モジュール組換型アーキテクチャにおける言語教示を用いた内部情報処理の学習

Learning Hidden Information Processing in a Modular Architecture from Instruction

本多　透[1*]　坂本　裕太[1]　尾関　基行[1]　岡　夏樹[1]
Toru Honda[1]　Yuta Sakamoto[1]　Motoyuki Ozeki[1]　Natsuki Oka[1]

[1] 京都工芸繊維大学 大学院工芸科学研究科
[1] Kyoto Institute of Technorogy Graduate School of Science and Technology,

Abstract: Although modular architectures are 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 learing module combination from 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. The system learns felicitous combination of modules using the series of module recombinations. An experiment showed that the system learned module recombinations which correspond to given instructions faster than those which correspond to environment.

1　はじめに

知的ロボットの行動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るための手法が数多く提案されてきた．その中でも，ロボットの要素的な機能をモジュール化し, それらを組み合わせていくことで複雑な問題に対応しようとする研究がある [1][2][3]．それらの多くはモジュールの組み合わせ方が階層型であったり, 競合型であったりと組み合わせに制限が設けられている．しかし，モジュールの自由な組み合わせを許すことで, ロボット自身がより良い内部情報処理方法を学習により発見する可能性がある [4].

そこで本研究では，自由度の高いモジュール組換型アーキテクチャにおけるモジュールの組み合わせを自動的に構成することを目指す. 本手法では，任意のモジュールが，共通のワーキングメモリを介して自由に結合しうるアーキテクチャ(図 1) を採用する．各モジュールとワーキングメモリ間の結合をゲートにより開閉した場合，ゲート数 n に対して組み合わせの数は 2^n となる．モジュールを組み換える回数を p とすると，組み合わせの数は 2^{np} となる．さらに，学習の際の計算量は状態数に比例して増加するため，組み合わせの数は容易に爆発してしまう.

この問題を緩和するため，本研究では，人から与えられる言語教示を利用してモジュールの組み合わせの学習を容易にする方法を提案する．この手法は，様々な状況におけるモジュールの組み換え方を直接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する．その対応付けを利用してさまざまな状況におけるモジュールの組み換え方を学習することで，問題の解決を容易にするという試みである.

本稿では，まず提案手法について述べ，環境から直接モジュールの組み合わせを学習した場合と，中間目標となる教示からモジュールの組み合わせを学習した場合とでの学習速度の差をシミュレーションによって評価する．最後に，教示を用いたモジュール組換型アーキテクチャの今後の展望を述べる.

2　モジュール組換型アーキテクチャ

モジュール組換型アーキテクチャ[5] の概要を図 1 に示す．A〜E はモジュールを表し, モジュール同士はワーキングメモリを介して自由な組み合わせで接続することができる．制御部から送られるモジュール制御信号により，各モジュール間の結合を開閉する．制御信号はモジュール結合数分の長さを持ったビット信号で表

*連絡先：京都工芸繊維大学　大学院工芸科学研究科
京都市左京区松ヶ崎橋上町 1 番地
E-mail: m9622032@edu.kit.ac.jp

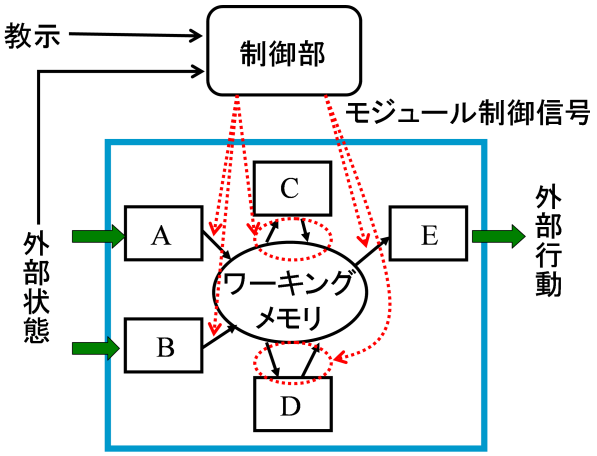

図 1: モジュール組換型アーキテクチャ

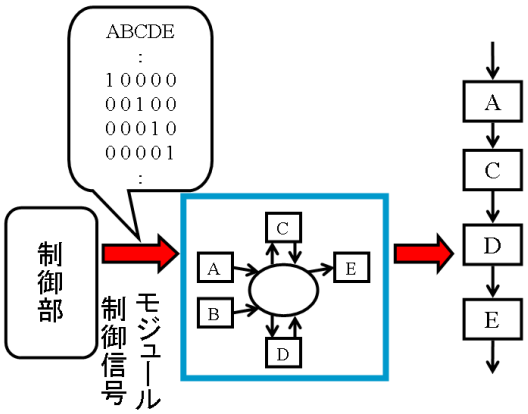

図 2: モジュール制御信号によるモジュールの組み換え

される．図 2 のようにいくつかの信号の系列により内部で処理が行われ，その結果として外部行動が出力される仕組みとなっている．このようにモジュール結合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．

制御部では，環境（外部状態）と抽象的な教示，モジュールの結合状態（内部状態）から適切なモジュール制御信号を出力するように学習を行う．制御部自体も複数のモジュールから構成されている．

2.1 制御部の構成と学習方法

制御部の構成と，学習を行うための仕組みを説明する．制御部の構成を図 3 に示す．制御部がモジュールを組み換えていくアルゴリズムは以下のとおりである．

1. 教示学習モジュールからは与えられた教示と内部状態に応じて，自律学習モジュールからは外部状態と内部状態に応じて，それぞれモジュール制御信号 a_I，a_A が出力される．

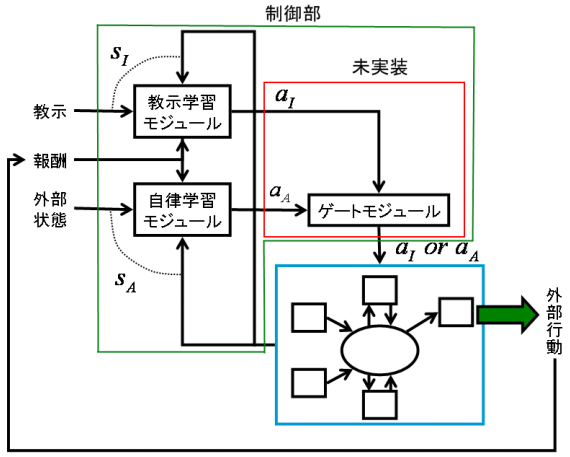

図 3: 制御部の構成

2. ゲートモジュールは，教示が与えられたなら a_I を，そうでなければ a_A を出力し，その信号に従ってモジュール結合状態が変更される．

3. 外部行動の結果として得られた報酬を教示学習，自律学習モジュールに与える．

4. 変更後のモジュール結合状態を新たな内部状態として両モジュールに入力し，1 の処理に戻る．

制御部の中では，教示学習モジュールと自律学習モジュールの 2 つのモジュールで強化学習が行われる．教示学習モジュールでは現在のモジュール結合状態と与えられた教示のセットが強化学習における状態相当しに，モジュール制御信号が強化学習における行動に相当する．自律学習モジュールでは外部状態と現在のモジュール結合状態のセットが状態に相当し，(教示学習モジュールと同じく) モジュール制御信号が行動に相当する．報酬は，外部行動の結果により，2 つのモジュールに同じものが与えられる．

教示学習モジュールは自律学習モジュールに比べて，学習する状態数が少なく学習が速く進むので，教示学習モジュールから出力される行動を用いて自律学習モジュールの学習を早く進めるのが本研究の狙いである．このとき，行動価値の更新は以下のように行われる．

$$
\begin{aligned}
if \quad & \text{ゲートモジュールが } a_I \text{を出力} \\
& \delta_A \leftarrow r + \gamma Q_A(s'_{AI}, a'_{AI}) - Q_A(s_A, a_I) \\
& Q_A(s_A, a_I) \leftarrow Q_A(s_A, a_I) + \alpha \delta_A e(s_A, a_I) \\
& \delta_I \leftarrow r + \gamma Q_I(s'_{II}, a'_{II}) - Q_I(s_I, a_I)
\end{aligned}
$$

$$
\begin{aligned}
& Q_I(s_I, a_I) \leftarrow Q_I(s_I, a_I) + \alpha\delta_I e(s_I, a_I) \\
else \quad & (\text{ゲートモジュールが } a_A \text{を出力}) \\
& \delta_A \leftarrow r + \gamma Q_A(s'_{AA}, a'_{AA}) - Q_A(s_A, a_A) \\
& Q_A(s_A, a_A) \leftarrow Q_A(s_A, a_A) + \alpha\delta e(s_A, a_A) \\
& \delta_I \leftarrow r + \gamma Q_I(s'_{IA}, a'_{IA}) - Q_I(s_I, a_A) \\
& Q_I(s_I, a_A) \leftarrow Q_I(s_I, a_A) + \alpha\delta_I e(s_I, a_A)
\end{aligned}
$$

ここで，s_A, a_A は自律学習モジュールの状態と行動を表し，s_I, a_I は教示学習モジュールの状態と行動を表す．s'_{AI}, a'_{AI} は s_A で a_I をとったときの次状態とそのとき取る行動を表す．同様に，s'_{II}, a'_{II} は s_I で a_I をとったときの次状態とそのとき取る行動を，s'_{AA}, a'_{AA} は s_A で a_A をとったときの次状態とそのとき取る行動を，s'_{IA}, a'_{IA} は s_I で a_A をとったときの次状態とそのとき取る行動を表す．δ_A，δ_I は行動価値の更新分を表し，r は報酬を，γ は割引率，α は学習率，$e(s, a)$ は適格度トレースを表す．Q_A は自律学習モジュールの行動価値関数を，Q_I は教示学習モジュールの行動価値関数を表す．

それぞれの行動価値は状態 s_A，s_I でゲートモジュールにより選択された方の行動の　価値を更新するので，教示学習モジュールが自律学習モジュールに比べ学習が早く進む場合，ゲートモジュールで a_I が選択することで，教示学習モジュールの学習結果を利用できるため，自律学習モジュールの学習も早く進むようになる．この更新方法は，入替更新トレースによる Sarsa(λ) に手を加えたものである．入替更新トレースによる Sarsa(λ)[6] を採用したのは，1 ステップ TD 誤差によって学習を行う Q-Learning などでは繰り返し同じ系列を経験する必要があるため学習に時間がかかることと，組み換えの系列はループが起こりやすいため，累積トレースより入替更新トレースの方が適しているためである．

3　評価実験

モジュールの組み換えを外部状態から学習する場合と，教示から学習する場合の学習性能を比較する実験を以下のとおりに行った．

3.1　実験タスク

評価実験で使用するシミュレーションフィールドを図 4 に示す．モジュール組換型アーキテクチャを実装したエージェントが，教示を与えられながら初期位置から障害物をよけつつゴールを目指す．フィールドは 5 × 8 の平面で，○印はエージェントの初期位置を表し，スピーカーは障害物で警告音を発しているという設定である．また，電球はゴールを表し，光によって

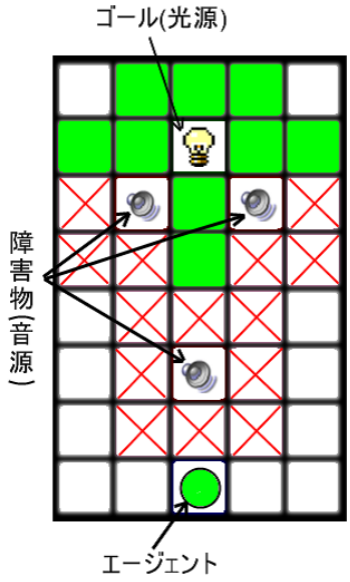

図 4: シミュレーションフィールド

エージェントはゴールを認識する．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．与えられる教示の種類は図中の灰色のマスでは “光に注目” という教示が，バツ印のマスでは “音に注意” という教示が与えられる．エージェント内部でモジュールを組み換える過程を 1 ステップとし，初期位置から障害物のあるマスに侵入するかゴールするまでを 1 エピソードとする．

3.2　実験で使用した各モジュールの仕様

本実験に用いたモジュールの構成を図 5 に示す．各モジュールは，外部状態を取得する認識モジュール，入力に応じてエージェントの取る外部行動を決定する方策モジュール，外部行動を出力する行動モジュール，モジュール間でやり取りされる情報を記憶するワーキングメモリに分類される．各モジュールの仕様は以下のとおりである．

3.2.1　光認識モジュール

エージェントの位置を基準にして光源のある方向 (上下左右) を認識するモジュールである．光源のある方向のセンサー値は “1” になり，ない方向のセンサー値は “0” になる．ただし，図 4 のフィールド上で，マンハッタン距離で 3 以上離れた位置では光源からの光が届かないものとする．光認識モジュールとワーキングメモリ間のゲートが開くことで現在のセンサー値がワーキ

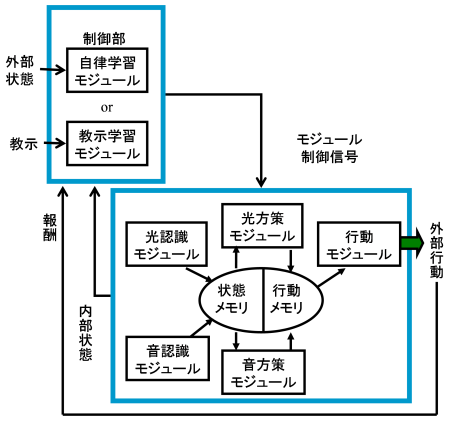

図 5: 実験に用いたモジュール構成

ングメモリに記憶される．例えば，エージェントの右上方向に光源がある場合，上方向と右方向のセンサー値が 1 で，下方向と左方向のセンサー値は 0 となり，距離が 3 以上離れた場所にある場合はセンサー値はすべて 0 になる．

3.2.2 音認識モジュール

光認識モジュールと同じく，音源のある方向を認識するモジュールである．モジュールが起動することで，現在のセンサー値がワーキングメモリに記憶される．音源のある方向のセンサー値が “1” になり，ない方向のセンサー値は “0” になるが，エージェントの周囲 8 近傍より離れた位置にある音源は認識できないものとする．

3.2.3 光方策モジュール

ワーキングメモリにある光源の位置情報を読み取りエージェントが外部に出力する行動を出力するモジュールである．出力される行動は上下左右のどちらかへ 1 マス移動するという 4 行動で，光源のある方向へ移動する行動が出力される．光源が右上，左上，右下，左下にある場合は 2 方向のどちらかがランダムで出力される．また，光源が認識できない場合，4 方向への行動がランダムに出力される．

3.2.4 音方策モジュール

ワーキングメモリにある音源の位置情報を読み取り，エージェントが外部に出力する行動を出力するモジュールである．出力される行動の種類は光方策モジュールと同じく上下左右のどちらかに 1 マス移動するという行動であり，音源のある方向以外へ移動する行動がランダムに出力される．

3.2.5 行動モジュール

ワーキングメモリにある情報を読み取ってエージェントを動かすモジュールである．

3.2.6 ワーキングメモリ

ワーキングメモリでは，入力部により認識された値は状態メモリに，方策部により出力された行動は行動メモリに記憶される．

3.3 制御部

制御部の中身は，外部状態からモジュールの組み換えを学習する場合は自律学習モジュールを用い，教示からモジュールの組み換えを学習する場合は教示学習モジュールを用いた．自律学習モジュールと教示学習モジュールは同じ報酬をもらい学習を行う．報酬は，ゴールにつくと成功として 10，障害物のあるマスに侵入すると失敗として -100，エージェントがフィールド上を移動するごとに -1 与えられるようにした．学習に用いたパラメータは両モジュールとも学習率 $\alpha = 0.1$，割引率 $\gamma = 0.9$，$\lambda = 0.9$ とし，行動の選択には $\epsilon = 0.1$ の ϵ-greedy を用いた．

3.3.1 自律学習モジュール

外部状態は，光と音，各 4 方向のセンサー値であり状態数は 2^8 である．内部状態数はモジュールの組み合わせ数であるが，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．つまり，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．この制約により，組み合わせの種類は 18 種類になる．モジュール制御信号の種類は組み合わせの数と同じである．よって，自律学習モジュールはでは状態数が $2^8 \times 18$，行動数が 18 の学習を行うことになる．

3.3.2 教示学習モジュール

教示は “光に注目” と “音に注意” の 2 つで，教示なしの状態も含めて 3 状態である．内部状態数とモジュール制御信号数は自律学習モジュールと同じである．よって教示学習モジュールでは，状態数が 3×18，行動数が 18 の学習を行うことになり，自律学習モジュールより学習が早く進む．

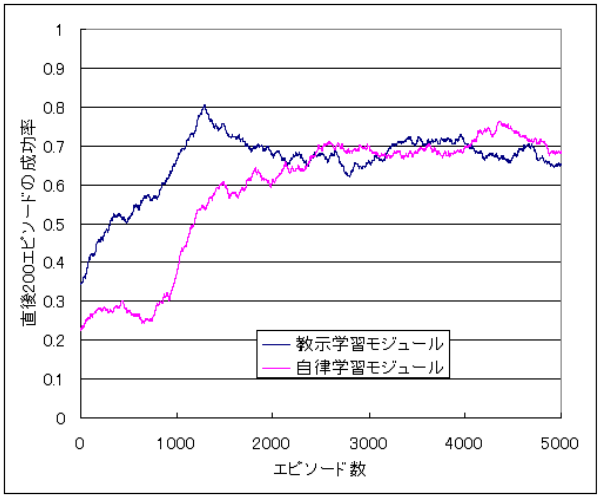

図 6: 自律学習モジュールと教示学習モジュールのタスク成功率

4 実験結果と考察

教示学習モジュールと自律学習モジュール単体で 5000 エピソードずつ学習させた場合の成功率の推移を図 6 に示す．グラフの横軸は経過エピソード数で，縦軸はその時点でのエピソード数から 200 エピソード分のタスク成功回数を 200 で割った値である．3000 エピソードほどで，両モジュールとも学習が収束している．学習初期では教示学習モジュールの方が，自律学習モジュールより，成功率が高くなっているが，これは，2 つの学習モジュールの状態数の差によるものだと考えられる．この結果から，教示学習モジュールを，自律学習モジュールの学習に利用することで，自律学習モジュールの学習速度を速くできる可能性があることが分かる．

5 まとめ

本研究では，ワーキングメモリを介して任意のモジュールの組み合わせが可能なアーキテクチャにおいて，教示を用いて学習を速く進める方法を提案し，外部状態から学習する場合より，教示から学習する場合のほうが速く学習できることを示した．現在，未実装部分 (ゲートモジュール) を含めた全体システムの実装を進めており，実装完了後，提案システム全体の動作評価を行う予定である．今後取り組むべき課題として，次のものがあげられる．

- 現在，エージェントが取る外部行動を決定する方策モジュールは作り込むだものを用いているが，本来方策モジュール自体も学習によって獲得されるべきである．そこで，方策モジュールの学習と，モジュールの組み換えの学習を同時に行う方法を考案する必要がある．
- このシステムで与えられる教示は抽象的な中間目標となりうるものである．よって，あるタスクで学習させた教示学習モジュールを，類似したタスクでも利用できる可能性がある．そこで，教示学習モジュールの学習結果がどの程度他のタスクで利用できるかの検討を行う．

謝辞

本研究は科研費 (17500093 および 21500137) の助成を受けたものである．

参考文献

[1] 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”, 電子情報通信学会論文誌, Vol.J87-D-II, No.4, pp.987-998 (2004)

[2] 鮫島和行, 銅谷賢治, 川人光男: “強化学習 mosaic: 予測性によるシンボル化と見まね学習”，日本ロボット学会誌, Vol. 19, No. 5, pp. 551-556, (2001)

[3] Jacobs, R., Jordan, M. I., Nowlan. S. J. and Hinton, G. E.: “Adaptive mixtures of local experts”, (1991) *Neural Computation*, vol.3, pp.79-87

[4] 本多 透，板舛 尚樹，岡 夏樹: “ロボットの内部情報処理に対する言語教示可能性”，人工知能学会全国大会，OS7-5 (2009)

[5] 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)

[6] Richard S.Sutton, Andrew G.Barto：“Reinforcement Learning”，三上 貞芳，皆川 雅章 共訳, pp.192-194 (2000)