

# 複雑な問題におけるエージェントの行動プログラムの自動獲得

秋本紗矢香†

長尾智晴†

†横浜国立大学 大学院環境情報学府

## 1 まえがき

掃除ロボットやサービスロボットといった自律エージェントの行動プログラムを、あらゆる状態に対して人手で設計することは困難である。そこで、設計の負担を軽減するため、進化計算を応用して自律エージェントの行動プログラムを自動で生成する研究が行われている。その中でも、実世界における自律エージェントが行うタスクは障害物回避や電源管理など、多数のタスクを含むことが多い。そこで、複雑なタスクをサブタスクに分割し、サブタスクごとにエージェントのコントローラ(サブコントローラ)を生成することで複雑なタスクを達成するという手法が提案されている [1]。しかし、有効なサブコントローラの組み合わせの決定や、複数のサブコントローラを生成することは人手では困難である。

本稿では、複雑な問題においてモジュール化を利用した自律エージェントの行動プログラム自動生成手法の提案を行う。提案手法では、複雑なタスクをサブタスクに分割し、サブタスクごとに異なる指標を用いてモジュールを抽出する。抽出されたモジュールをサブコントローラとし、各モジュールをメインプログラムで組み合わせることで機能分化するプログラムの自動獲得を目指す。モジュール化の方法については、モジュール化を利用した自律エージェントの行動プログラムの自動構築手法 [2] をベースにする。提案手法をシミュレーション上で部屋の掃除問題に適用し、性能の評価を行った。

## 2 関連研究

提案手法のベースとなるモジュール化を利用した自律エージェントの行動プログラムの自動構築手法 [2] は、グラフ構造状プログラムを進化的に獲得する手法である。グラフ構造の例を図 1 に示す。メインプログラムは、開始ノードから遷移が始まり、エージェントの知覚情報に応じた判定ノードで分岐する。そして、到達した出力ノードに応じた行動が実行される。モジュールノードに到達した場合は、対応するモジュールの開始ノードに遷移が移行する。モジュール内でリリースノードに到達すると、メインプログラムに処理が戻る。

モジュールの生成は、ノード間のエッジに設定されている信頼度を用いて部分構造の評価に基づいて行われる。信頼度は、エージェントが行動した際に環境から得られる報酬を基に更新する。報酬とは、エージェン

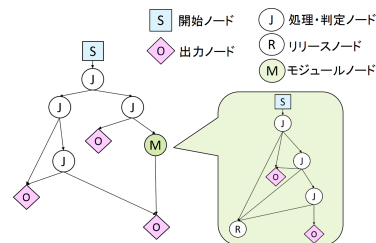


図 1: グラフ構造の例

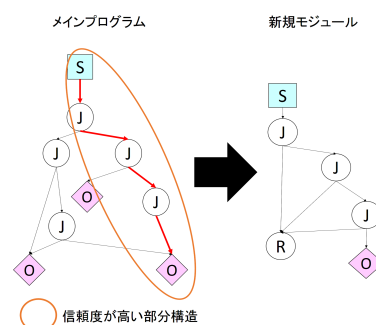


図 2: モジュール化の例

トに獲得させたい動作にあわせて設計者があらかじめ設定する値である。エージェントの行動選択時に、遷移したエッジをメモリに保存する。報酬を獲得した場合に、メモリに保存されているエッジの信頼度に対して信頼度の更新を行う。タスク終了後の信頼度が高いエッジをもつ部分構造を抽出し、1つのモジュールとする。モジュール化の例を図 2 に示す。

## 3 複雑な問題におけるエージェントの行動プログラムの自動獲得手法の提案

本研究では、手法 [2] をベースとして、複雑なタスクに対して機能分化したコントローラを自動的に生成する手法の提案を行う。

提案手法では、タスクをサブタスクに分割し、サブタスクごとに報酬を設定する。そしてサブタスクごとの報酬  $R_i$  をもとにサブタスクごとにモジュールを生成する。 $i$  は、サブタスクに対応する番号である。信頼度の更新式を式 (1) に示す。

$$Q'_{ijk} = Q_{ijk} + \alpha\beta^{n_{jk}}(R_i - Q_{ijk}) \quad (1)$$

式 (1) は直近の行動ほど報酬の獲得に貢献したと考える利益共有法の考えに基づいており、過去に遷移したエッジほど、信頼度の更新は小さくなる。ここで、 $j$  はノード番号、 $k$  はエッジ番号である。 $\alpha$  は学習率 ( $0 \leq \alpha \leq 1$ ) を、 $\beta$  は忘却率 ( $0 \leq \beta \leq 1$ ) を表す。また、 $n_{jk}$  は信頼度をもつエッジが報酬を得た時点から数えて何回前の行動選択時に遷移したかを示している。タスク終了後、

Automatic Acquisition of Agent Control Program for Complex Task  
 †Sayaka Akimoto †Tomoharu Nagao  
 †Graduate School of Environment and Information Sciences, Yokohama National University

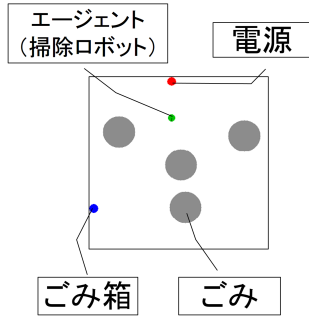


図 3: 実験環境

サブタスクごとに信頼度が大きいエッジをもつ部分構造をモジュールとする。モジュールは、サブタスクごとに全ての個体から参照可能なライブラリに保存する。ライブラリに保存されたモジュールは突然変異によってメインプログラムに組み込まれる。

#### 4 提案手法の性能評価実験

シミュレーション環境におけるエージェントによる部屋の掃除問題を用いて性能評価実験を行った。エージェントは掃除ロボットを想定し、有限の電池と吸い込んだごみをためるごみタンクを備えている。図3に実験環境、図4にエージェントのセンサの設定を示す。実験環境には電源とごみ箱があり、電源上に乗ると電池が補充され、ごみ箱上に乗るとタンクのごみを捨てられる。エージェントは知覚情報として、6つの赤外線センサ、前方に2つの接触センサ、ごみを吸ったかどうかかわかるごみセンサ、電源・ごみ箱からの距離と方向がある。

提案手法では、掃除問題のサブタスクを”掃除”，”電源管理”，”ごみ管理”に分割し、それぞれのサブタスクに対して報酬  $R_i (i = 1, 2, 3)$  を設定した。  $R_1$  は掃除の報酬，  $R_2$  は電源管理の報酬，  $R_3$  はごみ管理の報酬である。報酬で使用する更新式を式(2)に示す。

$$r^i = r^{i-1} + \gamma(C - r^{i-1}) \quad (2)$$

ここで  $C$  は定数である。また、本稿では  $\gamma$  は 0.4 に設定した。報酬の詳細を次に示す。

- $R_1 = r_{dust}$ 
  - 初期値  $r_{dust} = 0.01$
  - ごみを吸い込んだ場合  $r_{dust}, C = 2.0$
  - ごみを吸い込まなかった場合  $r_{dust}, C = 0.01$
- $R_2 = E \times r_{eng1} + r_{eng2}$ 
  - 初期値  $r_{eng1} = 0.01$
  - 電源の方向に近づいた場合  $r_{eng1}, C = 1.0$
  - 電源から遠ざかった場合  $r_{eng1}, C = 0.01$
  - 電池を補充した場合  $r_{eng2} = E$
- $R_3 = D \times r_{tank1} + r_{tank2}$ 
  - 初期値  $r_{tank1} = 0.01$
  - ごみ箱の方向に近づいた場合  $r_{tank1}, C = 1.0$

図 4: エージェントのセンサの設定

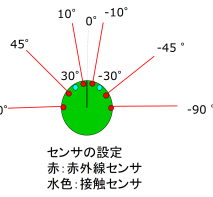


表 1: 実験結果

	提案手法	比較手法 1	比較手法 2
学習環境	62.93	34.26	37.40

- ごみ箱から遠ざかった場合  $r_{tank1}, C = 0.01$
- ごみを捨てた場合  $r_{tank2} = D$

$E$  は式(3)，  $D$  は式(4)で表される。

$$E = 1 - \frac{\text{残り電力量}}{\text{最大電力量}} \quad (3)$$

$$D = \frac{\text{現在のごみタンク容量}}{\text{最大ごみタンク容量}} \quad (4)$$

学習では、電源の位置(開始位置)、ごみ箱の位置、ごみをランダムに配置した3つの環境を用いた。ただし、電源の位置とごみ箱の位置は壁際にあるものとする。適応度  $F_{total}$  は式(5)で表され、各環境の適応度  $F$  は式(6)で表される。

$$F_{total} = \frac{1}{3} \times \sum_{k=1}^3 F_i \quad (5)$$

$$F = 100 \times \frac{\text{掃除した面積}}{\text{ごみの面積}} + \frac{\text{掃除した面積}}{\text{消費電力量}} \quad (6)$$

比較手法1としてモジュール化を利用したエージェントの行動プログラムの生成手法[2]を用い、比較手法2はモジュール化を使用しない手法[2]とする。

#### 5 実験結果

実験結果を表1に示す。学習環境の結果は20試行平均の適応度である。提案手法が比較手法に比べて良い結果を示した。比較手法では電源管理やごみ管理を行動を獲得する個体が少なかったが、提案手法では電源管理やごみ管理をする行動が獲得できた。これは提案手法で各サブタスクごとにモジュールを生成しているため、電源へ向かう行動やごみ箱へ向かう行動が比較手法に比べて獲得しやすくなったためだと考えられる。

#### 6 まとめ

本稿では、複雑な問題においてモジュール化を利用した自律エージェントの行動プログラム自動生成手法の提案を行った。提案手法を部屋の掃除問題に適用することで性能の評価を行った。比較手法に比べて、提案手法が良い結果を示した。今後は、パラメータの調整や他の問題への適用を検討していく。

#### 参考文献

[1] M. Duarte, S. Oliveira, and A. L. Christensen, "Hierarchical Evolution of Robotic Controllers for Complex Task", Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on, 2012.

[2] 近藤義隆, 長尾智晴, "エージェント制御プログラムの自動構築手法に関する研究", 計測自動制御学会 第40回 知能システムシンポジウム, 2013