ロボットの内部情報処理に対する言語教示可能性

The Possibility of Instructing the Hidden Information Processing of a Robot

本多透[*1] 板舛尚樹[*1*2] 岡夏樹[*1]
Toru Honda Naoki Itamasu Natsuki Oka

[*1] 京都工芸繊維大学
Kyoto Institute of Technology

[*2] 現 NEC ネッツエスアイ株式会社
Now at NEC Networks & System Integration Corporation

Although a modular architecture is promising for large and complicated problems, the number of combinations of the modules increases exponentially if there are no constraints on the module combination. This paper proposes a new method for giving constraints on the module combination by instruction. Instructions of an abstract level are given to a learning system with a modular architecture, and the system learns to compile the instructions into a series of module recombinations, which are hidden state changes to the instructor. An experiment showed that the system successfully learned module recombinations which correspond to given instructions.

1. はじめに

知的ロボットが様々な問題を解決する際に，行動のすべてをプログラミングするのではなく，ロボット自身に学習させ，変動する環境に対応させようと，様々な手法が提案されてきたが，その中でも，人間の脳は各領域が異なる機能を持ち，問題を解決する際にはそれらを組み合わせていくことで，複雑な問題に素早く適応している，という脳のモジュール性に注目した研究がある［小川 2004］．これは，学習を行う部分や，ロボットの要素的な機能をモジュール化し，それらの組み換えを行うことで問題を解決するシステムである．このように，各機能をモジュール化（学習済みのモジュールを共有）することで以下のような利点がある．

- 複雑な問題もモジュールごとに学習させることで 1 つのシステムで多様な問題に対応できる．
- モジュールを取り換えることで学習をし直さずに異なる動きをすることができる
- モジュールの組み換えを学習させることで，ロボットが自分で新しい行動を獲得することが可能になる．

しかし，モジュール数の増加に伴い，組み合わせの数が爆発するという問題は依然として残っている．モジュール数を n とし，1つの行動を達成するために必要なモジュールの組み換え回数を p とすると組み合わせの数が 2^{np} となってしまい，学習の際の計算量はさらに環境の状態数に比例して増加する．よって実用化するためには組み合わせを限定することなどが必要である．

そこで，本研究ではこの問題を解決する方法として，モジュールの組み換えという内部情報処理と言語教示（今回だと「光を見て動いて」など）を対応付ける方法を提案し，シミュレーションを行い，評価する．これは，環境とから直接モジュールの組み換えを学習するのではなく，抽象的なレベルの中間目標を教示として与え，その教示とモジュールの組み換えの対応付けを学習し，次にその教示と環境との対応付けを学習することで，問題の解決を容易にするという試みの前段階にあたるものである．これにより計算量が“環境の状態数× 2^{np} ”であった計算量が“中間目標数× 2^{np} ”と“環境の状態数×中間目標数”の和になり，うまく中間目標となる教示を与えることで計算量が減少し，学習が早く進むことを期待する．

連絡先：本多透，京都工芸繊維大学 大学院工芸科学研究科，e-mail：m9622032@edu.kit.ac.jp

2. モジュール組み換え型アーキテクチャ

本研究で用いたモジュール組み換え型アーキテクチャ[Oka 1999]のモデルを図 1 に示す．このモデルは制御部からの制御信号で，各モジュール間の結合の on-off を操作し，結合が on になることで両端のモジュールが機能するというものである．以降，場面ごとに on にするモジュールを制御信号によって変えていくことをモジュールの組み換えと呼ぶ．入力部はモジュールごとに対応したセンサーの値を処理するモジュールで，値は状態メモリに一時的に記憶される．方策部は状態メモリの値を読み取って，ロボットの取るべき行動を決定し，出力するモジュールで，とるべき行動は行動メモリに保持される．出力部のモジュールはロボットの出力に対応している．制御部は図2のようにいくつかのモジュールから構成されており，詳細は後述する．今回のモデルでは，“光認識モジュール→状態メモリ→光方策モジュール→行動メモリ→行動モジュール”または，“音認識モジュール→状態メモリ→音方策モジュール→行動メモリ→行動モジュール”というふうに，環境に応じて適切な入力部のモジュールを選択し，それに対応した方策部のモジュールにより行動を決定し，行動モジュールに出力することでセンサー入力に対して適切な行動を出力することができる．

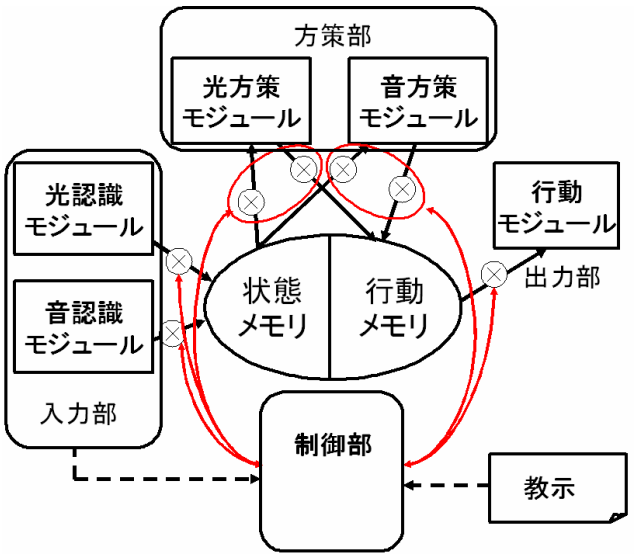

図1 モジュール組み換え型アーキテクチャ

2.1 入力部

(1) **光認識モジュール**

このモジュールは後述するシミュレーション環境下における現在のエージェントの位置から光源までの距離を認識するモジュールである．状態メモリとの結合が on になることで，現在の値を出力する．

(2) **音認識モジュール**

光認識モジュールと同様に，現在のエージェント位置から音源までの距離を認識するモジュールである．

2.2 方策部

(1) **光方策モジュール**

光センサーからの情報に基づいてロボットが取るべき行動を決定するモジュールである．今回はモジュールの組み換えを学習するのが目的なので，学習させず設計したモジュールを使う．状態メモリにある情報に基づいて光源に向かう行動を出力する．

(2) **音方策モジュール**

音センサーからの情報に基づいてロボットが取るべき行動を決定するモジュールである．光方策モジュールと同様に設計したものをもちいた．状態メモリにある情報に基づいて音源を避ける行動を出力する．

2.3 出力部

(1) **行動モジュール**

行動メモリにある情報に基づいてロボットを動かすモジュールである．

2.4 制御部

制御部のモジュール構成を図 2 に示す．制御部は 3 つのモジュールからなり，制御学習モジュールは環境情報と現在のモジュール間結合の on-off の状態を入力として各モジュール結合を制御する信号を出力し，教示意味学習モジュールは，現在のモジュール間結合の on-off と与えられた教示を入力として，各モジュール結合を制御する信号を出力するモジュールである．2つのモジュールから出力された信号は出力決定モジュールによりどちらの信号が出力されるか決定される．このとき，出力決定モジュールにより選択された出力を用いて制御学習モジュール，教示意味学習モジュールともに学習を行う．先に教示の意味を学習させ，教示意味学習モジュールの出力を選択することで教示意味学習モジュールが環境とモジュールの切り換えの対応付けの学習の仲立ちになり，簡単にすることを期待する．以上の手順をまとめたものを図 3 に示す．本研究では手順の右部分の内部情報処理系列と言語教示との対応付けの学習を行うので制御学習モジュールは何もせず，出力決定モジュールは教示意味学習モジュールの出力を常に選択する．

(1) **教示意味学習モジュール**

本研究ではこのモジュールで2種類の教示の意味を学習する．学習には強化学習の手法の一つであるモンテカルロ法[Michie 1968]を用いた．このモジュールは現在の各モジュール結合間の状態と教示を現状態 $S(t)$，その時のモジュール結合制御信号を行動 $a(t)$ として，教示とモジュール組み換えの対応付けを学習するモジュールである．このモジュールが制御する対象は入力部の各モジュールと状態メモリの結合の2つ，方策部の各モジュールと状態メモリ，行動メモリの結合は簡単のため同時に制御されるものとして2つ，行動メモリと行動モジュールの結合の1つで計5つであるが，制約として，各メモリを更新するモジュールは同時に起動しないことにする．これは光認識モジュールと音認識モジュール，光方策モジュールと音方策モジュールの両方を同時に起動させないことになる．これにより，出力される制御信号の種類は 18 種類になり，状態数は与えられる教示が 2 種類なので 36 種類となる．

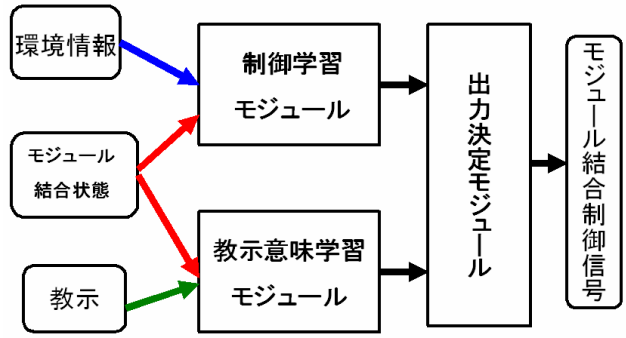

図2 制御部モジュール構成

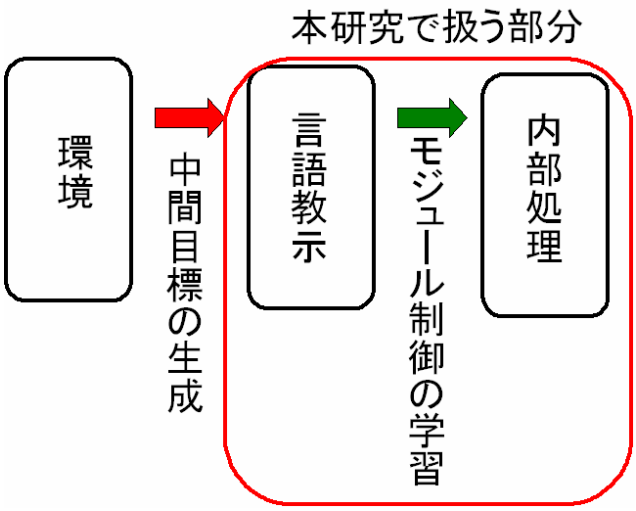

図3 学習手順

3. 実験方法

本研究で用いたシミュレーション空間を図4に示す．

3.1 実験タスク

本研究で行ったタスクを説明する．今回のシミュレーションでは 5×5 マスの平面空間，以下フィールドと呼ぶ，で実験を行った．図 4 のエージェントの位置を初期位置として障害物である音源をよけて，ゴールである光源にたどり着くことが目的である．エージェントは上下左右に 1 マスずつ進むことができる．エージェントは位置に応じて「光を見て動いて」という教示と，「音を聞いて動いて」という教示を与えられながら移動し，障害物のあるマスか，ゴールのあるマスに侵入するまでを 1 試行とする．また，エージェント内部で現在のモジュール結合状態と教示からモジュール結合制御信号を出力する過程を 1 ステップとする．報酬はゴールした時に大きい報酬，障害物にぶつかったときに大きい負の報酬，1 回移動するごとに小さい負の報酬が与えられ，エージェントがフィールド上のいちばん外側の周にいて，かつ，

マスの外に出るような移動を行ったとき，エージェントは移動させず 1 回移動したときと同じ負の報酬を与える．

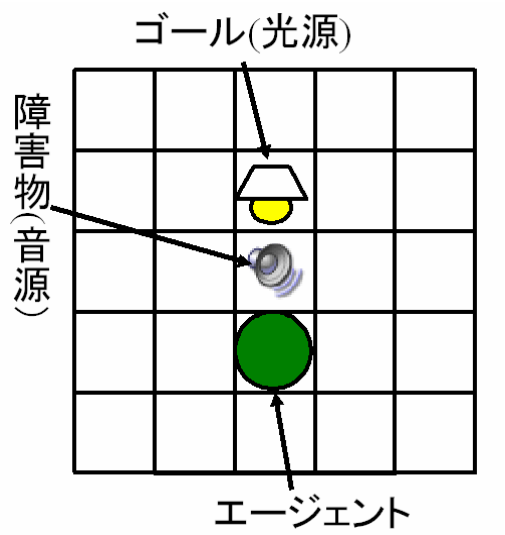

図 4 シミュレーション空間

3.2 与えられる教示

今回の実験で与えられる教示は「光を見て動いて」と「音を聞いて動いて」の 2 種類で，エージェントが障害物に隣接しているときは「音を聞いて動いて」という教示が，それ以外では「光を見て動いて」という教示が与えられる．

3.3 エージェントが行う学習について

今回の実験でエージェントは教示と内部のモジュール結合状態を入力としてモジュール制御信号の出力を学習するので，状態は内部のモジュール結合状態 18 状態と教示が「光を見て動いて」と「音を聞いて動いて」の 2 種類で計 36 状態，そのとき出力される信号は 18 種類なので，36 状態 18 行動の学習を行うことになる．エージェントはフィールド上では 4 回の移動でゴールまで辿りつくが，一回の移動の間に入力部のモジュールを起動し環境情報を取得し状態メモリに記憶，方策部のモジュールにより行動を決定し行動メモリに記憶，行動モジュールにより出力の 3 段階を踏まなければならない．よって見た目よりも大きな空間の学習を行っていることになる．

3.4 エージェントの行動について

各モジュールの仕様について詳述する．

(1) 光方策モジュールについて

光方策モジュールは，エージェントから上に 2 マス，左に 3 マス行ったところにゴールがある場合は左方向，というように 2 方向の光源との距離で遠いほうの行動を選択する．

(2) 音方策モジュールについて

音方策モジュールはエージェントの上方向に障害物がある場合左右どちらかの行動をランダムに選択し，それ以外では上方向に移動する．

(3) 状態メモリについて

状態メモリに記憶されている光源，音源までの距離は 1 試行ごとに初期化される．このときの値はエージェントが光源や音源を認識していないことにあたり、初期化された状態で各方策モジュール により行動を出力すると上方向への移動を出力する．

(4) 行動メモリについて

状態メモリと同様に行動メモリも 1 試行ごとに初期化する．このとき行動モジュールが起動してもエージェントはその場から移動しない．ただし，移動するごとの負報酬は与えられる．

4. 実験結果と考察

4.1 実験結果

本研究では提案手法で教示の意味を学習できることを示すため，前章で説明したタスクを 5000 試行いエージェントに学習させた．そのときの 200 エピソードごとのタスク達成率とタスク成功時の平均ステップ数を図 5，図 6 に示す．またこのとき，それぞれゴールしたときに 10，障害物にぶつかったときに -5，1 回移動するごとに-1 の報酬を与え，行動選択には $\varepsilon = 0.1$ の ε-greedy 法を用いた．また，5000 回の試行中，エージェントがゴールにたどり着いた回数は 4319 回である．

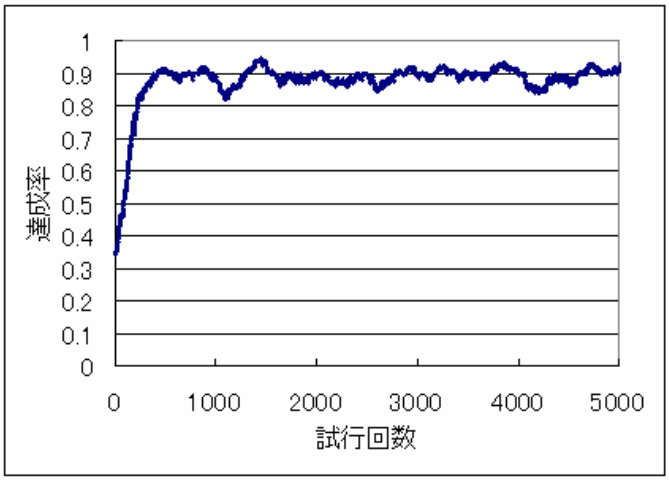

図5 200 試行回数ごとのタスク達成率

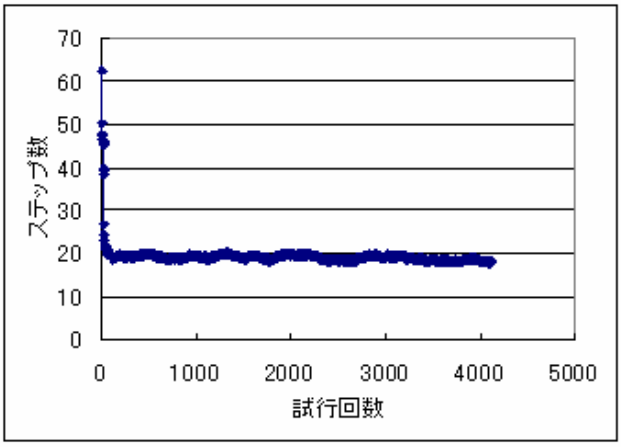

図6 タスク成功時の平均ステップ数

4.2 考察

図 5 よりエージェントは 500 回程度の試行数でタスクを達成することができるようになったことが分かる．また，図 6 の平均ステップ数を見ても比較的早い段階でタスク成功に要するステップ数が減少していっている．このことから，「光を見て動いて」，「音を聞いて動いて」というそれぞれの教示に対してゴールに向かうモジュール組み換え系列と障害物をよけるようなモジュール組み換え系列が対応付けることができたと考えられる．次に，そ

れぞれの教示について対応付けられた組み換えの系列に注目する．今回期待されるモジュール組み換えの系列は，「光を見て動いて」という教示に対しては最初に“光認識モジュールと状態メモリ間結合”を on に，次に“光方策モジュールと各メモリ間の結合”を on に，最後に“行動メモリと行動モジュール間の結合”を on にする，という 3 回の切り替えの系列の獲得を期待し，「音を聞いて動いて」という教示に対しては，“音認識モジュールと状態メモリ間の結合”を on に，“音方策モジュールと各メモリ間の結合”を on に，“行動メモリと行動モジュール間の結合を on にする”という系列の獲得が期待される．そこで，エージェントがゴールするまでにたどった経路と 1 試行にかかった平均ステップ数について考察する．今回のシミュレーション空間では 4 回の移動で初期位置からゴールにたどり着けることができ，その時期待されるステップ数は 12 回であるが，5000 試行学習後の成功時での平均ステップ数は 18.2 回である．また，試行中複数のモジュールが同時に起動することが多かった．移動経路としては初期位置から“右→上→上→上→左→下”という経路をとることが多かった．今回の実験では行動の選択方法に $\varepsilon = 0.1$ の ε-greedy を用いたので一回の試行中に greedy でないモジュール結合制御信号を出力する可能性が高く，探索的な行動をとることによる平均ステップ数の増加の影響を考えなければならない．そこで，5000 試行分学習させたデータを 5 つ用意し，学習後に greedy な行動をとった場合の移動回数とステップ数を表にしたものを表 1 に示す．ただし，このときの移動には初期位置で行動メモリが初期化された時に移動した場合（その場にとどまる）の回数も含まれている．

表1 学習後 greedy な行動時にかかるステップ数と移動回数

	移動回数	ステップ数
1回目	8	19
2回目	7	16
3回目	5	8
4回目	6	11
5回目	7	15
平均	6.6	13.8

表より，エージェントは 1 回の移動に対しておよそ 2.2 ステップ要していることになり，複数のモジュールを同時に起動させ，環境情報の取得，行動の決定，行動の出力の 3 段階のうちの 2 つを 1 ステップにまとめていると考えられる．このように，期待するモジュール組み換え系列と異なる系列が獲得された理由としては，

(1) **報酬の与え方：余分に移動をした場合の負の報酬が小さすぎた．**

(2) **タスク設定：上に移動することが多いため，最初に障害物をよければ失敗する可能性が低くなる**

以上の 2 点が理由として考えられる．

5. おわりに

本研究ではモジュール組み換え型アーキテクチャの持つ，モジュール数の増加による計算量の爆発という問題を扱い，その解決方法として教示を用いる手法の最初の段階である，言語教示とモジュールの組み換え系列という内部情報処理の対応付けを学習させた．結果として予想していたものとは異なるが，教示に対応したモジュール組み換え系列を学習することができたといえる．このように予想と違った系列を学習した原因は，タスク設定や，与えられる報酬という環境の違いによってもたらされたものであると推測できる．今後の課題として，適切なタスク設定や，教示方法の改善などがあげられる．教示については，内部情報，今回だと各結合の状態を表出し，部分的にも報酬を与えてもらうことでより，早く適切な教示と内部情報処理の対応付けが可能だと考えられるが，モジュール数の増加にともないそれぞれのモジュールの持つ機能の把握は困難になるため，複雑な内部状態を教示者にうまく表出する必要がある．また，今回の結果を踏まえて，環境に応じたモジュール切り替え系列の学習を行うことも今後の課題としてあげられる．

参考文献

[小川 2004] 小川 昭利，大森 隆司：“機能部品組み合わせモデルによるナビゲーション行動学習処理の獲得方式の提案”，電子情報通信学会（D-Ⅱ），Vol.J87-D-Ⅱ，No.4，pp.987-998，April 2004.

[Oka 1999] Oka，N:“Apparent“free will” caused by representation of module control”, No matter, Never mind : Proceedings of Toward a Science of Consciosness,: Fundamental Approaches pp.243-249, Tokyo (1999).

[Michie 1968] Michie, D., and Chambers, R. A.: BOXES: An Experiment in Adaptive Control, in E. Dale and D. Michie (eds.), *Machine Intelligence 2*, pp. 137-152, Oliver and Boyd, Edinburgh, 1968.