実機ロボットを用いた強化学習による行動獲得に関する研究

三橋 秀行† 神尾 正太郎‡ 伊庭 斉志‡
†東京大学工学部電気工学科
‡東京大学大学院 新領域創成科学研究科

1 はじめに

ロボットの行動を作りこみではなく自動的に生成する手法として遺伝的プログラミングや強化学習といった学習手法がある。しかしこれらの手法でも実ロボットだけで学習するには時間がかかり過ぎることが多く、シミュレータで学習したものを実ロボットに適応することが一般に行われている。その学習の際には正確なシミュレータが必要となるが、シミュレータが作成困難なタスクも多い。

シミュレータではなく実ロボットで学習することで、シミュレータでは再現されないハードウェアや環境の特性をも学習し、思いがけない行動を獲得できる可能性が生まれる。

そこで、本論文では実機を用いた、遺伝的プログラミングと強化学習を統合した、実ロボットのでの実時間学習が可能な手法のについて提案し、それを用いて実験を行った。

2 遺伝的プログラミングと強化学習の統合

2.1 先行研究

遺伝的プログラミングと強化学習を統合した研究はこれまでにいくつか行われている [1, 2]。しかしこれらで用いられている手法は集団学習であり、個体学習である強化学習が GP ループの中に入っているために集団中の多数の個体に対して強化学習を行なう必要がある。これをそのまま実ロボットに適用すると学習には膨大な時間が必要とされる。

2.2 提案手法

本研究では、個体学習に重点を置いた遺伝的プログラミングと強化学習の統合手法を提案する。個体学習である強化学習が GP ループの外側にあるため実ロボットに応用した場合には学習時間が従来の統合手法よりもはるかに短くて済む。

本手法は 2 段階から成っている。

1. 簡略化されたシミュレータ上で遺伝的プログラミングを行ない、タスクに必要な行動の判断基準をもつプログラムを作り上げる。
2. 1. により得られたプログラムを実ロボットに載せ個体学習 (強化学習) を行う。

第 1 段階で得られた判断基準により状態空間が部分空間に分割されるので第 2 段階で学習が高速化されることが期待できる。また、第 1 段階で学習しているために、ロボットは第 2 段階の開始時から目的に応じた行動をとることが期待できる。

3 問題設定

実験で用いる実ロボットとして、SONY の AIBO ERS-220 を利用した。

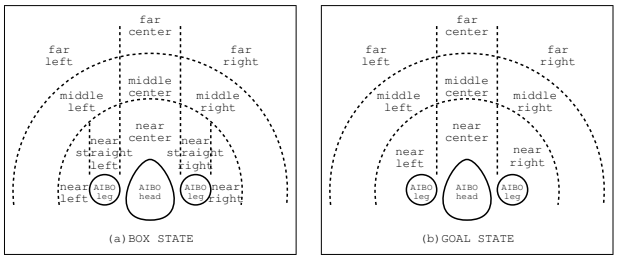

図 1: 状態の分割

本実験で設定したタスクは、荷物に見立てた箱をゴールまで押して運ぶというものである。この問題の難しい点はロボットが前進する場合であっても、箱と足の位置関係によっては箱が前に進んだり左右にそれるという異なった振る舞いとなることである。

状態空間は CCD から得られる画像を使い、箱とゴールが見える位置を基に構成した。箱とゴールの状態を地面上に投影するとそれぞれ、図 1(a),1(b) のようになる。このほかに箱が視界にない状態を `lost`、箱が視界にはないが 1 ステップ前に `left` にあった場合を `lost_left`、`right` にあった場合を `lost_right` とした。箱について 14 状態、ゴールについて 12 状態あるので、全体の状態数は $14 \times 12 = 168$ である。

ロボットが選択できる行動としては前進、後退、左回転、右回転、後退 + 左回転、後退 + 右回転の 6 種類を用意した。

4 提案手法による実験

4.1 遺伝的プログラミングによる学習

4.1.1 シミュレータ

第一段階としてコンピュータ上でシミュレータにより学習を行った。タスクに必要な情報でシミュレータで表現すべき情報は分割の仕方であるため、シミュレータには一切の実ロボットの物理パラメータなどは考慮に含まない簡単なものである。実ロボットで用いる状態空間を簡略化したものとして、図 1 から距離情報を除いた `left,center,right,lost,lost_left,lost_right` の 7 状態に分割した。

4.1.2 遺伝的プログラミングの設定

本研究で遺伝的プログラミングに使用したノードは以下のようなものである。

- 終端ノード:
  `move_forward, turn_left, turn_right`
- 関数ノード:
  `box_where, goal_where, if_box_ahead, prog2`

終端ノードは前進、左回転、右回転行動に対応している。関数ノードの `box_where, goal_where` は 6 引数の関数で、ロボットから見た、箱、 ゴールの状態に応じて引数の 1 つを実行するものである。`if_box_ahead` は条件判断で、箱がロボッ

Behavior Acquisition for Real Robots by means of Reinforcement Learning
† Hideyuki Mitsuhashi,
Department of Electoric Engineering, The University of Tokyo
‡ Shotaro Kamio, Hitoshi Iba,
Graduate School of Frontier Sciences, The University of Tokyo

トの正面手前にある時に第 1 引数を実行し、箱が正面手前にない場合に第 2 引数を実行する。prog2 は第 1 引数、第 2 引数を連続して実行する関数である。

初期位置はランダムでゴールするか一定回数の行動を取ると 1 試行が終了とした。適合度は以下のように決定する。

- タスクを達成した場合 $f_{goal} = 100$
- タスクを達成した場合には、$f_{left_steps} = 4 \times \frac{\text{残り歩数}}{\text{最大歩数}}$
- 少なくとも 1 回、箱を移動させた場合 $f_{move} = 10$
- 少なくとも 1 回、箱の方向を向いた場合 $f_{see_box} = 1$
- 少なくとも 1 回、ゴールの方向を向いた場合 $f_{see_goal} = 1$
- $f_{lost} = \frac{\text{(箱を見失った回数)}+\text{(ゴールを見失った回数)}}{\text{(回転した回数)}}$

以上の和が i 回目の試行に対する適合度 $fitness_i$ となる。

個体の適合度計算では 100 試行での平均をとっている。個体の適合度は以下の式で計算される。

$$fitness = \frac{1}{100}\sum_{i=0}^{99} fitness_i + \frac{\text{(最大遺伝子長)} - \text{(遺伝子の長さ)}}{\text{(最大遺伝子長)}} \quad (1)$$

最大遺伝子長を 150 とし、個体数 1000 で 50 世代の学習を実行し、得られた最も成績のよい個体を実ロボットでの学習に適用した。

4.2 実ロボットでの学習

4.2.1 個体学習の実行

本研究では強化学習の中のひとつ Actor Critic での TD 学習を用いた。

実現方法としては、行動ノードである終端ノードのそれぞれに対し、Actor Critic の場合状態価値 V と行動優先度 p のテーブルを割り当てる。つまり、全体で 3 つの Actor Critic をもつことになる。用いる状態空間は実ロボット上での状態である。また選択される行動に制限をかける。本研究では選択可能な行動として、move_forward の時は前進、後退 + 左回転、後退 + 右回転の 3 つ、turn_left のときは後退、左回転、後退 + 左回転の 3 つ、turn_right のときは後退、右回転、後退 + 右回転の 3 つとした。

また move_forward では前進が、turn_left では左回転が、turn_right では右回転が、選ばれやすいように最初にバイアスをかけた。以上の行動の行動優先度に 0.5000 を与え、その他の行動の値を 0 とした。

各値は、実ロボット上での状態が変化しているときにのみ更新するようにした。

学習時のパラメータとして、報酬はゴールしたときに 1.0、そのほかの状態では 0、学習率 $\alpha = 0.3$、割引率 $\gamma = 0.9$ と、ステップサイズパラメータ $\beta = 0.8$ とした。行動選択は Gibbs 分布によるソフトマックス手法を用いた。

4.3 実験結果

学習開始直後

シミュレータ上の GP を用いて学習した結果により、すぐに箱やゴールを見失ったりということは起こらず、ゴールと箱がロボットから近い位置にあるような配置ではタスクを達成することができた。

しかし、ゴールと箱が離れているような状態では、タスクを達成するのに時間がかかったり、ゴールや箱を見失いタスクを達成できなかったりということが見られた。

これは Actor Critic で行動が確率的に選択されるためであるといえる。例えば、一直線状にロボットと箱とゴールがあっても、設定のバイアス値では約 45%ほどでしか、前進の行動を選択しない。

6 時間後

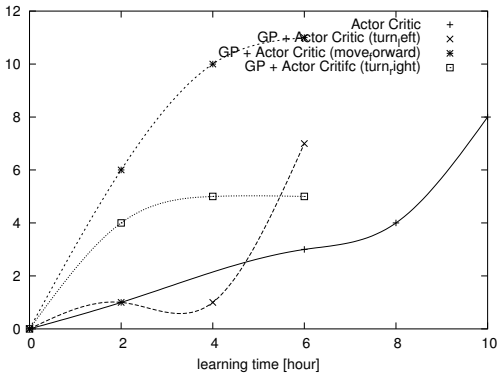

図 2: 行動確率の分散が 0.05 をこえる状態数

学習が進み、箱とゴールが離れているような状態であったり、ロボットから見て箱とゴールが左右分かれているような状態であっても、タスクを達成することができるようになった。

また、はじめはバイアスをかけていたために確率の低かった後退 + 回転の行動も、学習により確率を高め、選択するようになり、より効率的なタスクを達成ができるようになった。

これらは学習により必要な行動の確率が高められた結果といえる。

5 考察

比較実験として事前に Actor Critic の TD 学習のみを用いた学習を 10 時間行った。結果として、ゴールと箱が近いような状態からはじめても、タスクが達成できないということがしばしば起こった。これは状態空間が大きく、10 時間では実時間学習には不十分であったためである。特に、各状態においてとるべき行動の確率が高くなっていないことが考えられる。

各状態においてとるべき行動の確率が高くなると、その状態での行動確率の分散が大きくなるはずである。そこで、どの程度学習が進んだかの指標として、各状態での行動確率の分散を考えた。横軸を学習時間、縦軸を行動確率の分散が 0.05 をこえる状態の数としたグラフを図 2 に示す。turn_left, move_forward, turn_right いずれにおいても、比較実験の結果よりも、分散が 0.05 をこえる状態の数は多いことがわかる。したがって提案手法は通常の強化学習より早く効率的な学習が行われているといえ、有効に働いたことを示している。

6 おわりに

遺伝的プログラミングと強化学習を組み合わせた手法に基づき実ロボットでの実時間学習を行う手法を提案し、実験を通じてその有効性を示した。

Actor Critic を用いた学習は、連続値の入力や出力を対象とする問題に適用しやすい。今後は、そういった問題を対称にしたり、より複雑な問題に対しても適用していきたいとと思っている。

参考文献

[1] Hitoshi Iba. “Multi-Agent Reinforcement Learning with Genetic Programming”. In *Proc. of the Third Annual Genetic Programming Conference*, 1998.

[2] Keith L. Downing. “Adaptive genetic programs via reinforcement learning”. In *Proc. of the Third Annual Genetic Programming Conference*, 1998.