

8K-07 遺伝的プログラミングを用いたロボット協調行動の学習

柳井 孝介 伊庭 斉志

東京大学工学部電子情報工学科

1. はじめに

自律したロボットの実現には様々な問題があるが、本研究ではその中でも「最適な動作を考える」という問題に焦点をあてた。つまり、問題とするのは「最適なロボット行動プログラムの自動生成」である。ロボット行動プログラムの自動生成の研究は、「様々な環境への適応した行動プログラム作り」「プログラムの負担の減少」という点で有用であると言えるだろう。現在、ロボットの行動プログラムは人の手で書かれるのがほとんどであるが、様々な事態（環境）に対応できるプログラムの作成は、人の手では限界がある。本研究では、GP をロボットの行動学習に適用し、ロボットの合理的な行動プログラムの獲得を目的としている。ロボットが遂行するタスクとしては、より人の手によるコーディングが困難と考えられる協調行動を扱っている。また、最終的に実機ロボットで動かすことを念頭に研究を行っている。

2. ロボット学習という観点から見た GP

GA (遺伝的アルゴリズム、Genetic Algorithms) は、進化論的な考え方に基づいてデータを操作し、最適解探索や学習、推論を行う手法である。GA では、解を操作ができるようにコード化（コード化したものを自然界に倣って遺伝子と呼ぶことにする）し、最適解に近いもの（自然界に倣って、適合度が高いもの）は遺伝子の一部または全てを、次世代に残すようにしている。GP は GA の遺伝子型を構造的な表現が扱えるように拡張したもので、オペレータは木構造を扱うことになる。つまり、プログラムの合成、変更は、S 式の部分木を交換したり、変更したりすることによって実

現する。これにより、選択、生成を繰り返しより優れたプログラムが探索できる。GP の適用により、大規模なロボット行動プログラムの自動生成が可能になると期待されるが、1つ難関がある。数百の個体を数百世代にわたり実行させ行動プログラムを評価する必要があり、多大な実験コストを要するのである。よって、実機ロボット上での学習は困難なものになる。しかし、当然シミュレータの世界と実際の世界ではロボットも環境も異なるので、シミュレータで学習した結果がそのまま実ロボットに使えるとは限らない。GP を用いる利点は、次のようなものが挙げられる。

- ・人の介入度が調整でき、比較的小さい。
- ・局所解に陥りにくい。
- ・学習は適合度を与えられるだけで済み、
細かな「教育」をする必要がない。
- ・学習結果が visible である。

3. 実験

本稿では、GP を用いて脱出問題を解いた例を紹介する。脱出問題とは、ここでは「3 台のロボットが協力して、3つのボタンを押し、出口から脱出する」というものである。3 つボタンを押すと出口が開くようになっている。この行動プログラムは、「火災時に貴重なものだけを回収して出口へ向かう」ロボットに応用できる。3 台のロボットは、それぞれ移動速度が異なり、A、B、C の順に速く移動することができる。また、それぞれカメラを持っており、決められた情報を得ることができる。制限時間内に出来るだけ速くボタンを押し、出口から脱出するようなプログラムの獲得を目的とした。

A Robot Learning of Cooperative Behavior Using Genetic Programming

Kosuke Yanai

Department of Information and Communication Engineering School of Engineering, University of Tokyo

7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, JAPAN

この実験で用いた GP システムは、遺伝子長制限式 LinearGP (ルーレット戦略) である。1チームの3台のロボットに、それぞれ独立にこのシステムを用いた。また、実験環境 (シミュレータ) には、Webots (Cyberbotics 社) を用いた。適合度には以下の計算式を用いた。

適合度 = $a + b$
 + ボタンを3つ押し終わった時の残り時間
 + 3台とも脱出した時の残り時間
 a : ボタンを3つ押すと +20
 1台脱出するごとに +20
 b : 制限時間終了時の出口と各ロボットの距離に応じた値 (出口に近いほど値は大きく、脱出しているとき Max)

個体数 1024、世代数 50、交叉7割、突然変異3割、遺伝子長 200 ノードでの実験結果を図1~6に示す。これは、50世代目の最も優秀な個体 (チーム) の行動の様子である。

4. 結果の解析と考察

実験結果を見るとおもしろいことがわかる。step 3 (図3) では、すでにロボット A は出口に到着し、出口が開くのを待っている。またロボット C は1つ目のスイッチを押したところである。step 4 (図4) ではロボット B が2つ目のスイッチを押し、ロボット C は3つ目のスイッチに向かっている。step 5 (図5) ではロボット C がついに3つ目のスイッチを押し、出口で待っていたロボット A が脱出できている。step 6 (図6) では、ロボット B も出口に向かって動き出し脱出するに至っているが、ロボット C は立ち往生している。

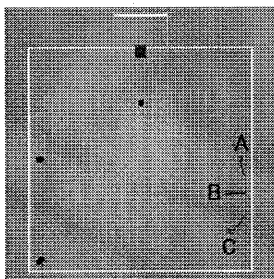


図1 step1

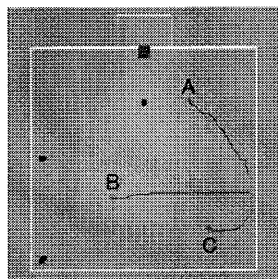


図2 step2

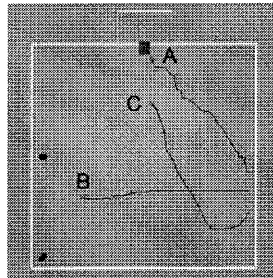


図3 step3

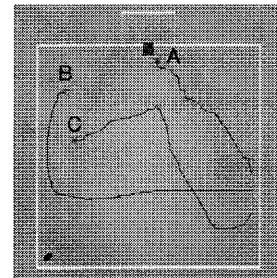


図4 step4

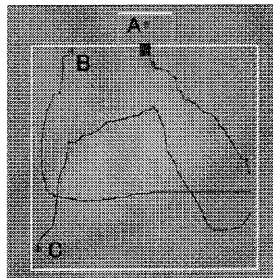


図5 step5

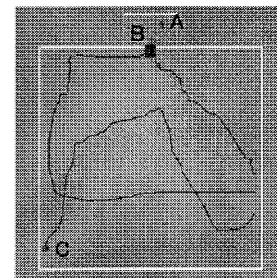


図6 step6

まとめると、まず最も移動速度の遅いロボット A は一目散に出口を目指して移動している。最も移動速度の速いロボット C は多くスイッチを押しているが、ロボット A とロボット B の犠牲になって、結局脱出できていない。また、ロボット B は、スイッチがすべて押されるまでランダムに動いき、出口が開いた時点で出口に向かって移動している。

現状のノードでは、非常に賢い行動プログラムが得られたと言える。ノードを変更したり、増やしたりすることにより、より優秀な行動プログラムを得ることができると思われる。

5. おわりに

今後は、学習結果が実機で動かせるようにしていきたいと考えている。一番の問題は、実機カメラを用いた際のノイズに耐え得る頑強なプログラムを生成しなければならないということである。また、その他のタスクの学習も行っていきたいと考えている。

参考文献

- [伊庭 96] 伊庭齊志 遺伝的プログラミング 東京電機大学出版局
 [寺尾 00] 寺尾誠、石塚満、伊庭齊志 遺伝的プログラミングを用いた協調行動の創発 卒業論文