t-1)$ や

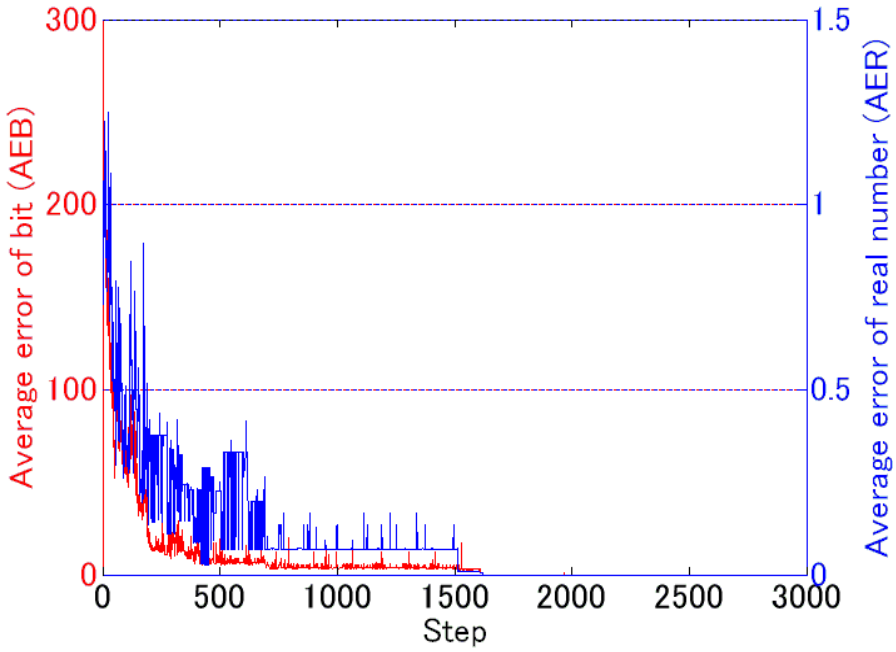


図 3.3 Experiment without noise

$I(t-2)$ などの過去の履歴を通じて形成・記憶しておくことで，$O(t)$ を算出する際に活用することができる．

3.2.3 フィードバックループを用いた学習に関する考察

まず，SONE による時系列学習に対する考察を行う．SONE が I-track，C&I-track を学習するためには，$I(t-1)$ や $I(t-2)$ などの過去の履歴を利用して $O(t)$ を出力することが必要となるため，ネットワークにはメモリ機能が必要となる．

一方で，先の SONE の仕様でメモリ機能を獲得できるとすれば，ネットワーク内にフィードバックループが形成され，学習に活用される以外にはありえない．なぜならば，メモリ機能を持たない素子の組み合わせによってフィードフォワードな回路を作成した場合，得られる $O(t)$ は必ず過去の入力 $I(t-1)$ や $I(t-2)$ とは独立したものとなるからである．

よって，先の試験において I-track や C&I-track が学習できたという結果から，SONE によってフィードバックループが適切に獲得され，時系列学習が行えている可能性は

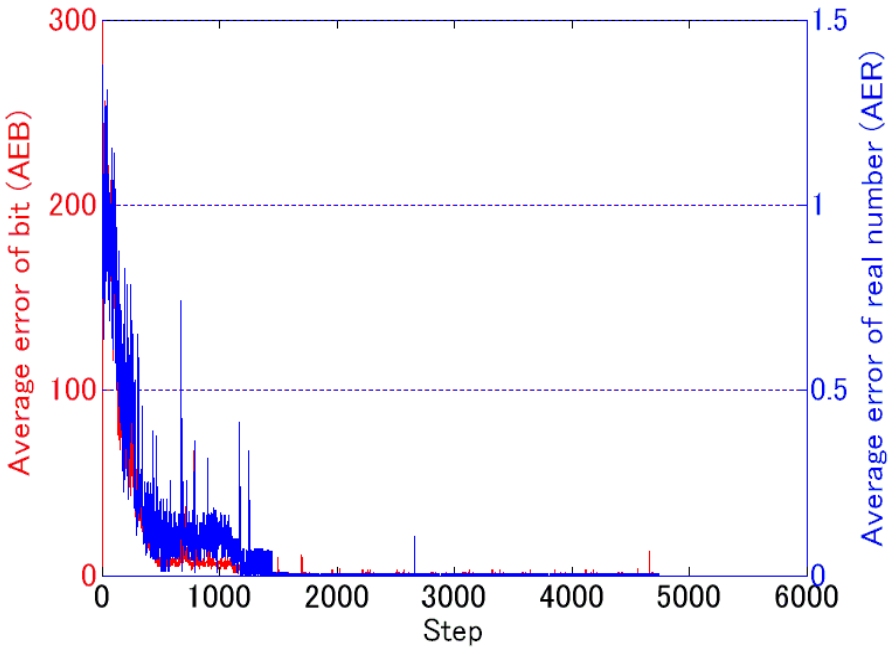


図 3.4　Experiment with noise

極めて高いといえる．

3.2.4　メモリ機能を有する素子による学習

このように，SONE ではフィードバックループを用いて時系列学習を行うことができたと考えられる．しかしながら，先の実験による方法では十分に学習できない軌道も発見できたため，時系列学習をさらに強化するための別の工夫が必要であると考えた．

本節では，フィードバックループによって十分に学習できなかったと考えられる軌道の例を示し，その軌道郡に対して効果的な SONE の構成法を提案する．具体的には，SONE によって学習できなかった軌道は図 3.8 に示すような，長期の隠れ状態を伴った軌道である．この軌道は I-track のように重複点を持つ軌道であるが，中心の重複部分が長いシーケンスとなっている軌道である．以下では，このような軌道を学習できる SONE の構成法を示し，新しく作成した SONE とリカレントニューラルネットワーク Recurrent Neural Network (RNN) の両方に対して試験を行う．

3.2.4.1 フリップフロップ素子の導入

従来SONEではフィードバックループの形成により，単純な時系列問題を扱えていると考えられる．しかしながら，この枠組みでは図3.8の軌道を学習する際に必要となる，長期的な隠れ状態を記憶・保持することは困難であった．そこで本節では，後者の方法を用い，SONEによるネットワーク内にメモリ機能を備えた素子を自己組織的に獲得させることとした．今回使用した素子は表3.3の真偽表に従ってその出力を決定するフリップフロップ素子（FFノード）である．本節ではSONEによりネットワーク内の各素子が評価できることを利用し，FFノードを自己組織的に獲得させた．FFノードの生成は，他のノードと同様に，ネットワークの各リンクにテスト用FFノードを備えることで，テスト用FFノードの評価値が閾値を上回った際に昇格・実用化することとした．またFFノードの解体には，FFノードの出力側リンクが0本となった際に解体することとした．

表 3.3 Truth table of FF-node

入力 1($I_1(t)$)	入力 2($I_2(t)$)	出力 ($O(t)$)
T	T	T
T	F	$O(t-1)$
F	T	$O(t-1)$
F	F	F

3.2.4.2 試験

今回の試験の方法も軌道学習に関する試験に準じて行い，図3.8に示すような軌道に関して行う．図3.8に示す軌道は，中央の重複軌道へ右上より進入した場合左下へ抜け，中央の重複軌道へ左上より進入した場合右下へ抜ける軌道である．この軌道は8の字軌道（I-track）と較べ，中央の重複点が多いことが特徴となっており，右上，左上のいずれから進入したかという情報を記憶，保持しておかなければならない期間が比較的長い問題である．この問題の特徴としては，記憶保持の期間が中央重複点の長さによって変更できる．したがって，中央重複点の長さを様々に変えて試験を行うことで，

学習制御器の獲得するメモリー機能が，どれだけ長期にわたった記憶を扱えているのかを検証することができる．

今回，SONEに関してはFFノードを導入したSONEを用いた．また，RNNに関しては，入力層，出力層のノード数を2とし，中間層のノード数を5から20，コンテキスト層のノード数を1から10，さらには学習率を0.001から0.2の間で様々に変えて実験を行った．具体的なRNNのパラメータを図3.4に示す．このRNNによる学習には，学習則としてバックプロパゲーション・スルータイム（BPTT）[50]を利用し，80万ステップを上限としてオフラインで学習を行った．

試験の結果，FFノード導入前のSONEには学習できなかった，図3.8に示すような軌道が学習できるようになった．また，SONE，RNNともに，学習に失敗したケースでは図3.9に示すような局所解軌道への収束が多数確認された．表3.5にSONEによる結果と，RNNによる代表的な結果を示す．また，図3.10に重複軌道の長さを変えたときのSONEとRNNの正答率を示す．

まず，局所解（図3.9）に関する考察を述べる．従来のSONEやRNNが収束した局所解軌道は重複軌道の出口の点と，出口の1ステップ先に到達すべき点が重なってしまっている．これは重複軌道の出口において記憶の活用が行われておらず，左右どちらの軌道へ移動するかの判断ができていないためである．その後左右の軌道への復帰が見られるが，これは教師データからの入力を得てからの復帰であり，学習による隠れ状態の利用は行えていない．この局所解軌道はここで目的としている隠れ状態の学習に関しては失敗していると考えられるので，学習の正答率を算出する際には「失敗」に含めている．

表3.5は，図3.10においてSONEとRNNに顕著な差が見られる，9点連続の隠れ状態を持つ軌道に関しての試験データである．ここでは，中間層10，コンテキスト層5のRNNが比較的良好な収束を見せたため，それを用いている．FFノードを導入したSONEでは，平均誤差が最も低くなり，隠れ状態に関しても学習ができている（隠れ状態に関する学習成功率は，学習後の軌道を目視で確認して算出している）．FFノードを導入していないSONEとRNNはほぼ同様の平均誤差に収まっているが，いずれもほとんどが図3.9に示すような局所解軌道となっており，隠れ状態の学習は成功して

いない．図 3.10 の結果とも併せて，FF ノードを導入した SONE では，従来の RNN や SONE では学習が不可能であった長期の隠れ状態を伴った軌道が学習できたといえる．

FF ノードの効果についての考察を行う．従来の SONE や RNN の場合，記憶の保持はフィードバック結合によって行われるため，減衰や外乱の影響が生じ易く長期間の記憶保持は困難であると考えられる．一方で FF ノードを導入した場合，FF ノード自身がスイッチのように動作し，この動作には減衰を伴わないため，比較的に長期間の記憶保持が可能であると考えられる．よって，適切に FF ノードが自己組織化された場合，任意時間にわたってのメモリ構造が獲得できるため，長期の隠れ状態を持つシーケンスにも対応できると考えられる．しかしながら，この軌道学習の例では，学習できる範囲が最大でも 20 点連続の隠れ状態にとどまっており，さらなる工夫が必要であると考えられる．

表 3.4 Parameters of recurrent neural network

パラメータ	試験における設定
学習率	0.001〜0.2
入力層ノード数	2
中間層ノード数	5〜20
出力層ノード数	2
コンテキスト層ノード数	1〜10

表 3.5 Result of learning (9 continuous hidden states)

学習器	ステップ数	平均誤差	平均中間ノード数	平均 FF ノード数	隠れ状態の学習確率 [%]
SONE（FF）	2751.0	4.25×10^{-2}	190.2	38.6	90
SONE	8947.6	2.41×10^{-1}	208.8	-	0
RNN（学習率 0.01）	−*	2.49×10^{-1}	10	-	0
RNN（学習率 0.1）	−*	2.15×10^{-1}	10	-	0
RNN（学習率 0.05）	−*	2.27×10^{-1}	10	-	0

*80 万ステップ学習時

3.2.5 追加学習に関する試験

オンラインに学習を行う学習システムでは，新しく学習するべき情報と，既学習の情報が一つのシステム内でどのように共存・保持できるかが問題となる．そこで，新

しい情報が追加された際の頑健性を検証するために，追加学習に関する特性の試験を行う．

先の試験では各軌道を一度学習するのみであったが，この試験ではある軌道を学習した後に別の軌道の再学習を行う．そして，再学習による既学習情報の忘却の度合い，さらには再学習を繰り返し行った場合の特徴などを調べる．

今回の試験の方法も軌道学習に関する試験に準じて行う．この試験では，C-track を学習したネットワークにさらに I-track を学習させ，I-track が学習し終わると再度 C-track を学習させるという操作を繰り返し行うこととする．この試験では FF ノードを導入しない状態の SONE を用いた．また，この試験においても RNN との比較を行った．

図 3.11(a) に SONE における学習誤差の履歴を示す．図 3.11(a) においてオーバーシュートがおきている部分は，C-track と I-track の切り替えのステップに相当する．C-track を I-track へ切り替え，再び C-track を学習した際には最初に C-track を学習した場合よりもピークの低い位置から学習が再開できている．

これによって，以前に C-track を学習した記憶を活用して，再び同じ問題が与えられた場合には以前の記憶を利用して対処していること，さらにはその記憶が他軌道 (I-track) を学習している間にも完全には消失されないことが確認できる．

今回の試験では比較のためリカレントニューラルネットワーク Recurrent Neural Network (RNN) に対しても試験を行った．RNN は予備実験によって比較的良好であった，2-20-2 のトポロジーを持つ三層型を用い，コンテキスト層は 5，学習は BPTT により学習率 0.01 でオフラインに行った．

図 3.11(b) に RNN を用いた際の学習誤差の履歴を示す．図 3.11(b) より，RNN による学習は SONE による学習よりも 10,000 倍程度のステップ数を要することがわかる．また，RNN による学習では誤差のピークは収束せずに跳ね上がってしまっている．

RNN の追加学習では，学習方法の性質上追加学習が困難である．例えば，Catastrophic Forgetting という現象によって，追加学習時に既学習の情報に致命的な忘却が生じることが指摘されており [34]，RNN によってこのような学習を行うことは非常に困難だと考えられる．この Catastrophic Forgetting は，コンソリデーションラーニングという学習法 [35, 36] 等によってある程度克服することができるとされているが，この手法を

用いた場合には時系列的な，または連続的な学習を行うことは難しくなる．

コンソリデーションラーニングでは二つのニューラルネットワークを用いて学習を行う．本試験を例に説明をすると，まず一つのニューラルネットワーク（ネットワークA）にC-trackを学習させる．次に，I-trackに切り替え，学習を行う際には，もう一方のニューラルネットワーク（ネットワークB）を用いる．ネットワークBが学習を行う際には，ネットワークAの入力にランダムな入力を加える等して得られる，ネットワークAの入出力情報と，学習するべきI-trackの情報を同時にネットワークBに提示することで，既学習の情報に対する損失を防ぐことができる．

この方法を時系列的な枠組みに拡張する場合には，既学習のネットワークAに，どのようにして情報を吐き出させるかが問題となる．一般に，時系列問題を扱う際のネットワークの出力は，入力の時間遷移によって決定されるため，ネットワークAにランダムな入力を与えるだけでは適切なシーケンスデータが得られる保証がない．また，静的（時系列を含まない）な問題を扱う場合と較べ，ネットワークAの入力にシーケンスを導入しなければならないため，そのシーケンスの組み合わせが発散するおそれもある．

さらに，この方法ではオンライン学習への拡張も困難であるといえる．SONEがオンライン学習を行う場合の例では，C-track，I-trackの切り替えはシステム内部からは意識されていない．つまり，C-track，I-trackの軌道情報自体は外部の系によって切り替えられてはいるが，SONEの学習はその切り替え情報を利用してはいない．

それに対し，コンソリデーションラーニングでは，C-trackとI-trackの切り替え情報によって二つのネットワークを切り替えなければならない．そしてこのシステムでは，この切り替え情報によって，C-trackの学習，I-trackの学習を二つに分けて扱い，それぞれをオフライン学習するために，C-trackとI-trackの学習は連続的には行えない．

本試験では，SONEの追加学習に関する特性試験を行うとともに，SONEとRNNの比較を行った．SONEの追加学習は比較的良好な特性を示し，RNNでは困難である学習形式を実現できたといえる．

3.3　二重螺旋問題 Two-spiral Problem に関する試験

本節では，SONE の汎化能力を確かめるために，二重螺旋問題による試験を導入した．この問題は，主にニューラルネットワークの分野で用いられているベンチマークであり，この問題を解くには，非線形データへの対応や汎化能力を必要とする．

二重螺旋問題は，ニューラルネットワークのベンチマークとして Alexis Wieland によって初めて導入されており，この問題を解くには極端に非線形な入力空間の分割を行う必要がある．よって，非線形性の強いデータに関する学習能力や，汎化能力を問う際の基準として使用できる．そして Baum と Lang は，通常の三層型ニューラルネットワークに誤差逆伝播法を適用したのみではこの問題を解くことができないことを指摘しており [51]，それは Susan による解析によっても確かめられている [52]．Lang と Witbrock はショートカットコネクションと呼ばれる特別な結合を持ち，2-5-5-5-1 の層構造をしたネットワークを導入することで解決できることを示している [53]．さらに Fahlman と Lebiere は，カスケードコリレーションという特殊な学習システムを用いることでこの問題を解くことができることを示しており [54]，Weenink もまた category adaptive resonance theory neural network（Categoty ART）を用いることでこの問題を解いている [55, 56]．

このような二重螺旋問題を試験として扱うことで，SONE の汎化能力の有無を検証できると考え，検証を行った．

3.3.1　試験

図 3.12(a) に問題となる二重螺旋のデータを示す．二重螺旋問題は，ふたつの螺旋上の点を分類，区別する問題であり，ネットワークは入力として 2 次元平面上に描かれた螺旋上にある点の X-Y 座標を受け取り，その点がどちらの螺旋に含まれるかを判断する 2 値出力を返すように学習を行う．

この問題では，ネットワークへのデータ入力の方法に Normalization Cording，Binary Cording，Weighted Binary Cording，Temperature Cording 等が提案されており [57]，ここでは Temperature Cording を用いて試験を行った．

3.3.2 試験結果

図3.12(b) にSONEによる二重螺旋問題の学習結果を示す．また，図3.13(b) にFahlmanのシステムによる結果を示す．いずれの結果においても，二重螺旋は領域として二つに分離されており，SONEでもニューラルネットワークと同様に，学習データの汎化が行われていることがわかる．これによって，学習データに無い渦上の点をネットワークへ入力した場合にも，高い確率で良好な出力が得られることが期待できる．

また，線形分離が非常に難しいデータに対して良好な学習ができた結果から，SONEによる学習精度の良さが再確認できた．

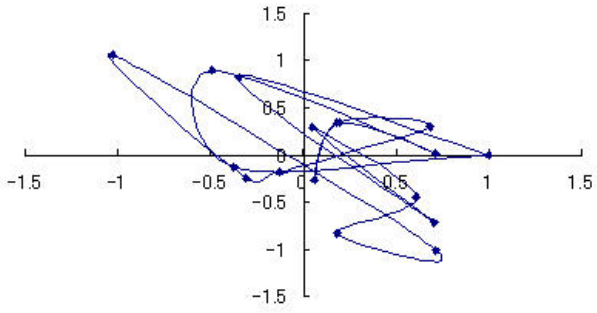
3.4 全体の考察とまとめ

本章ではSONEに対し，軌道学習に関する試験，二重螺旋問題に関する試験を行った．軌道学習に関する試験では，SONEには入力信号に対する耐ノイズ性が有ること，時系列学習ができること，SONEにフリップフロップ素子を導入することで長期の隠れ状態へ対応できること，RNNと比較して良好な追加学習能力を有することが明らかとなった．さらに，二重螺旋問題に関する試験より，SONEによって汎化を伴った学習が実現できることも明らかとなった．

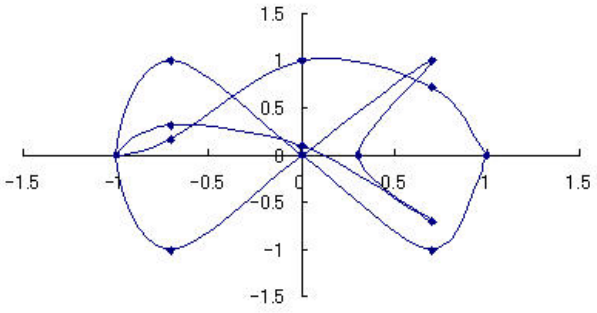
自律型ロボットの学習制御器に必要となる項目との対比を行う．汎化・抽象化に関してはニューラルネットワークの汎化能力を試す二重螺旋問題への有効性から汎化能力が確保できるといえる．また，抽象化に関してはフリップフロップ素子を用いた実験において特定の軌道領域を抽象化した素子が検出できたことから可能だといえる．柔軟性に関しては，全ての実験を通じて同一の学習パラメータが適用できたことから，ある程度の柔軟性が確認できたと考えられる．オンライン性に関しては，全ての実験を通じてオンラインに学習しているため，確認できている．漸次性に関しては，追加学習の実験を通じてニューラルネットワークよりも高い漸次性が確認できた．

本論文では，比較対象のニューラルネットワークとして，BPTTによるRNNを用いたが，ニューラルネットワークを用いた時系列学習としてはReal Time Recurrent Learning（RTRL）も考えられる [58–60]．RTRLはオンライン型の学習アルゴリズム

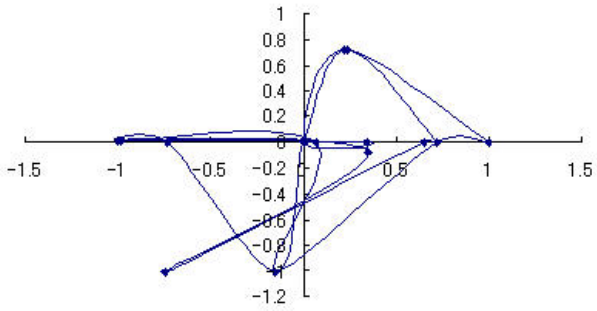
であり，BPTT による学習よりも SONE に近い．しかしながら，多くの RTRL は BPTT で用いている最急降下法をオンライン型にアレンジしているため，学習精度では BPTT が上回る場合が多いと考え，本論文では BPTT との比較を行った．今後は，BPTT に基礎を置かない特殊な RTRL や他の学習手法との比較を行っていきたい [61–64]．また，FF ノードによって区切られたシーケンスがどのような基準で切り出されているのかをさらに解析する必要がある．これに関しては谷らによる RNNPB や，GA によるネットワーク自己組織化による方法等とも特徴を比較してききたい [65–67]．そして，汎化能力の評価法としては他にも，横井らによるリーマン幾何学を用いた方法 [68] などもあり，適用を検討していきたい．



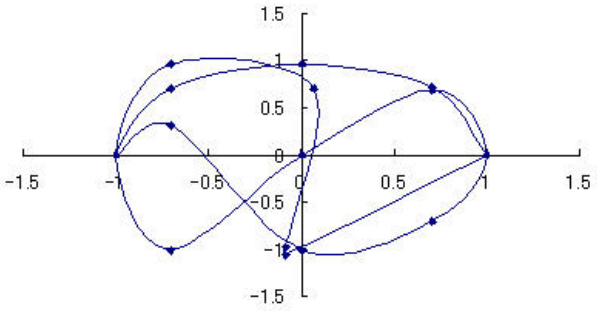
(1) 0 step



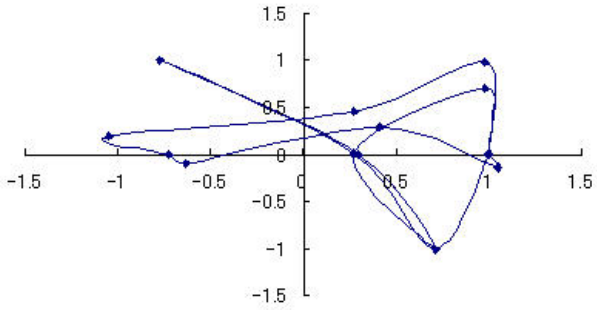
(5) 400 step



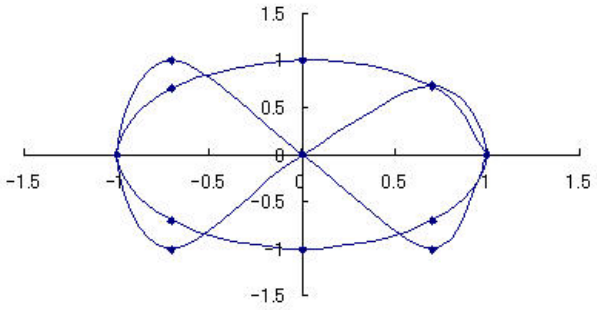
(2) 100 step



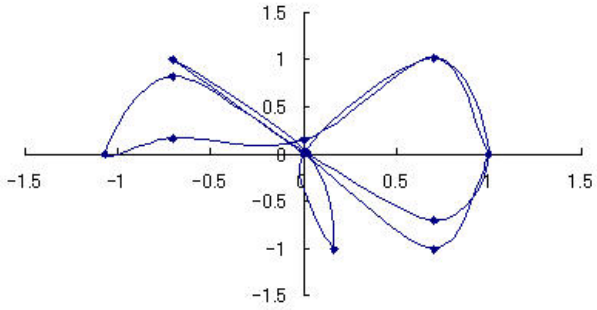
(6) 500 step



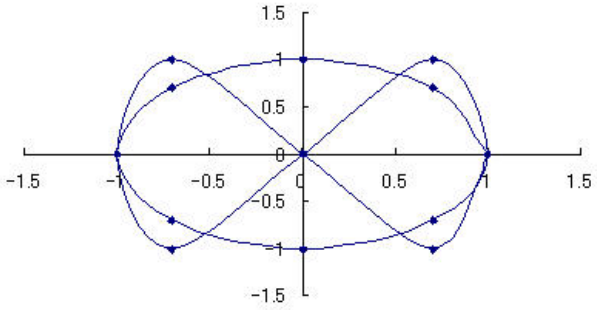
(3) 200 step



(7) 600 step



(4) 300 step



(8) 772 step

図 3.5 History of the track

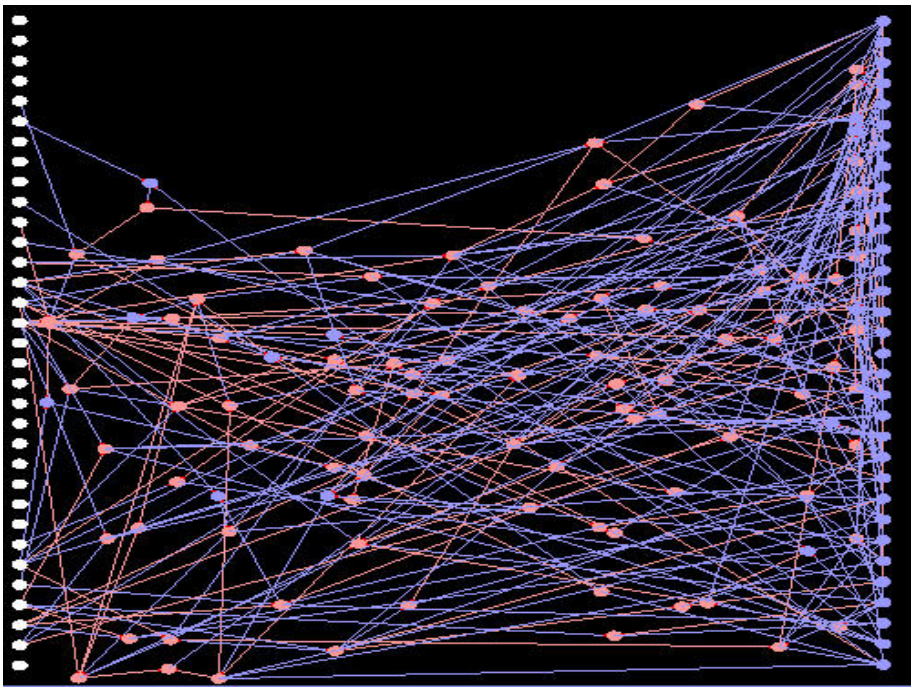


図 3.6　Generated network structure(C-track)

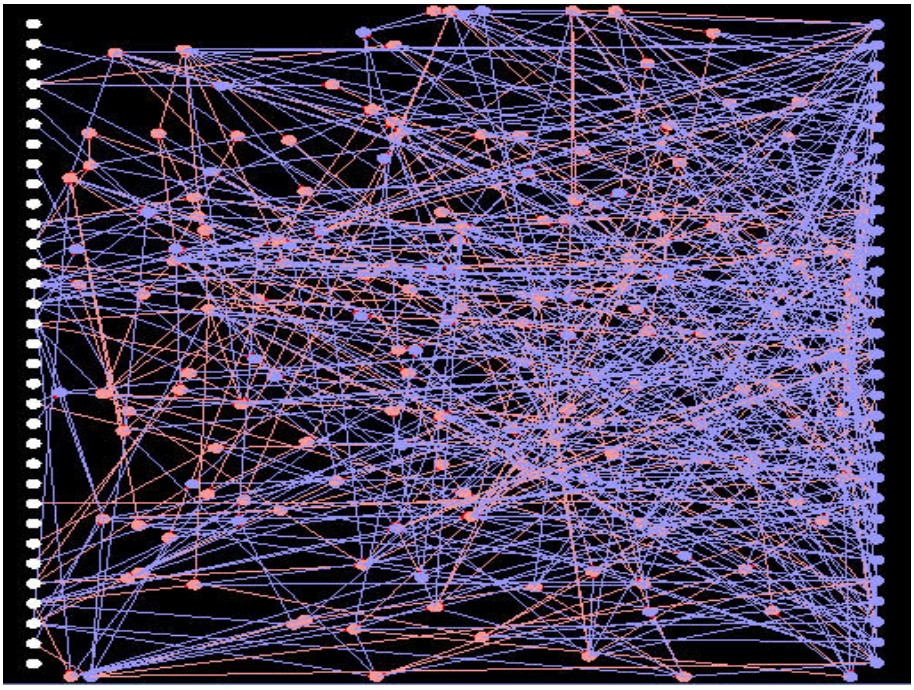


図 3.7　Generated network structure(I-track)

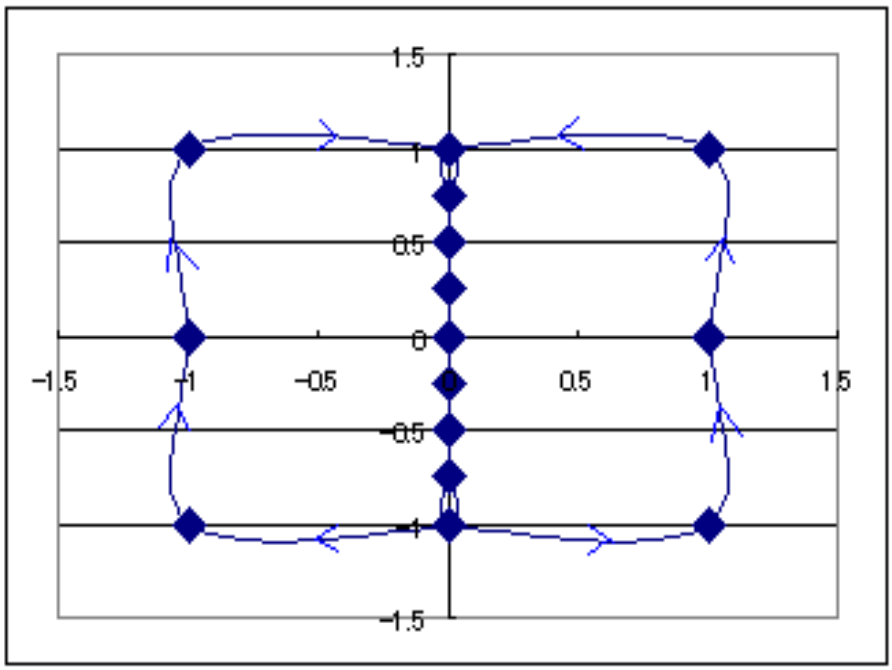


図 3.8 A track which has long term hidden state

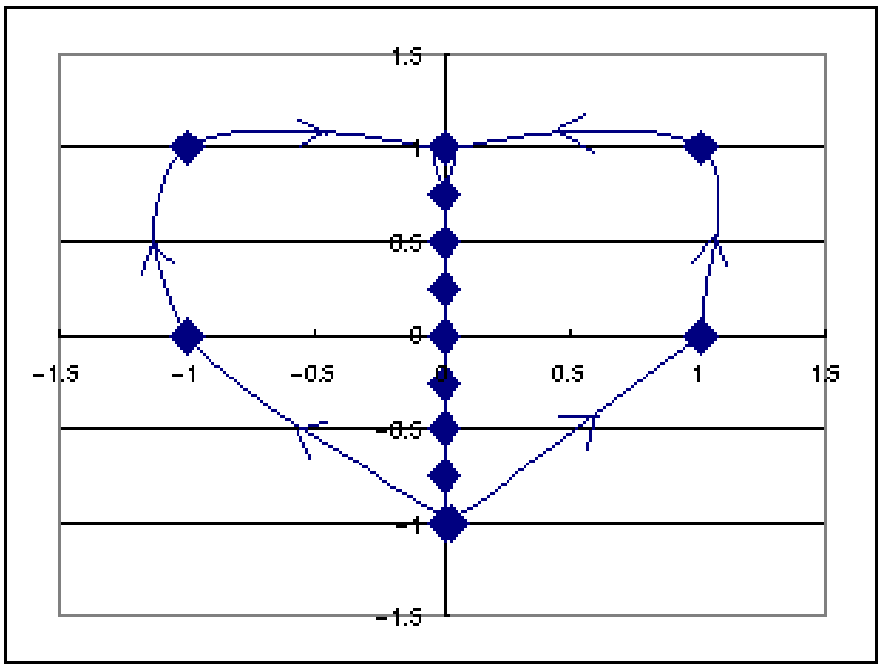


図 3.9 A local solution

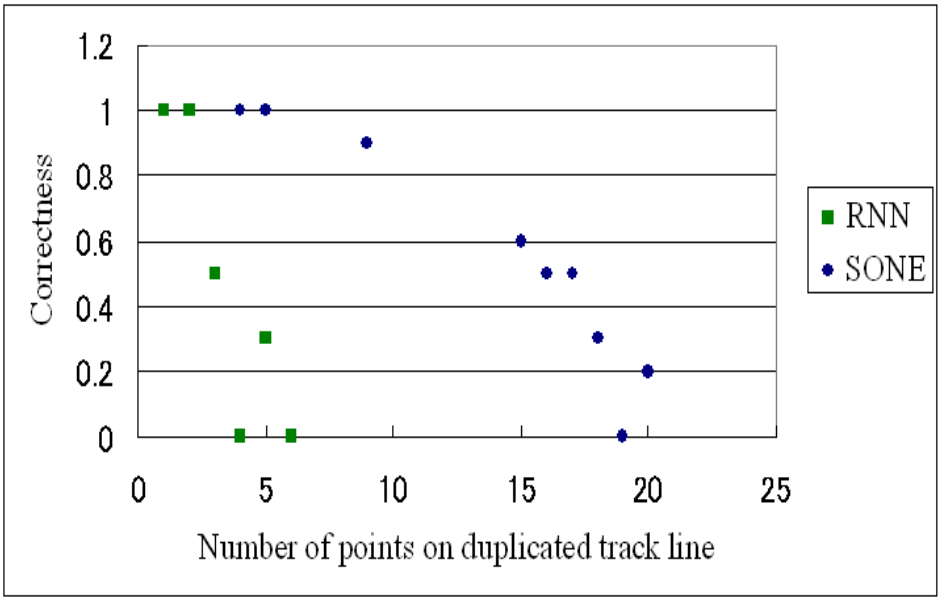
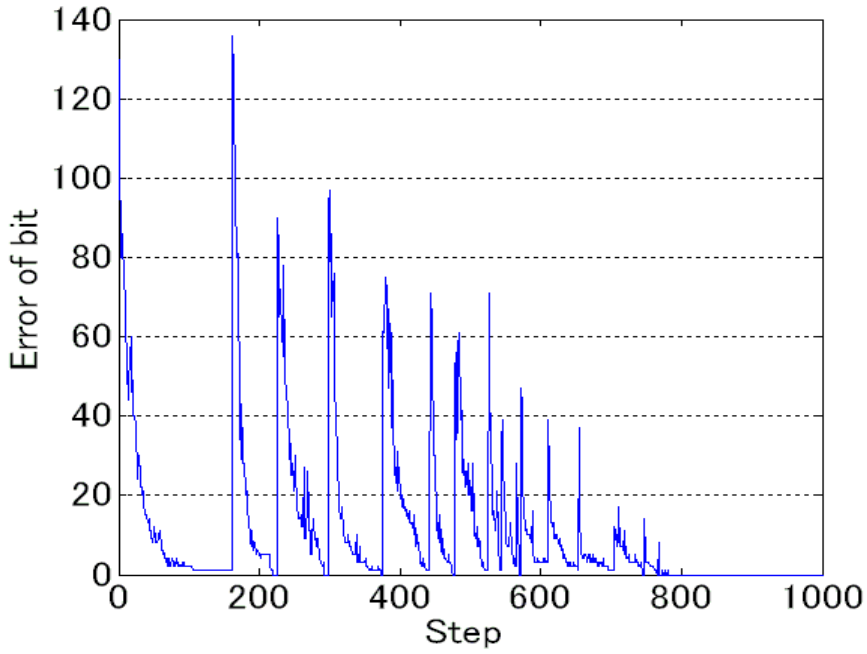
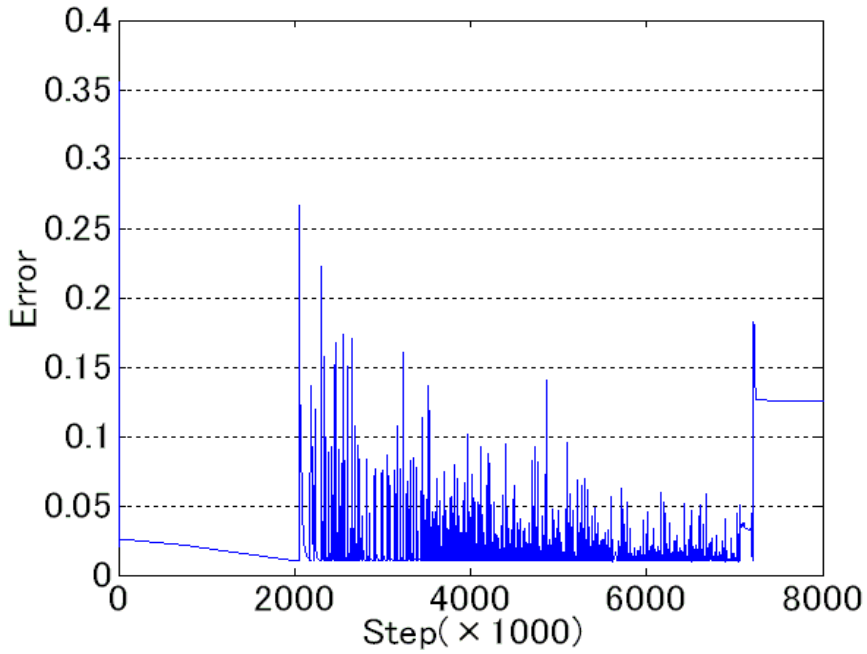


図 3.10　Comparison of acculacy

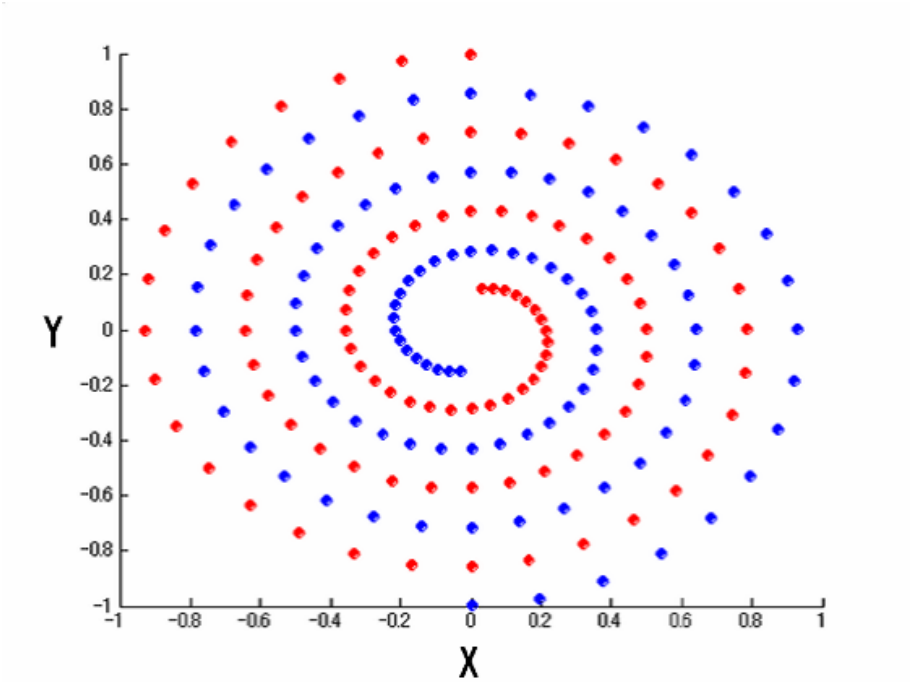


(a) SONE

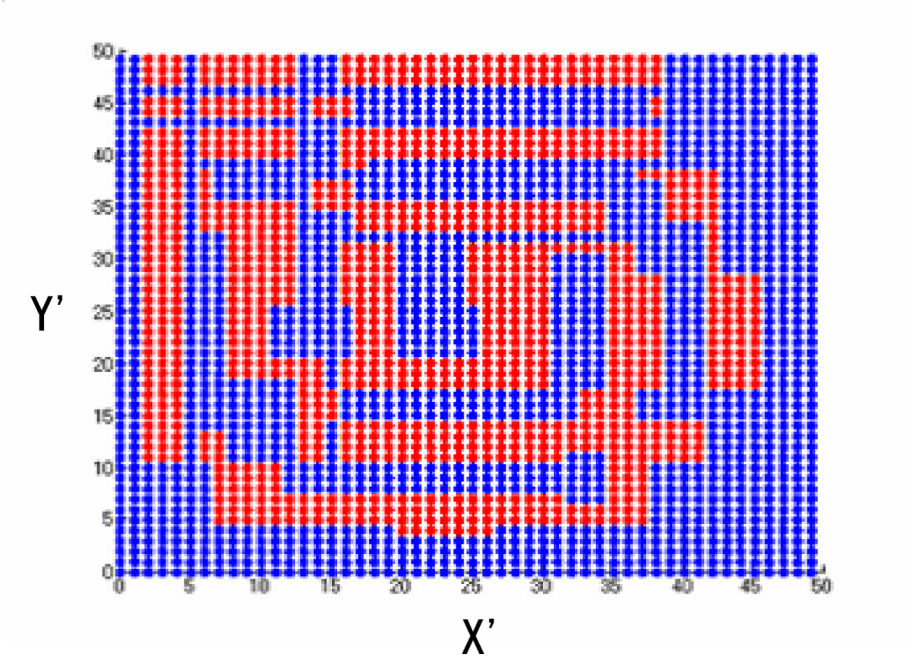


(b) RNN

図 3.11 incremental learning

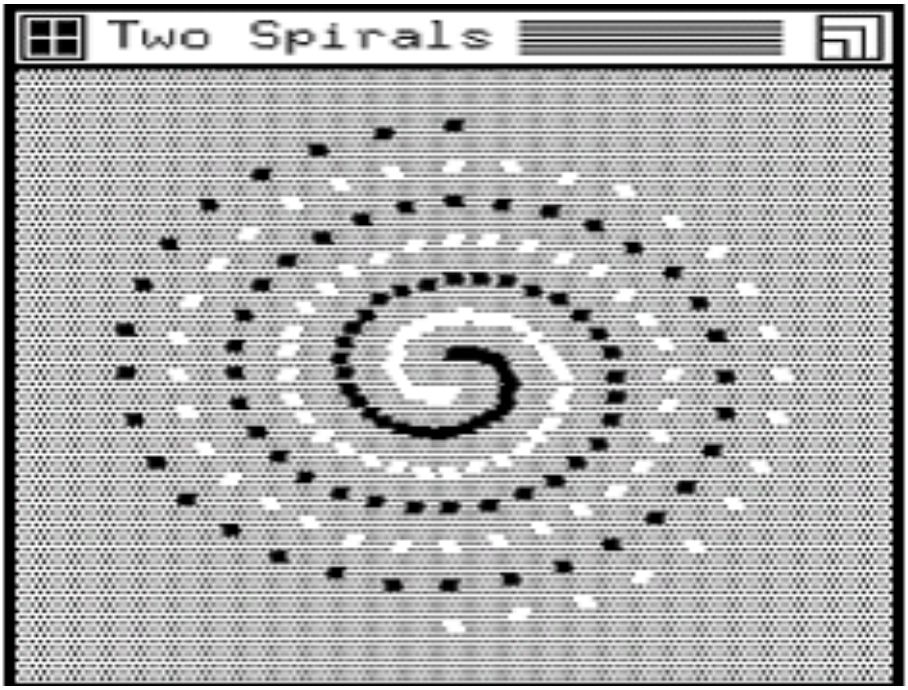


(a) Learning data

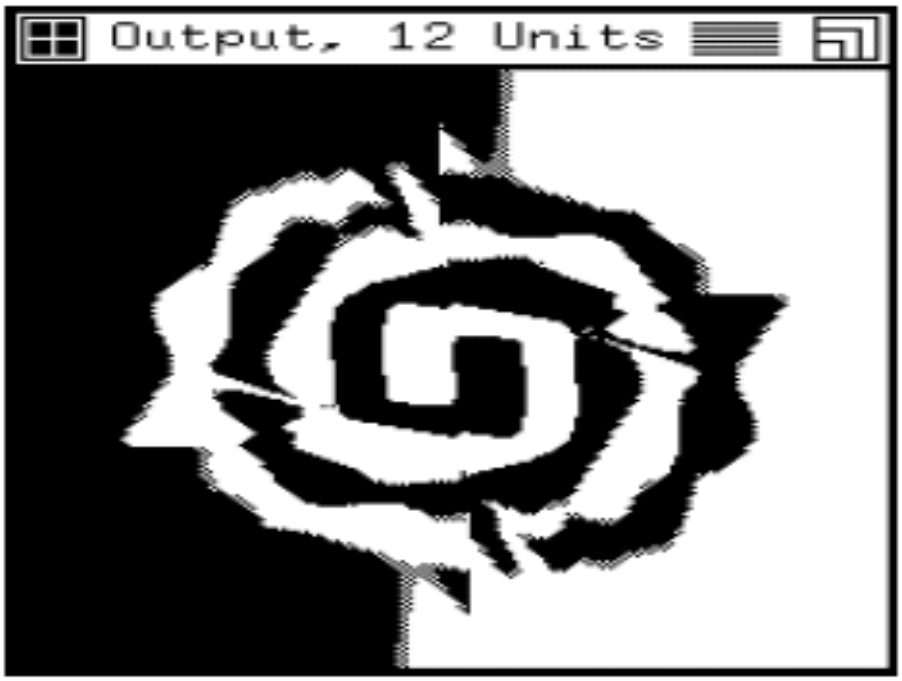


(b) Learning result

図 3.12　Two spiral problem (SONE)



(a) Learning data



(b) Learing result

図 3.13 Two spiral problem (Fahlman's neural network)

第4章　移動ロボットにおける衝突回避学習実験

本章では自律型ロボットの学習制御器へ必要な五項目のうち，主に行動創発，柔軟性，オンライン性に関する検証を行うために，移動ロボットにおける衝突回避実験を行う．具体的には，シミュレーション上の移動ロボットにおける衝突回避を学習するタスクを扱い，強化学習によってロボットの制御ルールに改善が見られること，設計者による状態空間の分割や，ネットワーク構造の決定を必要としないこと，オンライン・リアルタイムに学習ができること等を確かめる．

4.1　実験環境

実験はコンピュータ物理シミュレーションによって行う．具体的には，ロボットシミュレータ Webots5 を用いて移動ロボット Khepera2 のモデル化と衝突回避学習実験を行う．

Webots5 は車輪移動型・歩行型・飛行船などの各種移動ロボットを容易にモデル化，シミュレーションできる他，各種エンコーダやセンサ類に対し，範囲，ノイズ，応答，視界などの詳細な設定できる．また，シミュレーション中に仮想時間が設定できるため，コンピュータの性能によらない実験時間の測定ができる．

Khepera2 は図 4.2 に示すような移動ロボットであり，入力として 8 つの赤外線センサ，出力として車輪を回転させるための 2 つのモータを備えている（詳細：表 4.2)．シミュレーション上での赤外線センサの有効範囲は図 4.2(b) において，ロボットから伸びる放射状の赤線で示してある．また，今回の実験ではロボットの軌跡を観察するために，Khepera2 の胴体下部にペンを設定し，Khepera2 が移動するとその軌跡が自動的に描かれるように設定した．

4.2　学習制御器の設定

以上で示した Khepera2 へ SONE を導入する．本節ではその設定について説明する．

4.2.1　入出力の設定

SONE は初期状態として，図 4.1 のように入出力間に何も結合を持たない状態で実装する．この際の SONE の入出力ノードの構成は，入力ノード 256 個 (16×8)，出力ノード 32 個 (16×2) とした．この SONE の入力として，Khepera2 の持つ赤外線センサからの入力を 16bit のバイナリデータに置き換えて用いる．また出力は，SONE から得られる 16bit の情報を実数値に置き換えて各モータに速度信号として伝える．ただしここでは，モータ出力を 16 段階としているため，実際には実数値より選択されたいずれかの段階がロボットのモータ出力として適用される．

4.2.2　強化信号の設定（Actor-Critic 法の導入）

SONE に与える強化信号を設定するために，Actor-Critic 法を基にしてロボットの状態価値を定めた．Actor-Critic 法は学習制御器を Actor 部と Critic 部に分け，Actor 部ではロボットの行動選択を行い，Critic 部では Actor 部の行動選択を評価するためのロボットの状態価値を生成するという学習制御器の構成法である [13].

ここでは，Critic 部を表 4.1 に示す状態価値生成ルールとして記述して，SONE を用いて Actor 部の学習実験を行った．ただし，表 4.1 の入力合計 Sum of inputs は一つの赤外線センサからの入力を 0 から 1 に正規化しており，その合計値として 0 から 8 のレンジを示すものである．また，この状態価値生成ルールは，ロボットの直進行動へ報酬を与え，壁付近に近寄った際に罰を与えることで，速やかな直進回避行動を促すように設定したものである．

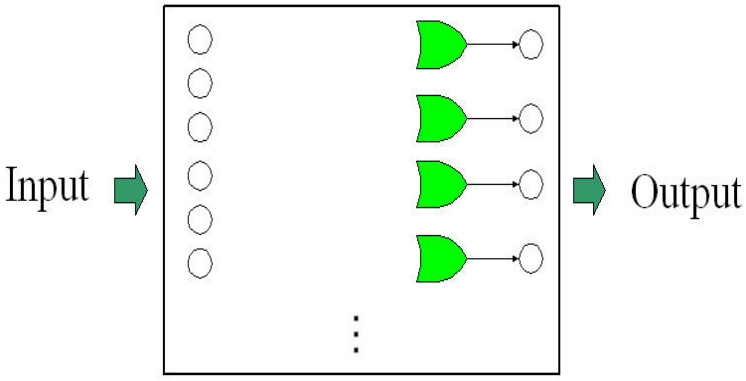


図 4.1 Initial state of network

4.3 実験結果

以上の設定を行った Khepera2 を，障害物のあるフィールド上で移動させる実験を行った．実験開始直後のロボットはランダムな行動をとり，開始地点をしばらくさ迷い続け，約 5 秒後直進行動を開始した．

Khepera2 の胴体下部のペンによって描かれた軌跡を図 4.3 に示す．図 4.3(a) において丸で囲まれた部分の軌跡より，学習初期の段階では壁付近でのランダムな回避行動が試みられていることがわかる．この段階では，壁に向かう，左の赤外線センサが反応したので左旋回を行う，右の赤外線センサが反応したので右旋回を行うといったもがき行動も見られた．その後，図 4.3(b) ではランダムな回避行動が消失し，壁付近での安定した速やかな回避行動が見られるようになった．効率の良い直進回避行動を一度獲得した Khepera2 は，その後も安定して同様の行動を行うことができた（図 4.3(c)）．

表 4.1 Generation rule of state value

State	State value
Near walls	differential of(−Sum of inputs−0.1)
Straight ahead and other than avobe	1
other	-0.001

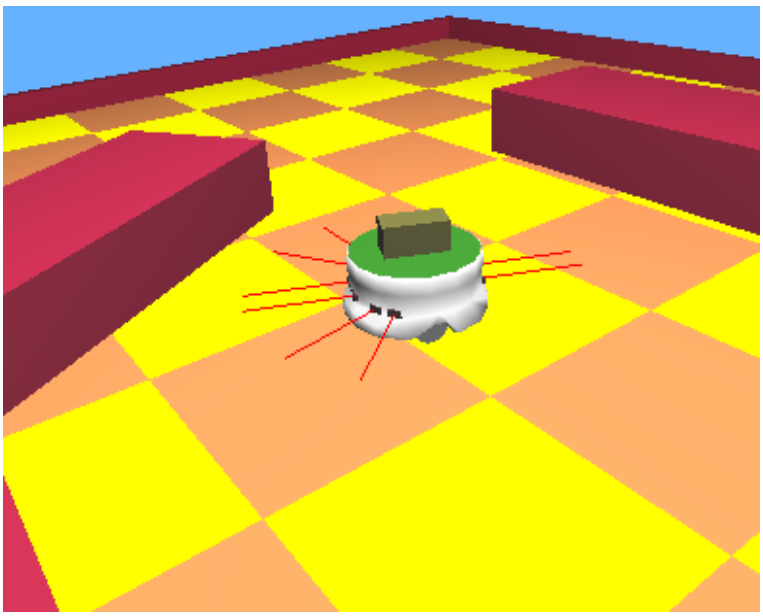
4.4　考察

SONE の作成にあたっての要求仕様である，行動創発，柔軟性，オンライン性の両立について考える．先の実験結果より，学習初期の SONE はロボットをランダムに動作させ，行動創発のための探索を行っている．またこの探索によって，SONE は直進行動や壁の回避方法を発見し，多くの報酬が得られる行動ルールが獲得できている．また，このような学習を行うにあたって，本実験では前章との比較によるパラメータの再設定を行っておらず，柔軟性の確保が再確認できた．さらに本実験はロボットの行動，学習を順次繰り返すオンライン学習によって行っており，環境内でのリアルタイムな学習も実現できている．

表 4.2　Specification of Khepera2

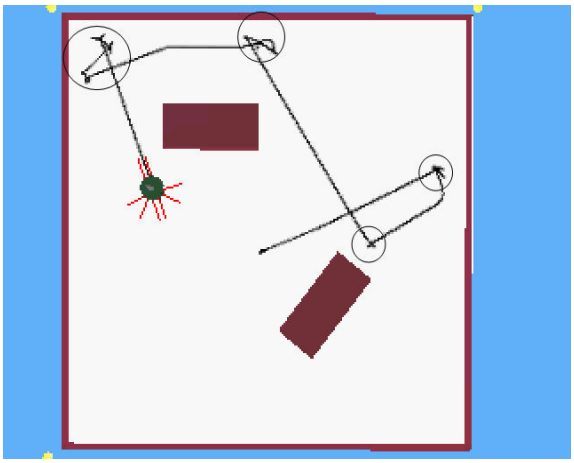
モータ	エンコーダ (12 パルス/mm) 付き DC モータ× 2
速度	2〜60[cm/s]
センサ	赤外線センサ (非飽和有効範囲：14〜70[mm]) x 8 個
走行時間	約 1[h]
サイズ	直径 70[mm]，高さ 30[mm]
自重	約 80[g]

(a) Actual equipment

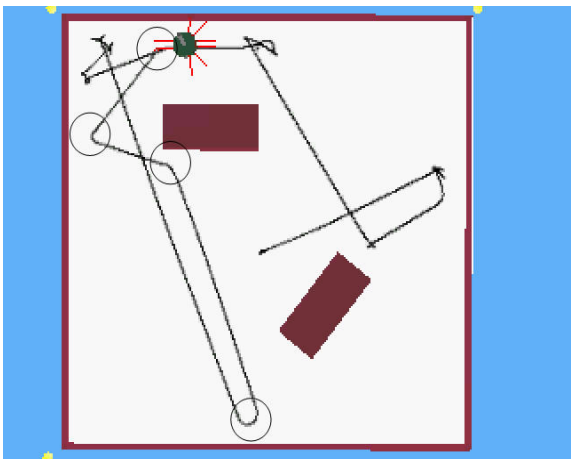


(b) Simulation

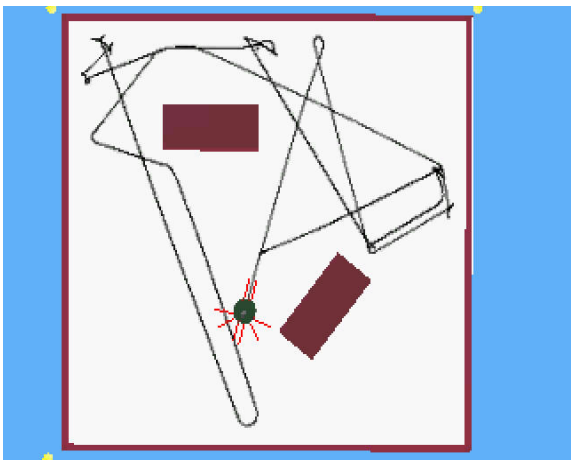
図 4.2 Khepera2



(a) 5 minutes after start



(b) 8 minutes after start



(c) 10 minutes after start

図 4.3　History of robot motion

第5章　SONEにおけるノイズ対策

本章では，SONE の強化学習時に SONE 内部に発生するノイズと，その発生原因について述べるとともに，SONE 内部のノイズを抑制することで，SONE の学習性能を向上する方法を考える．

5.1　ノイズの発生とその影響

SONE の内部には原理的にノイズが発生するため，十分なノイズの除去無しにはさらに効果的な学習が難しい問題があることがわかった．まず，ノイズの発生メカニズムについての説明を行う．

先の章で示したように，SONE は強化学習と教師あり学習の二通りの学習方法をとることができる．図 5.1 は SONE が強化学習を行う際の強化信号の分配法であり移動ロボットの学習制御実験に用いた方法である．また，図 5.2 は SONE が教師あり学習を行う際の強化信号の分配法であり，基本特性の試験に用いた方法である．

著者らは，SONE に教師あり学習を用いる場合との比較によって，強化学習時の SONE にはノイズが発生すると考えた．

教師あり学習ではネットワークの出力それぞれに対して目標値が存在するため，出力と目標値が一致する場合には報酬に相当する正の強化信号を，出力と目標値が一致しない場合には罰に相当する負の強化信号をそれぞれの出力ノードに付与することで学習が行える（図 5.2）．

一方で，強化学習では外部より与えられる強化信号はスカラー量である．よって，SONE の学習方式では，このスカラー量の強化信号をネットワークに分配・付与する必要がある．しかしながら，どの出力ノードが強化信号を得るために寄与したかを加味して分配・付与することは困難であり，ネットワーク出力の貢献度を加味した強化信

号の付与は難しい．そこで SONE の強化学習では，出力ノードに対して均等に強化信号を付与することを行っている．しかし，この方法を教師あり学習と較べた場合，出力ノードの貢献度が加味されていないため，誤って評価される出力ノードが発生すると考えられる．そしてこのようにして発生した誤評価は，個々の出力ノードに対して強化信号のノイズとして作用し，ネットワークの誤学習を発生させると考えられる．

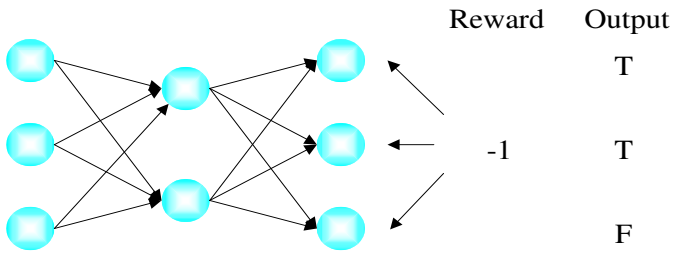


図 5.1　Reinforcement learning

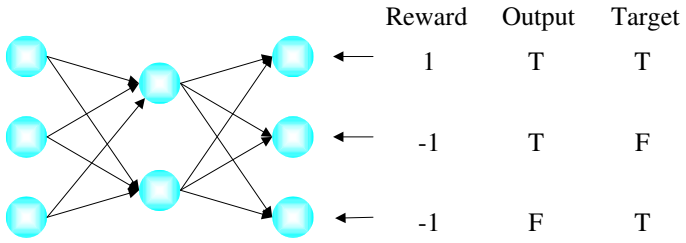


図 5.2　Supervised learning

5.2　ノイズの効果的な除去方法

本節では，ロボットの学習進行状況に応じてノイズを効果的に除去する方法を提案する．

SONE では各リンクの受け取った強化信号の総和である R 値の正負によってリンクの除去に関する判定を行っている．R 値の正負は各リンクの「受け取る強化信号の期待値 Er」の正負に対応し，この Er が正となるリンクを残すことで各ノードの Er を増加できることが証明によって示されている．そして従来機構では，これを利用した

リンクの生成と自己解体によって各ノードの Er を高めることで，ネットワーク全体の Er を増加させている．このような期待値を用いた統計処理によってノイズの影響を軽減させている．

ただし，各リンクの Er はリンクが生成された時点からの強化信号の和である R 値として統計的に観測されるため，十分な信頼度を得るまでには時間がかかる．その信頼度を得るまでの時間を考慮して，SONE では前述である閾値 $Th1, Th2, Th3$ を導入している．

出力ノードの受け取る強化信号にノイズが乗る場合，各リンクの受け取る強化信号にもノイズが乗る．そのノイズ対策としてはこの閾値を十分大きく設定することでノイズの除去を行うことが望ましい．しかしながら，閾値を大きくしすぎるとネットワーク構造の改変が進まず，学習が停滞するという問題が生じる．そこで本論文では学習中に閾値を自動調整する機構を提案し，これらの問題の解決を行う．

5.2.1 閾値の自動調整

ロボットが逐次的に学習を行っていく系においては，学習初期の段階において行動の創発は期待できない．よって，学習初期のネットワークは多くの可能性を試すために新たな素子を多数形成し，探索を行うことが必要である．この場合にはノイズの除去よりも，多くの探索を行うことを優先するために，閾値は低く設定することが望ましいと考えられる．一方，学習が充分に行われた段階では多くの可能性を試すよりも，得られた知識・経験に従って有効な出力を生成することが望ましい．この段階では，多少のノイズにも影響を受けずに正確な Er を算出できるよう，閾値を大きくすることが望ましいと考えられる．

このような考えに従い閾値を自動調整するにあたって，その関数に求められる要件は以下の通りである．

1. 素子の正解率 RC に従って単調増加する
2. RC が 1 に近づくとき，無限大に近づく

3. RC が低いとき，充分に効果的な探索を補償する値をとる

以上を満たす関数としては対数関数が候補に挙がるため，本論文では対数関数を用いて実装を行った．

$$Th = log_B \frac{0.5}{(1 - RC)} + Th(0.5) \tag{5.1}$$

この対数関数を用いるにはパラメータである B，$Th(0.5)$ を決定しなければならない．まず，$Th(0.5)$ は RC が低い際にも充分に効果的な探索を保障するため，従来の SONE と同じ値 $\{Th1, Th2, Th3\} = \{3, 1.5, 1.5\}$ を用いた．対数関数の底である B を決定するにあたっては，ノイズを付加した 3-bit 演算に関する実験を行った．

5.3　評価実験

本論文ではまず，パラメータ B を決定するための 3-bit 演算実験について述べた後，ノイズ除去機構の評価実験である，Webots を用いたロボットシミュレーション実験について述べる．

5.3.1　3-bit 演算に関する実験

本節では，パラメータ B を決定するための 3-bit 演算実験について説明する．3-bit 演算実験は，入力数 3，出力数 1 の論理演算 256 通りのそれぞれの演算をオンライン教師あり学習によって学習する実験である．この実験によって，小規模なネットワーク構造の獲得が確実に行えるか否かを判別できる．

この実験では，ネットワークの初期状態として三つの入力ノードと一つの出力ノードを用意する．ネットワークの行動フェイズでは，入力にランダムな値を設定して出力値を算出させる．伝播フェイズでは，算出された出力値を入力された信号と照らし合わせ，正解であれば出力ノードに正の強化信号 (1) を与え，不正解であれば負の強化信号 (-1) を与えて信号の伝播を行う．構造変更フェイズでは，各素子の R 値をもとに

ネットワークの構造を変更する．以上の操作を繰り返すことで，正解となるネットワーク構造が獲得されるか否かを実験する．

さらにこの実験では，B を求めるためにネットワークをノイズ環境下に置くこととする．先に述べたように，実際の使用環境において出力ノードには強化学習による誤評価が発生するため，各素子の受け取る強化信号にはノイズが乗る．このノイズの代用として，本実験ではネットワークの出力の正否に関わらず，出力ノードに対する報酬値を一定確率で反転させる処理を行い，これをノイズ率とした．

学習の終了条件はネットワークが 1000 ステップ連続正解を達成した時点とし，学習終了後のネットワークに対してテストを行い，入出力の論理関係が正しく形成されているか否かを判別した．

以上の実験をそれぞれの 3-bit 演算，それぞれのノイズ強度，それぞれの B 値に対し各 5 回繰り返し，平均値を算出した．本実験の結果を表 5.1, 図 5.3 に示す．表 5.1 は，図 5.3 の実験結果の一部を抽出したものである．

図 5.3 における各点の色の濃さは正答率を表している．表 5.1 のように B 値が 1〜2 の範囲では，ノイズ発生率が 20〜30[%] であっても正解率が高く，正解までに要した平均ステップ数も低く抑えられているが，ノイズ発生率が 0〜10[%] においては，正解率が 100[%] に至っていない．B 値が 2〜4 の範囲では，ノイズ発生率が 20[%] 以上の場合は正解率が低くなり，ノイズ発生率が 0〜10[%] においては正解率が高くなっている．

この結果をもとに二つの条件を定めて B 値の選定を行い，B を 2.2 と決定した．

1. ノイズ 0[%] の環境下において 100[%] に近い正解率が達成されている

2. ノイズ環境下において高い精度が達成されている．

5.3.2 移動ロボットにおける衝突回避実験

本論文で改良した SONE をロボットシミュレータ Webots5 上の Khepera2 (図 4.2) に実装し，ローカルな素子の耐ノイズ性能の向上が，グローバルなネットワークの強化学習効率へどのように影響するかを検証した．

表 5.1 Correctness ratio

Noise[%]	Previous	B=1.2	B=1.4	B=2.2
0	99	91	99	100
5	98	97	99	100
10	98	94	99	99
15	91	96	97	100
20	67	94	96	97
25	14	96	94	69
30	1	92	76	7

本節では，両輪の回転が一致している場合を直進行動，両輪が停止している場合を停止行動，以上のいずれにも該当しない場合のその他の行動の 3 つの状態を定義してロボットの行動を分類した．ロボットには 1[step] あたり 64[ms] で，56250[step]=1[h] の行動・学習を計 10 回行わせ，その結果を集計した．本実験におけるロボットの振る舞いの結果を表 5.2 に示す．また自己組織化論理回路のノード数の遷移を，10 回の学習に関する平均値として図 5.4 に示す．

表 5.2 Result

SONE	Ahead[%]	Stop[%]	Etc.[%]	Ave. nodes
Previous	29.47	3.15	67.38	254.8
Imploved	38.86	2.21	48.93	438.9

5.4 考察

以下では，3-bit 演算実験，ロボット制御実験に対する考察について述べる．

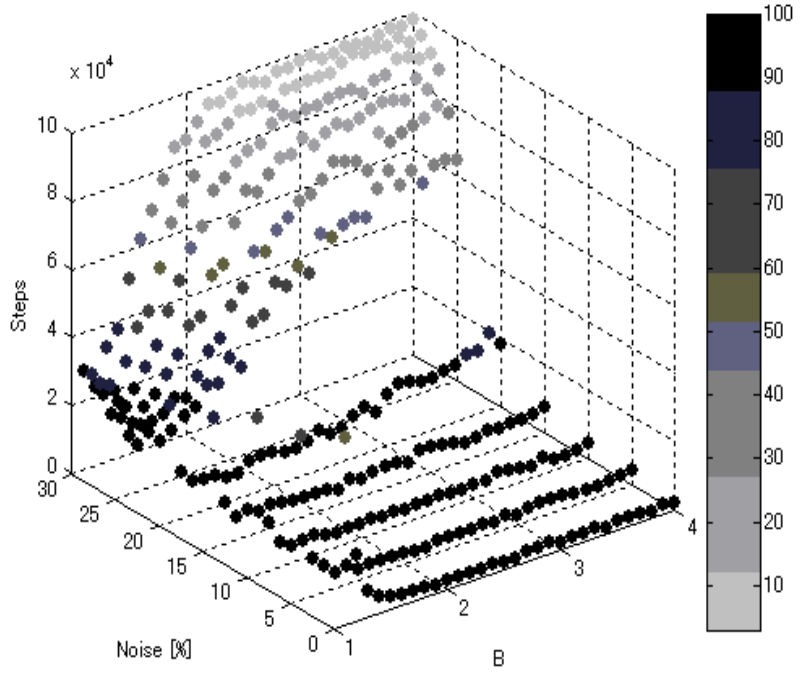


図 5.3 Correctness ratio graph

5.4.1 3-bit 演算実験

5.4.1.1 耐ノイズ性能

SONE のようにローカルルールを基にしてグローバルなネットワークを構成する場合，ローカルな構造の決定は高い精度で行えることが望ましい．表 5.1 より，改良した $B = 2.2$ の SONE では約 20[%] のノイズまでは 100[%] に近い正答率が得られた．そして，以前の SONE と較べ本論文で示した手法は，高い耐ノイズ性能を有する素子を作成するうえで有効であることが確認された．

5.4.1.2 局所解

一般に，ノイズの増加によって正答率が増加するという現象は稀である．しかしながら，表 5.1 の $B = 1.2$ ではその現象が見られる．従来，NN の研究において学習データに微小なノイズを与えることで，局所解の影響を緩和し学習を促進する研究が成さ

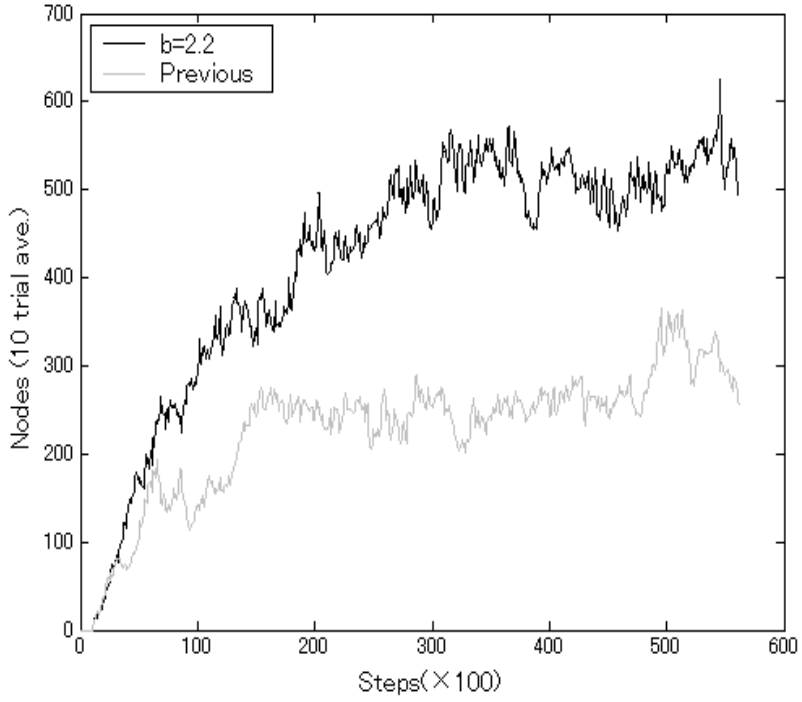


図 5.4 History of the number of nodes

れており，SONEにおける教師あり軌道学習実験においても学習データにノイズを乗せることによって学習が促進する現象が確認できている [69].

SONEにおいてもNNと同様にノイズの影響によって局所解からの脱出可能性が高まり学習が促進されると考えた場合，表5.1の$B = 1.2$に示される実験データには妥当な解釈が行える．この説に従えば，ノイズ0[%] においては局所解の影響で正答率が低かったが，ノイズ5-25[%] 付近では局所解脱出の可能性が高まり正答率が上がった．しかしながら，ノイズ30[%] 付近ではノイズの影響により再び学習困難に陥っていると解釈できる．

5.4.2 移動ロボットにおける衝突回避実験

この実験でも，前章における実験と同様に，学習初期にはランダムな行動を行っていたロボットの出力が改善され，速やかな壁の回避行動と直進行動が次第に見られるようになった．表5.2の結果より，直進行動は29.47[%] から38.86[%] へ増加し，停止

行動は 3.15[%] から 2.21[%] へ低減されたことがわかる．よって，ロボットは与えられた強化信号の解釈としてより適切な行動を獲得できた．

3-bit 演算による実験で決定したパラメータ B を用いることで，別のタスクであるロボット制御に対する改良が見られた．SONE の内部パラメータに関する改良が複数のタスクに対し効果的に作用することが確かめられ，SONE の汎用性を再確認できたといえる．

5.5 まとめ

本節では，報酬量の期待値 Er の推定の精度を決定するパラメータを自動調整する手法として，対数関数を利用した調整法を提案した．提案した調整法を用いた 3-bit 演算実験によって，素子単位での耐ノイズ性能が向上することがわかった．また，状態空間の分割，入力信号の変換，ネットワークトポロジーの決定，評価時間の導入を全く行わない，リアルタイムかつオンラインな自律型ロボット制御シミュレーションにおいて，素子単位での耐ノイズ性能の向上が SONE の学習効率を改善し，より効果的なロボット制御系の獲得に寄与することを確認した．よって，タスク環境毎の設定を省力化した，単純な外部パラメータによるロボットへの実装を可能とした学習器 SONE によって，さらに効果的な出力を有意な時間によって探索・学習することができるようになったといえる．しかしながら，各素子毎のノイズへの対応は依然として 20[%] 程度の水準に留まっており，今後さらなる対策が必要であると考えられる．

第6章　総括

本章では，これまでの自己組織化回路素子に関する基礎実験とロボットシミュレーションのそれぞれに対するまとめに基き，本研究に対する全体の考察と結論，さらには今後の展望について述べる．

6.1　考察

本節では，本論文全体に関連する項目である，強化信号伝播規則の構成，ノイズの抑制，前章まででは取り上げていない他分野の研究との関連を考察し，前章までの考察を補完する．

6.1.1　強化信号伝播規則の構成

本論文で述べた強化信号伝播規則は，筆者らの定めたいくつかの拘束条件のもとで決定している．しかしながら，これらの拘束条件は筆者らの経験則によるものが多く，伝播規則にはまだ改良の余地が有る．

例えば，著者らの示した拘束条件のうち「各素子が自らが受け取った強化信号を他の素子へ伝播する際に，受け取った強化信号以上の信号を周囲の素子へ与えることを禁止する.」という条件は拘束が強すぎるかもしれない．この条件は，ネットワーク内部で自発的に報酬が発生することを禁じているが，強化学習の分野ではシステムの内部で自発的に報酬を発生させるメカニズムの有効性が示されている（例えば [70]）．SONE の作成にあたって与えた拘束条件は系を安定的に学習させるための一つの指針であるが，その最適性は保障できていない．

6.1.2　ノイズの抑制

第 6 章において行ったノイズの抑制に関する実験より，20[%] までのノイズに対して耐性を持つ素子が獲得できた．しかしながら，理想的には 50[%] に近いノイズに対しても順次学習していく中でノイズが除去できる機構が備わっていることが望ましい．なぜならば，第 6 章に示したノイズの発生メカニズムに関する仮説によると，出力ノードの数が増加するについれてノイズの影響が大きくなる．そして，SONE を多出力な系に適用して学習を行った場合には，50[%] に近い領域までのノイズ除去が必要となるだろうと考えられるからである．

6.1.3　他分野との関連

本節ではさらに，他分野との関連について，前章まででは触れていない部分をまとめる．

6.1.3.1　強化信号伝播法と誤差逆伝播法の違い

本論文で提案している SONE は強化信号伝播法によって構成されている．従来，誤差逆伝播法 [1, 71] は脳科学の知見に基いてニューラルネットワークの教師あり学習法として導入されてきた [72]．一方，強化信号伝播法に関する文献は非常に少なく，その多くは強化信号を複数の並列処理システムに分配するに留まっている．SONE のように複雑な形状をしたネットワークに対する伝播法は非常にまれであり，著者らの調査においては発見できなかった．

本節では従来手法である，誤差逆伝播法と強化信号伝播法の違いについて述べる．一つ目の大きな違いは，誤差逆伝播法では出力の値と目標値が等しい場合，ネットワークは実質的に何も学習をしないのに対し，強化信号伝播法ではその時点の出力に関与の深い構造を強化していることである．この構造強化のプロセスによって，有用な情報が淘汰を受けにくくなり，頑健な追加学習が可能となると考えられる．

次に，強化信号伝播法を用いると直接的に強化学習が扱えるということが挙げられる．誤差逆伝播法を用いたニューラルネットワークに強化学習をさせる場合，外界か

ら与えられる強化信号をシステム内部で誤差信号に読み替える必要がある．それに対し，強化信号伝播法では，外界から与えられた強化信号を直接的にネットワークへ伝達して学習することができる．

さらに，誤差逆伝播法ではシナプス荷重等のネットワークパラメータに関する調節はできるが，ネットワークの構造を調節するためには他の方法が必要である．それに対し，強化信号伝播法では各素子に蓄えられた強化信号の量をもとに容易に構造の調整が可能である．

以上のように，強化信号伝播法と誤差逆伝播法は大きく異なった性質を持っており，SONE に見られるような，複雑なネットワークに対する強化信号伝播法は著者らのオリジナルである可能性が高い．

6.1.3.2 ブースティング

SONE による学習法はブースティングとも関連が深い可能性がある [73–76]．ブースティングの概念はもともと，弱い学習能力を持つ学習器を組み合わせて強い学習能力を持つ学習器が構成できるか否かという問題に対し，Schapire が肯定的な結論を打ち出したことに端を発する [77]．ここでは，近似精度 $50+\delta_k$ [%]（k=1,2,⋯,n) の多数の弱仮説（弱分類器）を組み合わせることで，近似精度 $50+\delta$'[%] ($\delta' > \max \delta_k$) となる強仮説（強分類器）を作成することができるかという問いに対し，肯定的な結論が得られており，この概念は多数の学習器を組み合わせることで，正答率の高い学習器を作成できるかという問題に対しても拡張ができる．

ここで，SONE のリンクに着目し，仮にリンクの評価として伝播される強化信号の絶対値が常に一定量である場合を考えよう ($|R| = Constant$)．この場合には，各リンクはそれぞれのネットワークの状態 x に対して T と F のいずれを出力するべきかを分類する分類器として働いている．さらに，$|R| = Constant$ という条件を取り除くと，$|R| = Constant$ では各ネットワークの状態 x に対する T と F の分類作業の重要度が一定であったものが，変化することになる．つまり，分類作業（問題）の重要度に重み $|R|$ が加わったものとして解釈できる．よって，リンクの評価として伝播される強化信

号の絶対値が変化する場合においても，各リンクは重み付きの問題に対する分類器として作用していると捉えることができる．

一方，SONE では各リンクの淘汰基準を強化信号の期待値 Er が 0 を下回った場合としている．これは，$|R| = Constant$ とした場合には，各ネットワークの状態 x に対する T と F の分類精度が 50[%] 未満のリンクが淘汰されることに等しい．また，$|R| = Constant$ という条件を取り除いた場合には，先ほどと同様に，重み $|R|$ が加わった分類において分類精度が 50[%] 未満のリンクが淘汰されることに等しい．

以上のように，SONE の各リンクは Schapire の言う弱仮説（弱分類器）として捉えることができる．また強化信号伝播規則を定めるにあたって，SONE では，ノードがより多くの強化信号を得るために貢献しているリンクのみが生き残るように強化信号を伝達している．そしてこれは，$\cup_i L_T(i)$ が A_T をより良く近似するために貢献しているリンクを生存させる機構と言い換えることができる．つまり，SONE で追加される新しいリンクは，各ノードのがそれぞれのネットワークの状態 x に対して T と F のいずれを出力するべきかを分類するために組み合わされており，さらには，その組み合わせの効果によって近似精度を増大させているといえる．この意味で，SONE のノードを強分類器として捉えることができる．

以上の考察から，SONE はブースティングの効果を利用して学習している可能性が高いと考えられる．しかしながら，通常のブースティングは，一階層のネットワークにおける重みを学習する場合が多く，SONE のようにオンラインに複雑なネットワーク構造を自己組織化する系におけるブースティングアルゴリズムは筆者の調査範囲において発見できなかった．この関連性に対してはさらに調査を行っていく必要がある．

6.1.3.3　マルチエージェント

SONE の各素子は，報酬量を増大させるように構造の変更を行うエージェントとして捉えることができる．この意味で SONE による学習はマルチエージェントの考え方に近い [78–80]．マルチエージェントの分野では，エージェント同士の相互作用による新しい秩序の創発や，エージェントの集団全体としてのタスクの遂行に関する研究等が行われている．

SONE をマルチエージェントとして捕らえた場合，ここでのエージェントに相当する，ネットワーク素子（Or ノード，And ノード，反転リンク，非反転リンク）は，個々独立したルールと，その相互作用によってネットワークを創発している．さらに，これらのエージェントで構成されたネットワークは，個々の素子で表現可能な能力を大きく超えたルールを備えており，タスク遂行能力も高い．よって，SONE はマルチエージェントに関する研究としても非常に興味深い対象だと考えられる．

6.1.3.4 スモールワールドネットワーク

SONE はスモールワールドネットワークに関する研究とも関連がある可能性がある [81–85]．スモールワールドネットワークは，Watts らによって提唱されたネットワークの様相に関する研究であり，格子状のネットワーク等の整然とした結合関係を持つネットワークと，ランダムネットワークの中間に位置するネットワークの効率等が議論されている．また，この考え方は，インターネットや神経ネットワーク [86,87]，または社会構造のネットワーク等，多くのネットワーク構造に対してあてはまる．特に Lu らによる研究 [86] では，教師なし学習を行うネットワークの構造をスモールワールドネットワークとして設計することで，ネットワークの記憶容量が増大するという事例が報告されており，SONE においても同様の効果が認められるか否かを検討していく必要がある．

6.2 結論

本論文では，自律型ロボットへ使用する学習制御器の要求仕様（行動創発，汎化・抽象化，柔軟性，オンライン性，漸次性）をまとめ，これらを両立できる学習制御手法として自己組織化回路素子 Self-Organizing Network Elements (SONE) を提案した．さらに，5 種類の論理回路素子（Or ノード，And ノード，FF ノード，反転リンク，非反転リンク）に対して提案手法の実装法を示すとともに，学習効果に関する証明と冗長性に関する証明を行った．

第 3 章の基本特性に関する試験では，要求仕様のうち特に汎化・抽象化，柔軟性，オンライン性，漸次性の四項目に対して試験を行うために，軌道学習に関する試験と二重螺旋問題を用いた試験を導入した．

軌道学習に関する試験は耐ノイズ性能試験，時系列問題に対する性能試験，追加学習性能試験を行い，それぞれ SONE によって実現されるネットワーク構造は環境からのノイズに耐性を持つこと，SONE にフリップフロップ素子を応用するとリカレントニューラルネットワーク（RNN）に較べ，長時間の隠れ状態を伴った時系列問題を学習できること，RNN と較べて頑健な追加学習ができること等が明らかとなった．また，耐ノイズ性能試験から SONE が汎化学習している可能性が示唆され，FF ノードの代表しているシーケンスの解析から抽象化の能力が確認できた．以上の試験の全てをネットワークパラメータの変更無しにオンライン学習によって行えたことから柔軟性とオンライン性が確認できる．そして追加学習試験によって RNN との比較において高い漸次性を有することがわかった．

二重螺旋問題を用いた試験からは，SONE に汎化能力があることの裏づけが得られた．また，複雑な線形分離問題が扱えることも明らかとなった．

第 4 章では，行動創発，柔軟性，オンライン性の両立を確認するために，移動ロボットにおける衝突回避実験を導入した．この実験ではシミュレーション上の移動ロボットの行動学習を通じて，SONE により効果的な行動創発が行えること，第 3 章の実験と同様のパラメータによって学習可能であること，オンライン・リアルタイムに学習ができること等を明らかとした．

第 5 章では，SONE の内部には原理的にノイズが発生するという仮説に対し，SONE を構成する各素子に対するノイズ耐性を高めるための手法を提案し，検証実験を行った．この結果，3-bit 演算に関するノイズ耐性の高い素子を得ることに成功した．さらには，その SONE で移動ロボットにおける衝突回避実験を行うことで，SONE の素子単位でのノイズ耐性の向上がロボットタスクにおける強化学習性能を向上させることが明らかとなった．

6.3 今後の展望

以上のように，SONE は開発当初の目的を達成しており，今後はその性能向上に関する研究に焦点が当てられる．以下では，SONE の連続性，強化信号伝播規則の構成論，ノイズ除去，強化学習時の Critic の実現についての今後の展望について述べる．

6.3.1 連続性

現在，SONE へ使用できる素子は論理回路素子に限定されているため，連続性のあるダイナミクスの記述が難しい，ネットワークの素子数が増大しやすいという問題がある．これは，論理回路素子の入出力が不連続であることに起因する．

この問題はネットワークを構成するために用いる素子にニューロ素子や Radial Basis Function を利用した RBF 素子を用いることで解決できる可能性がある．ただし，これらの素子を用いる場合にはそれぞれの素子の持つパラメータを学習によって調整する必要が生じるため，強化信号伝播規則の他にその調整のための仕組みも考える必要がある．

6.3.2 強化信号伝播規則の構成論

本論文で示した強化信号伝播規則は論理回路素子に限定されているため，他の一般的な素子を用いた場合には同じ方法で実装できるとは限らない．また，本論文で示した論理回路素子に対する強化信号伝播規則も論理回路素子に対しての最適な規則ではない．よって，今後さらに研究を深めるためには一般的な素子に対して最適な伝播規則を設定するための方法論が必要である．

しかしながら，対象とする素子を一般化したままでその方法論を議論することは非常に難しいため，いくつかの素子に対する伝播規則を実験的に求め，その規則に関する知見から一般化を行うのが良いように思われる．

個別の素子に対して強化信号伝播規則を作成するための方法にも，実験的手法と理論的手法の両方が考えられる．本論文で示した実験的手法では，素子単体の性能を評

価するためのベンチマークとなる試験（3bit 演算に関する試験等) を設け，実験を繰り返し，素子単体の性能向上を行うことで，SONE で構成されるグローバルネットワークに関しても性能向上が見られるということがわかっている．ただし，ベンチマークはノイズの有無や演算精度，演算時間といった複数の尺度から構成できるため，そのいずれに対しても有効な規則を構成するための方法論が必要である．

6.3.3　ノイズ抑制

本論文で行った方式によるノイズ除去では各素子につき 20[%] 程度のノイズにまでしか対応できていない．しかしながら，サンプルデータを追加的に得られる環境にいるロボットを考えた場合，学習の進度に応じて 50[%] に近いノイズをも除去できる方式が取り得る可能性はある．つまり，環境からのサンプルデータ数が無限大に近づき，環境内で起こり得るほぼ全ての状況を網羅できる場合には完全に近いノイズ除去が可能となるからである．

ただし，そのような機構を考えた場合には SONE が素子を淘汰する方式をもう一度見直す必要があるかもしれない．なぜならば，強化信号に 50[%] に近いノイズが加わった状況（ほぼランダムな状況）においては，本来であれば有効である素子も淘汰を受けやすくなるため，素子が淘汰された時点での情報の損失が起こるからである．

ノイズ除去に関する SONE の利点は，各素子毎が生成淘汰の基準となる強化信号の期待値に関する計算する再に，過去の全てのローカルなサンプルに渡っての平均値計算がサンプル数によらず同等の計算時間で成されることで，計算効率の良い平均化によるノイズ除去が実現できることである．しかし SONE は動的に構造を変更するため，素子が淘汰されてしまい，また新たに同じ結合の素子ができあがっても，そこには以前のサンプルの履歴に関する情報記述されていないため，ノイズ除去に必要なサンプル数が確保できなくなるという問題がある．

ここでは，動的に変化する環境へ対応するためにはネットワーク構造に柔軟性をもたせる必要があるが，ネットワーク構造に柔軟性をもたせ素子を淘汰すると，データの損失によって完全なノイズ除去に至ることはできないというジレンマがある．

素子を淘汰した際に，そのデータを完全に消去してしまうのではなく，ネットワークの実体とは別に保存しておき，同じ結合の素子を作成する際に再利用することもできるが，計算量の点では実用上の困難があるだろう．

6.3.4 Critic の実現

第 4 章における，移動ロボットにおける衝突回避学習実験では，Actor-Critic 法に従って SONE をロボットの Actor に用いることとした．そして，Critic は著者らが構成する方法をとっている．一般に，Actor-Critic 法を用いたロボットが遅延報酬に対して適切な学習を行うためには，Actor と Critic の両方に対しての学習が必要であると言われており，今後は Critic の構成法も検討していく必要がある．

参考文献

［1］E. D. Rumelhart, G. E. Hinton, R. J. Williams. Learning representations by back-propagating errors. *Nature*, Vol. 323, p. 533, 1986.

［2］Philip D. Wasserman. ニューラル・コンピューティング―理論と実際. 森北出版, 1993.

［3］馬場則夫, 小島史男, 小沢誠一. ニューラルネットの基礎と応用. 共立出版, 1994.

［4］武藤佳恭, 斎藤孝之. 応用事例ハンドブック ニューラルコンピューティング. 共立出版, 2001.

［5］坂和正敏. ニューロコンピューティング入門. 森北出版, 1997.

［6］萩原将文, 田中雅博. ニューロ・ファジー・遺伝的アルゴリズム. 産業図書, 1994.

［7］松本元, 大津展之. 脳とコンピュータ 1 ニューロコンピューティング. 培風館, 1992.

［8］西森秀稔. ニューラルネットワークの統計力学. 丸善, 1995.

［9］甘利俊一. ニューロコンピュータ読本. サイエンス社, 1989.

［10］Teuvo Kohonen. 自己組織化マップ. シュプリンガーフェアラーク東京, 2005.

［11］徳高平蔵. 自己組織化マップの応用. 海文堂, 1999.

［12］秋山剛 (他多数). AI 辞典. 共立出版株式会社, 2003.

［13］Richard S. Sutton, Andrew G. Barto. 強化学習. 森北出版, 2000.

[14] Anonimous. Honda debuts new asimo. *Electromagnetic News Report*.

[15] Anonimous. Sony releases 'aibo' robot dog with diary-keeping function. *Knight Ridder Tribune Business News*.

[16] http://www.humanoid.waseda.ac.jp/index-j.html.

[17] Albert F.C. Haldemann James K. Erickson, John L. Callas. The mars exploration rover project:2005 surface operations results. *Acta Astronautica*, Vol. 61, pp. 699–706, 2007.

[18] 尹祐根, 高橋三恵, 鎌田大輔, 妻木勇一, 内山勝. 双腕宇宙遠隔操作実験システムの構築, 2003.

[19] Tetsuya Ogata, Shigeki Sugano. Communication between behavior-based robots with emotion model and humans, 1998.

[20] 仲川こころ, 小杉大輔, 安田有里子, 小嶋秀樹. Keepon：子どもからの自発的な関わりを引き出すぬいぐるみロボット. 人工知能学会 言語・音声理解と対話処理研究会 (NICT 京都), pp. 7–14, 2004.

[21] 三輪, 石引, 荒井, 西嶋. 身体性に着目したエントレインメント創出過程の計測. ヒューマンインタフェース学会論文誌, Vol. 2, No. 2, pp. 185–191.

[22] 北野宏明（編）. 遺伝的アルゴリズム. 産業図書, 1993.

[23] C. J. C. H. Watkins. *Learning From Delayed Rewards*. Ph.D. thesis of Cambridge University, 1989.

[24] C. J. C. H. Watkins. Q-learning. *Machine Learning*, Vol. 8, pp. 279–292, 1992.

[25] R. E. Bellman. A markov decision process. *Journal of Mathematical Mechanics*, Vol. 6, pp. 221–229.

[26] 小林祐一, 湯浅秀男, 細江繁幸. 強化学習のための矩形基底による自律分散型関数近似. 計測自動制御学会論文集, Vol. 40, No. 8, 2004.

[27] Y.Takahashi, M.Asada. Multi-controller fusion in multi-layered reinforcement learning, 2001.

[28] 高橋泰岳, 浅田稔. 階層型強化学習機構における状態行動空間の構成. 日本ロボット学会誌, Vol. 21, No. 2, pp. 164–171, 2003.

[29] 柴田克成, 岡部洋一, 伊藤宏司. ニューラルネットを用いた強化学習－センサからモータまでの合目的的・調和的学習－. 計測自動制御学会論文集, Vol. 37, No. 2, pp. 168–177, 2001.

[30] Kenneth O. Stanley, Risto Miikkulainen. Efficient reinforcement learning through evolving neural network topologies, 2002.

[31] 宇谷明秀, 小林元, 山崎裕司, 登坂宣好. 自己構造化ニューラルネットワーク：構造と結合係数の統合学習アルゴリズム. 日本計算工学会, No. 20010043, 2001.

[32] Kenneth O. Stanley, Bobby D. Bryant, Ristio Miikkulainen. Real-time neuroevolution in the nero video game. *IEEE Transactions on Evolutionary Computation*, Vol. 9, p. 653, 2005.

[33] Shimon Whiteson, Peter Stone. Evolutionary function approximation for reinforcement learning. *Machine Learning*, Vol. 7, pp. 877–917, 2006.

[34] Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, Vol. 3, No. 4, 1999.

[35] Bernard Ans, Stephane Rousset. Avoiding catastrophic forgetting by coupling two reverberating neural networks. *Neuroscience*, Vol. 320, pp. 989–997, 1997.

[36] A. Robins, S. McCallum. The consolidation of learning duaring sleep:comparing the pseudorehearsal and unlearning accounts. *Neural Networks*, 1999.

[37] 浅田稔, NPO ロボカップ日本委員会. ロボットの行動学習・発達・進化 — RoboCup-Soccer. 共立出版, 2001.

[38] 松原仁, 竹内郁雄, 沼田寛. ロボットの情報学. NTT 出版, 2001.

[39] Hiroaki Kitano. Neurogenttic learning: an integrated method of designing and training neural networks using genetic algorithms. *Physica D*, Vol. 75, pp. 225–238, 1994.

[40] Mototaka Suzuki, Dario Floreano. Evolutionary active vision toward three dimensional landmark-navigation. *Lecture note in computer science*, 2005.

[41] Masumi Ishikawa. Structural learning with forgetting. *Neural Networks*, Vol. 9, No. 3, pp. 509–521, 1996.

[42] Jie Ni, Qing Song. Dynamic pruning algorithm for multilayer perception based neural control systems. *Neurocomputing*, Vol. 69, p. 2097, 2006.

[43] Gang Leng, Girijesh Prasad, Thomas Martin McGinnity. An on-line algorithm for creating self-organizing fuzzy neural networks. *Neural Networks*, Vol. 17, pp. 1477–1493, 2004.

[44] Weiming Hu, Dan Xie, Tieniu Tan, Steve Maybank. Learning activity patterns using fuzzy self-organizing neural network. *IEEE Transactions on systems, man, and sybernetics-PART B: CYBERNETICS*, Vol. 34, No. 3, 2004.

[45] Russell Reed. Pruning algorithms-a survey. *IEEE Transactions on Neural Networks*, Vol. 4, No. 5, 1993.

[46] Penqfei Xu, Chip-Hong Chang. Self-organizing topological tree, 2004.

[47] Frank Dieterle, Stefan Busche, Gunter Gauglitz. Growing neural networks for a multivariate calibration and variable selection of time-resolved measurements. *Analytica Chimica Acta*, Vol. 490, pp. 71–83, 2003.

[48] 柴田尚子, 北川輝彦, 福永哲也. 強化学習を使った自己組織化ネットワークによる自律移動ロボットの行動制御, 2003.

[49] Bertrand Mesot, Christof Teuscher. Deducing local rules for solving global tasks with random boolean networks. *Physica*, Vol. 211, pp. 88–106, 2005.

[50] Paul J. Welbos. Back propagation through time: What it does and how to do it, 1990.

[51] E.B. Baum, K.J. Lang. Constructing hidden units using examples and queries. *Advances in Neural Information Processing Systems*, Vol. 3, pp. 904–910, 1991.

[52] Susan B. Garavaglia. The two spirals benchmark lessons from the hidden layers. *IEEE Transactions*, 1999.

[53] Lang K. J., Witbrock M. J. Learning to tell two spirals apart, 1988.

[54] S.E. Fahlman, C. Lebiere. The cascade-correlation leaming architecture. *Advances in Neural Information Processing Systems*, Vol. 2, pp. 524–532, 1993.

[55] David Weenink. Category art: A variation on adaptive resonance theory neural networks, 1997.

[56] Gail A. Carpenter. A massively parallel architecture for a self-organizing neural pattern recognition machine, 1987.

[57] Jiancheng Jia, Hock-Chum Chua. Solving two-spiral problem through input data representation, 1995.

[58] Ronald J. Williams, Jing Peng. An efficient gradient-based algorithm for on-line training of recurrent network trajectory. 1990.

[59] Ronald J. Williams, Daviod Zipser. A learning algorithm for continually running fully recurrent neural network. 1998.

[60] Ronald J. Williams, Daviod Zipser. Gradient-based learning algorithms for recurrent networks and their computational complexity. 1995.

[61] A.A. Vartaka, M. Georgiopoulosa, G.C. Anagnostopoulos. On-line gauss newton-based learning for fully recurrent neural networks. *Nonlinear Analysis*, Vol. 63, pp. e867–e876, 2005.

[62] M. W. Mak, K. W. Ku, Y. L. Lu. On the improvement of the real time recurrent learning algorithm for recurrent neural networks. *Neurocomputing*, 1999.

[63] Sepp Hochreiter, Jurgen Schmidhuber. Long-short term memory. *Neural Computation*, Vol. 9, No. Vol.9 pp.1735-1780, pp. 1735–1780, 1997.

[64] Baum Bakker. Reinforcement learning with long short-term memory. *Neural Information Process System*, 2001.

[65] Jun Tani, Masato Ito. Self-organization of behavioral primitives as multiple attractor dynamics: A robot experiment. *IEEE Transaction on System Man and Cybernetics A*, Vol. 33, No. 4, pp. 481–488, 2003.

[66] Reiner W. Paine, Jun Tani. Evolved motor primitives and sequences in a hierarchical recurrent neural network. *Lecture Nodes in Computer Science*, Vol. 3, pp. 603–614, 2004.

[67] Jun Tani, Masato Ito, Yuuya Sugita. Self-organization of distributedly represented multiple behavior schemata in a mirror system: reviews of robot experiments using rnnpb. *Neural Networks*, Vol. 17, pp. 1273–1289.

[68] 本村亮, 横井博一. リーマン幾何学を用いた人工ニューラルネットワークの汎化機能の評価法. 電子情報通信学会, 2003.

[69] Chyon Hae Kim, Tetsuya Ogata, Shigeki Sugano. Enhancement of self organizing network elements for supervised learning, 2007.

[70] Johane Takeuchi, Osamu Shouno, Hiroshi Tsujino. Modular neural network for reinforcement learning with temporal intrinsic rewards, 2007.

[71] D. B. Parker. Learning logic. *Office of Technology Licensing in Stanford University*, 1982.

[72] 臼井支朗. 脳・神経システムの数理モデル. 共立出版, 1997.

[73] Yoav Freund, Robert E. Schapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, Vol. 14, No. 5, pp. 771–780, 1999.

[74] Yorv Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 1995.

[75] Robert E. Schapire. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, Vol. 37, pp. 297–336, 1999.

[76] Ron Meir, Gunner Ratsch. An introduction to boosting and leveraging.

[77] Robert E. Schapire. The strength of weak learnability. *Machine Learning*, Vol. 5, pp. 197–227, 1990.

[78] 高玉圭樹. マルチエージェント学習 ー相互作用の謎に迫るー. コロナ社, 2003.

[79] 生天目章. マルチエージェントと複雑系. 森北出版株式会社, 1998.

[80] 山影進, 服部正太. コンピュータのなかの人工社会. 発行:構造計画研究所 販売:共立出版, 2002.

[81] Duncan J. Watts, Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, Vol. 393, pp. 440–442, 1998.

[82] Duncan J. Watts. Small worlds: The dynamics of networks between order and randomness. *Princeton Univ. Press*, 1999.

[83] Mark Buchanan. 複雑な世界、単純な法則 ネットワーク科学の最前線. 草思社, 2005.

[84] L. A. N. Amaral, A. Scala, M. Barthelemy, H. E. Stanley. Classes of small-world networks. *Applied Physical Science*, Vol. 97, No. 21, pp. 11149–11152, 2000.

[85] M. E. J. Newman, D. J. Watts. Renormalization group analysis of the small-world network model. *Physics Letters A*, Vol. 263, pp. 341–346, 1999.

[86] Jianquan Lu, Juan He, Jinde Cao, Zhiqiang Gao. Topology influences performance in the associative memory neural networks. *Physics Letters A*, Vol. 354, pp. 335–343, 2006.

[87] Juan I. Perotti, Francisco A. Tamarit, Sergio A. Cannas. A scale-free neural network for modelling neurogenesis. *Physica A*, Vol. 371, pp. 71–75, 2006.

謝辞

本研究を遂行するにあたり，懇切な御指導と御激励を賜りました早稲田大学理工学部機械工学科　菅野重樹教授に心より感謝致します.

本論文をまとめるにあたり，有益な御助言と御討論を賜りました，早稲田大学理工学部　藤江正克教授，山川宏教授，高西淳夫教授をはじめとする，機械工学科の諸先生方に深く感謝致します.

また，京都大学工学部准教授尾形哲也氏，早稲田大学高等研究所准教授岩田浩康氏，早稲田大学創造理工学部助手菅佑樹氏，早稲田大学 COE 助手有江浩明氏，WABOT-HOUSE 客員研究員坂本義弘氏をはじめとする，菅野重樹研究室の諸先輩後輩方，および早稲田大学 21 世紀 COE プロジェクトの方々に深く感謝致します．出澤純一君，阿部博行君をはじめとする菅野研究室学習研究グループの諸氏からは，研究遂行にあたり多大な協力を頂きました．ここに記して感謝致します.

さらに，学会や研究会を通じて多くの先生方，研究者の方に様々な御指導を頂きましたことに厚くお礼申し上げると共に今後のご活躍をお祈り致します.

研究業績

金　天海

種類別	題名	発表・掲載誌名	発表発行年月	連名者
1 ○論文	自己組織化論理回路におけるノイズの抑制	日本ロボット学会誌	2007 年	出澤純一 尾形哲也 菅野重樹
○論文	Enhancement of Self Organizing Network Elements for Supervised Learning	IEEE International Conference on Robotics and Automation	2007 年	尾形哲也 菅野重樹
○論文	ローカルルールに基いた論理回路の自己組織化アルゴリズム	計測自動制御学会論文誌	2006 年	尾形哲也 菅野重樹
○論文	Efficient Organization of Network Topology based on Reinforcement Signals	IEEE International Conference on Intelligent Robots and Systems	2006 年	尾形哲也 菅野重樹
○論文	Improvement against Noises in Self-Organizing Logic Circuit	IEEE International Conference on Information Acquisition	2006 年	尾形哲也 菅野重樹
○論文	Self-Organizing Algorithm for Logic Circuit based on Local Rules	IEEE/ASME International Conference on Advanced Intelligent Mechatronics	2005 年	尾形哲也 菅野重樹
2 講演	自己組織化回路素子 (SONE) への教師あり学習の付与	情報処理学会	2007 年	尾形哲也 菅野重樹

種類別	題名	発表・掲載誌名	発表発行年月	連名者
講演	自己組織化回路素子SONEへの教師あり学習機能の付与	計測自動制御学会システムインテグレーション部門講演会	2006年	尾形哲也 菅野重樹
講演	自己組織化回路素子SONEにおけるフリップフロップ素子導入によるシーケンスの分節化と統合	計測自動制御学会システムインテグレーション部門講演会	2006年	尾形哲也 菅野重樹
講演	自己組織化論理回路における対ノイズ性能の向上	日本ロボット学会学術講演会	2006年	尾形哲也 菅野重樹
講演	自己組織化論理回路における学習アルゴリズムの解析	計測自動制御学会システムインテグレーション部門講演会	2005年	尾形哲也 菅野重樹
講演	ローカルルールに基づいた論理回路の自己組織化アルゴリズム	計測自動制御学会システムインテグレーション部門講演会	2004年	尾形哲也 菅野重樹
講演	自己組織化ネットワーク素子群における対ノイズ性能向上	ロボティクス・メカトロニクス講演会	2006年	出澤純一 尾形哲也 菅野重樹
講演	An Algorithm for Self-Organizing Logic Circuit	The 3rd COE-CIR Joint Workshop	2006年	尾形哲也 菅野重樹
講演	Self-Oganizing Logic Circuit based on Local Rules	The 2nd COE-CIR Joint Workshop	2005年	尾形哲也 菅野重樹
特許	情報処理システムおよび情報処理方法，並びにプログラム	特願2004-363742	2004年	尾形哲也 菅野重樹
特許	情報処理システムおよび情報処理方法，並びにプログラム	PCT/JP2005/21062	2005年	尾形哲也 菅野重樹
特許	情報処理システムおよび情報処理方法，並びにプログラム	特開2006-172141	2006年	尾形哲也 菅野重樹