

軌道学習における試行回数削減のための強化学習手法 Trials Reduction Method for Reinforcement Learning in Trajectory Discovery

嘉藤 佑亮^{*1,2} 中村 友昭^{*3} 長井 隆行^{*3} 山野辺 夏樹^{*1} 永田 和之^{*1} 小澤 順^{*1}
Yusuke Kato Tomoaki Nakamura Takayuki Nagai Takayuki Nagai Kazuyuki Nagata Jun Ozawa

^{*1} 産業技術総合研究所

Advanced Industrial Science and Technology

^{*2} パナソニック株式会社

Panasonic Corporation

^{*3} 電気通信大学

The University of
Electro-Communications

In recent years, there are many researches of deep reinforcement learning to realize autonomous motion of robots. In deep reinforcement learning, a large number of trials such as thousands of times or more are required to realize sufficient performance as a learning result. However, learning in a real environment often requires assistance by people, so it is difficult to do thousands of trials. In this research, we create a learning database from efficient reinforcement learning that utilizes knowledge about tasks given by people in advance, and realize learning with a relatively small number of trials by performing mini batch learning using that database. We apply our proposed method to learning of picking task in the logistics warehouse and show the usefulness of our proposed method by comparing the results with other methods.

1. はじめに

近年、ロボットの自律的学習の実現を目指して深層強化学習の研究が多くなされている[Finn 2016] [Gu 2017] [Levine 2016]. 深層強化学習の手法では一般的に学習を完了させるために多くの学習データが必要とされる[Levine 2016] [Pinto 2016]. 特に、学習のための行動を評価した報酬は各行動ではなく、一連の行動の結果として与えられるため、学習初期の段階ではどの行動が良い結果につながるかが分からない. そのため試行錯誤の軌道制御を繰り返す必要があり、1 つの学習に何千回、何万回といった試行回数が必要となる. しかしながら、実環境での学習、特にロボットを用いたマニピュレーションタスクではロボットの周囲環境の条件を整えるためにユーザの補助が必要となる場合が多く、何千回という回数の施行を行うことは難しい.

そこで、本研究では、ロボットが把持した物体をとある場所から別の場所へ移動させるための軌道の学習において、比較的少ない試行回数で学習を完了するための手法を提案する. 人間があらかじめ軌道探索について方針を与えることでロボットを用いた効率的な探索を行い、そのときの状態遷移の情報についてデータベースを構築する. そして、構築したデータベースを用いてミニバッチ学習を行うことで、比較的少ない回数での学習を完了させる.

本手法の有効性を確認するために、倉庫環境における物品のピッキング作業を通して評価実験を実施した. その結果、100回という比較的少ない試行回数で 95.6%の成功率を実現した.

2. 強化学習の試行回数削減手法

2.1 実環境での強化学習における課題

本研究ではロボットが把持した対象物体をとある地点から別の場所へ移動させるようなマニピュレーションを行うタスクについて考える. マニピュレーションのための軌道制御を図1左に示すようなロボットの先端位置の移動によって定義する. 図1では、

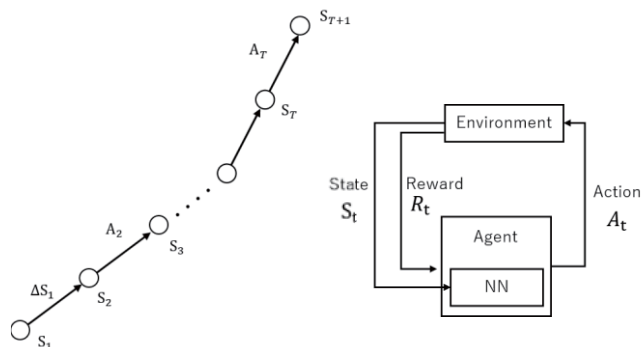


図1 軌道制御(左)とシステム構成(右)

ある時刻 t における状態として、ロボットの手先位置姿勢 $P_t = (x_t, y_t, z_t, \varphi_t, \theta_t, \psi_t)$ と手先にかかる力 $F_t = (F_x, F_y, F_z)$ とトルク $M_t = (M_x, M_y, M_z)$ からなる状態 $S_t = (P_t, F_t, M_t)$ と定義し、次にロボットの手先の行動(移動量)を A_t で表現している. とある状態で選択した行動に対して、環境から報酬 R_t を得ることで、もっとも高い報酬を得られるよう学習するシステムは図1右のように構築できる.

ロボットが把持した物体をとある地点から別の場所へ移動させるための軌道について考えるとき、軌道を構成する行動のひとつひとつに対して報酬を定義することは難しく、一連の行動の結果として報酬を与えることとなる. そのため学習初期においては、良い報酬を獲得できる軌道を発見するまでに多く試行を必要とする. また、実環境でロボットを用いて強化学習を実施した場合、ロボットが稼働する時間や動作後に対象物を初期位置へ戻すための時間が必要となるため、一般的にTVゲーム等の学習で行われている何千回といった試行を行うことはあまり現実的ではない.

そのため、実環境ではなくシミュレーション上で環境を再現し、強化学習を行うシステムも提案されている[Mahler 2016]. しかしながら、対象物の変形、摩擦、干渉等の厳密な環境モデルを構築することが困難なため、使用できる環境には制限がある.

2.2 強化学習データベースを用いたミニバッチ学習

本論文では、実環境でロボットを稼働させながらも比較的少ない試行回数で軌道学習を実現するための学習システムを提案する. 所定の施行回数で強化学習を実環境でロボットを用い

連絡先: 嘉藤 佑亮, 産業技術総合研究所 知能システム研究部門, 〒305-8560 茨城県つくば市梅園 1-1-1 中央第1つくば本部・情報技術共同研究棟, TEL:029-861-3537, E-mail:kato.yusuke@aist.go.jp

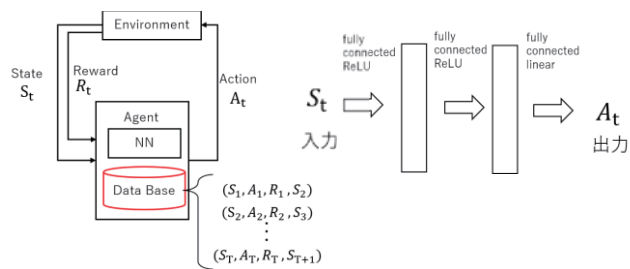


図 2 学習システム構成

で行い、そのときの状態、行動、報酬の情報を強化学習データベースに蓄積する。その後、先の学習で得た強化学習データベースを用いてミニバッチ学習を行い、その結果を最終的な学習結果として利用する。図 2 に本提案システムの概要を示す。

従来の Deep Q Network[Mnih 2013] [Mnih 2015]のような学習では少ない試行回数ではニューラルネットワークの学習回数も少なくなってしまうため、学習の結果が十分に安定しなかった。そこで本論文では、実環境でロボットを使用した強化学習時の状態、行動、報酬をすべて強化学習データベースにいったん蓄積し、その後、蓄積したデータベースを用いてミニバッチ学習を行うことでニューラルネットワークを十分に学習させる。実環境での強化学習から蓄積するデータベースのデータ構造を表 1 に示す。ここでは 1 回の施行を既定の T 回の行動、または途中で行動不能となったときと定義する。総試行回数を N 回の内、 n 回目の試行の t 回目の行動での行動前の状態を $S_{n,t}$ 、選択した行動を $A_{n,t}$ 、行動後の状態を $S_{n,t+1}$ 、行動した結果の報酬 $R_{n,t}$ で表す。これら $(S_{n,t}, A_{n,t}, R_{n,t}, S_{n,t+1})$ を 1 個としてデータベースに蓄積していく。蓄積するデータ数は行動不能で動作を途中で中断することがあるため、最小 N 個、最大 $N \times T$ 個となる。 N 回の施行終了後、蓄積されたデータを用いて改めてミニバッチ法によりニューラルネットワークを学習させる。

2.3 人の知識に基づいた探索

従来手法では学習初期にランダムな探索を行い、学習を進めいくため、少ない試行回数での学習がどれだけうまく進むかはそのランダムな探索にどれだけ有効な情報を得られるかに大きく左右されていた。そこで本手法ではユーザが軌道に関する

表 1 データ

Num	Step	State	Action	Reward	Next State
1	1	$S_{1,1}$	$A_{1,1}$	$R_{1,1}$	$S_{1,2}$
	2	$S_{1,2}$	$A_{1,2}$	$R_{1,2}$	$S_{1,3}$
	...	...	...	...	...
	T	$S_{1,T}$	$A_{1,T}$	$R_{1,T}$	$S_{1,T+1}$
2	1	$S_{2,1}$	$A_{2,1}$	$R_{2,1}$	$S_{2,2}$
	2	$S_{2,2}$	$A_{2,2}$	$R_{2,2}$	$S_{2,3}$
	...	...	...	...	...
	6	$S_{2,6}$	$A_{2,6}$	$R_{2,6}$	$S_{2,7}$
3	1	$S_{3,1}$	$A_{3,1}$	$R_{3,1}$	$S_{3,2}$
	2	$S_{3,2}$	$A_{3,2}$	$R_{3,2}$	$S_{3,3}$
	...	...	...	...	...
...	...	...	...	...	...
N	1	$S_{N,1}$	$A_{N,1}$	$R_{N,1}$	$S_{N,2}$
	2	$S_{N,2}$	$A_{N,2}$	$R_{N,2}$	$S_{N,3}$
	...	...	...	...	...
	T	$S_{N,T}$	$A_{N,T}$	$R_{N,T}$	$S_{N,T+1}$



図 3 物流倉庫でのピッキング作業

知識を提供することで探索に一定の方向性を持たせ、効率的な探索を行う。

従来の強化学習では、タスクに関する事前知識などは与えず、取りうる行動をランダムに探索を行っていた。しかしながら、本研究で対象とする把持した物体を移動させるタスクでは移動先の地点、または領域が存在するため、大まかな移動方向えお決めることができる。また、把持した物体を移動させる場合、移動の軌道としては急激な変化は少ない方が望ましい。このことから本手法では、探索のための方針として以下の 2 つを与える。

- (1)初期の移動方向の教示
- (2)急激な移動方向の変化の制約の教示

(1)に関して、移動のための軌道学習を行う場合、タスクの解となる全体の詳細な軌道は分からなくとも軌道の方向性は分かっていることが多い。そこでその方向を基準に重点的に探索することで、完全ランダムに探索を行うよりも効率よく質の良いデータを集める。

(2)に関して、把持した物体を移動させるための軌道制御においては、安定した運搬を行うために軌道はある程度滑らかな軌道の方が良く、極端な移動が極力避けるべきである。そこで、ランダムに探索を行うときの取りうる行動の選択確率に差を付け、現在の移動方向と同じ方向が選ばれやすくなるよう設計する。

3. 実験

この章では第 2 章で提案した手法を倉庫環境におけるピッキングタスクに適応し、学習させた例を示す。

3.1 倉庫環境におけるピッキング作業

本提案手法を用いて動作の学習を行うことに適した環境の一つとして物流倉庫環境を紹介する。物流倉庫内では小売店舗向けのピッキング作業が行われている。作業者は各小売店舗からの発注に応じた個数の商品をピッキングし、輸送用コンテナへ移動させる作業を行う(図 3)。物流倉庫内で扱われる商品はビニールやフィルム等で包装され(図 4a)、箱の中に隙間なく整列して詰められた状態で棚に置かれている(図 4b)。我々は図 4b のように箱の中に隙間なく整列して詰められている状態を充

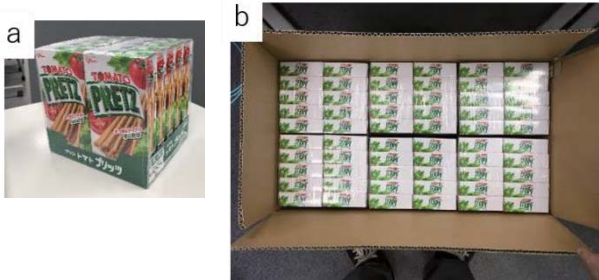


図 4 物流倉庫内の商品の一例

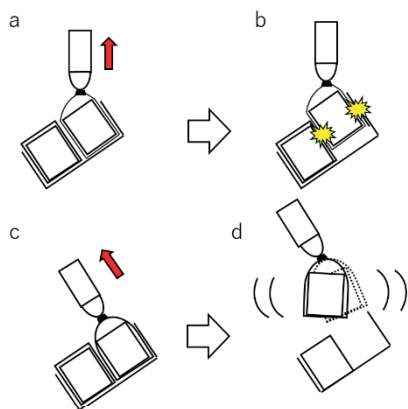


図 5 物流倉庫環境での課題

填状態と呼んでいる。また、商品の置かれている棚の棚板には取り出しやすさを考慮して傾斜がつけられていることが多い。

以上のような環境である物流倉庫内でロボットシステムが自律的にピッキング作業を行うことを考える。物流倉庫内の環境で吸着によって商品を持ち上げて取り出すときに課題が 2 つある。1 つは、持ち上げる方向である。対象物を吸着によって把持し、地面と垂直方向に持ち上げようとしたところ、物体同士の隙間が小さいため、傾斜をつけて置かれているこの環境下では周囲の物体と衝突し、持ち上げることができなかった(図 5a-b)。そのため、ピッキング動作の初めは周囲の物体と干渉しない方向に持ち上げる必要がある。

もう一つはピッキング動作の軌道である。棚についている傾斜に対して垂直方向に引っ張ると、周囲と干渉することなく持ち上げることができる。しかし、箱から対象物の全体が取り出されたときに対象物が回転し大きく揺れてしまい吸着が外れ落とすことがある(図 5c-d)。

これらのことから物流倉庫内のピッキング作業では単純な方法によるピッキングは成功せず、対象物を落とさないように取り出せる軌道を見つける必要がある。

また、物流倉庫内のピッキング動作はビニールの伸びや物体同士の摩擦などに影響を受ける。そのためシミュレーション上で動作学習をさせることは難しく、実環境で行う必要がある。

3.2 実験システムの構成

図 6a のように倉庫環境を模した実験環境を用意した。棚板に 20 度の傾斜がついた棚に充填状態にある商品が置き、その正面に手先に吸着機構を備えたマニピュレーションロボットを設置している。対象物体の 1 つあたりの大きさは 17[cm]×14[cm]×16[cm]、重さ 915[g]であり、1 つの段ボール箱の中に 12 個詰められている。ロボットは KUKA 社の 7 軸多関節ロボットアーム LBR iiwa14 R820 を使用した。図 5b にロボットの手先部分を示す。手先には真空吸着式の吸着機構と力覚センサが備え付けられている。力覚センサは wacoh-tech 社の静電容量型 6 軸力覚センサ DynPick を使用しており、3 方向の力成分と 3 方向のモーメント成分の 6 軸を同時に検出することが可能である。以上のロボットシステムを使用し、棚に置かれた充填状態の箱から対象物体を 1 つ吸着し取り出す軌道を学習させる。

3.3 学習動作

学習動作は、物体の取り出し軌跡であり、2.2 節で紹介したような各状態における一連の行動として定義する。今回の実験では 1 回の行動での移動量は 2.0[cm]と固定し、各状態における移動方向の変化を学習させる。なお、手先の姿勢は移動方向に合わせて変化させるものとする。

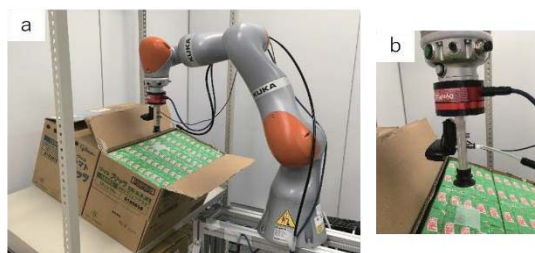


図 6 実験システム

まずロボットはユーザからあらかじめ与えられた手先位置で図 5c のように対象物に対して垂直となる姿勢で吸着を行う。この位置姿勢をロボットの手先の初期の位置姿勢とする。この初期の位置姿勢から吸着したまま 2[cm]単位の移動を繰り返し行い、12 回移動する、または 12 回の移動を終えるまでに対象物体から吸着が外れたとき動作を終了する。終了後、実験者によって対象物体を元の位置に戻す。ここまで一連の流れを 1 試行として扱う。

また、本実験環境ではロボットの手先移動は x-z 平面上の移動のみを行い、y 軸方向への移動は行わない。同様に手先姿勢についても y 軸まわりの回転だけを行い、x,z 軸の周りの回転は行わない。

3.4 軌道学習のための制御プログラム

動作学習のための制御プログラムをオープンソフトウェアである深層学習ライブラリ chainerrl を利用し、Deep Q Network (DQN) の手法を基本として実装した。

2.3 節、3.1 節で説明したタスクに関する知識を本実験における人の知識として実装に組み込む。本実験ではロボットが取りうる行動を、手先を 2[cm]移動するときの移動方向の変化量として表す。変化量は -10[deg] から 10[deg] までの 5[deg] 刻みの値とし、変化量が 0[deg] のとき、現在の移動方向のまま移動することになる。また、探索の手法として Linear Decay ϵ -greedy 法があるが、本実験ではこれに変更を加えたものを探索の手法として使用する。通常の Linear Decay ϵ -greedy の手法では行動ごとに線形的に減少する ϵ の値を用いて、 ϵ の確率でランダムな行動(探索)、 $1-\epsilon$ の確率で最も Q 値の高い行動を選択するが、本実験では ϵ の確率での探索時にランダムに行動選択をするのではなく、移動方向の変化を 0[deg] とする行動が他の行動に比べて選択されやすくなるように設計する。今回の実装では、探索時に移動方向の変化を 0 度とする行動が選ばれる確率を 50% とし、残りの 4 つの行動はそれぞれ 12.5% の確率で選ばれるように設計した。

学習に使用するロボットの手先位置などの関係を示した図を図 7 に示す。学習に際して、ニューラルネットワークの入力となる状態として対象物を吸着した位置(初期位置)からの現在のロボットの手の位置へのベクトルと手先姿勢、力覚センサから得られた手先に掛かる力とモーメントを用いた。ただし、本実験では y 軸方向への移動と x,z 軸周りの回転は行わないため、y 座標、y 方向の力、x,z 軸周りのモーメントは入力として使用しない。ネットワークはチャンネル数 30 の中間層 2 層からなるものを使用する。出力として、手先の現在の移動方向の変化量を -10[deg] から 10[deg] までを 5[deg] 刻みの 5 つの選択肢に対する Q 値を出力する。図 8 にネットワークの構造を示す。

以上のような設定を用いて強化学習による動作学習を進め、2.2 節で紹介したように状態と行動、報酬をデータベースに蓄積していく。この実験では 100 試行分のデータを蓄積させた。そして収集したデータを用いて改めてミニバッチ学習を用いてニュー

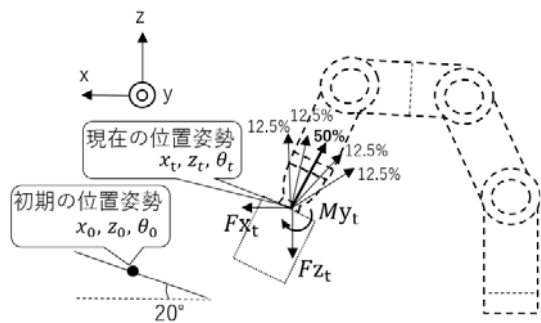


図 7 実験システムの概略図

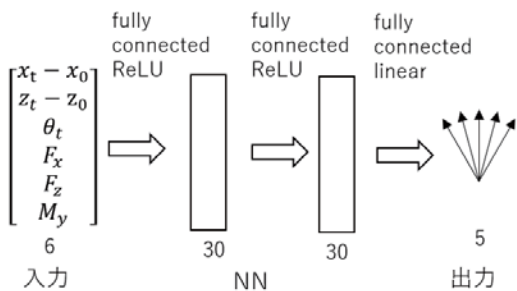


図 8 ネットワーク構造

ーラルネットワークの学習を 100epoch 行い、その結果を最終的な学習結果として使用する。このときの学習に使用するネットワークの構造も図 8 と同じものを使用する。

4. 評価

ここでは 3. 4 節で紹介した学習を行った結果を用いて、取り出し動作を実施し、本提案手法の有用性を評価する。

評価のために学習データベース(DB)や人の知識の有無で下記の 4 条件を用意し、それぞれの条件での学習結果を用いた物流倉庫内の取り出し動作の成功率を比較する。

- (a) 強化学習 DB と人の知識、どちらも使用しない学習
- (b) 強化学習 DB を使用した学習
- (c) 人の知識を使用した学習
- (d) 強化学習 DB と人の知識を利用した学習 (提案手法)

試行回数を 100 として各条件で学習を 3 回ずつ行い、それぞれの学習結果を用いて 30 回ずつの取り出し動作を行った。

各条件での取り出し動作の成功率を表 2 に示す。本提案手法は 100 回という比較的少ない試行回数の強化学習において全体で 95.6%と高い成功率を示し、学習ごとのばらつきも小さくなった。強化学習 DB を使用していない条件 a,c は、成功率に大きく差が出ており、全体的な成功率も 50.0%, 62.2%となった。条件 b は条件 a,c に比べると学習ごとのばらつきは小さくなっているが、全体的な成功率は 87.8%と条件 d よりも低い結果となった。

今回の実験では 1 種類の物体のみを対象として学習を行った。そのため、重さや大きさが変わったときに、今回の学習結果が利用できるかは確認できてない。学習時に入力として対象物体の情報を使って無次元化すれば、ある程度の汎用性は出せると予想できる。しかしながら、物体同士の隙間がより狭いなど環境が大きく異なる場合は、求められる軌道も変わってくるため、その環境に適した学習が必要になると考えられる。

また、今回の学習の特徴として、学習の中に人の知識を組み込んだが、人の知識は画一的なものではなく、その人の熟練度によって内容に差が生じる。そのため、知識の質の良し悪しが学習結果に影響を及ぼすことも考えられる。

表 2 学習結果の比較

学習条件	a	b	c	d
1	50.0	73.3	56.7	100.0
2	0.0	100.0	93.3	100.0
3	100.0	90.0	36.7	86.7
全体	50.0	87.8	62.2	95.6

5. おわりに

本稿ではロボットが把持した物体をとある場所から別の場所へ移動させるための軌道の学習において、比較的少ない試行回数で学習を完了するための手法を紹介した。所定回数の強化学習を行い、その時の状態、行動、報酬をデータベースに蓄積する。そして蓄積したデータベースを用いて後に再度ミニバッチの手法で学習させる。この手法により比較的少ない施行回数で安定した学習を実現する。また、データの蓄積の際に人から与えられた教示に基づいて学習を進めることで、効率よく有用なデータを収集する。評価のため本提案手法を用いて学習した結果を、データベースや人の知識を使わないで学習させたものと比較した。その結果データベースや人の知識を使わないで学習させた結果が 50.0%, 62.2%, 87.8%だったのに対し、提案手法は 95.6%の成功率を示した。

参考文献

[Finn 2016] Finn, Chelsea, et al. "Deep spatial autoencoders for visuomotor learning." Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016.

[Gu 2017] Gu, Shixiang, et al. "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates." Robotics and Automation (ICRA), 2017 IEEE International Conference on. IEEE, 2017.

[Levine 2016] Levine, Sergey, et al. "End-to-end training of deep visuomotor policies." The Journal of Machine Learning Research 17.1, 1334-1373, 2016.

[Levine 2016] Levine, Sergey, et al. "Learning hand-eye coordination for robotic grasping with large-scale data collection." International Symposium on Experimental Robotics. Springer, Cham, 2016.

[Mahler 2016] Mahler, Jeffrey, et al. "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards." Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016.

[Mnih 2013] Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." arXiv preprint arXiv:1312.5602, 2013.

[Mnih 2015] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." Nature 518.7540, 2015.

[Pinto 2016] Pinto, Lerrel, and Abhinav Gupta. "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours." Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016.