

過去の学習経験を組み合わせることで適応するロボット

田中 一敏

システムインテグレータは顧客の要望する作業を自動化するためにロボットをプログラミングする。ロボットシステムが構築される時だけではなく生産内容に軽微な変更が生じる時にも都度、プログラミングに大きな労力が必要とされている。もしロボットが自律的に新しい作業を学習できれば、この労力を省ける。このため、筆者らはロボットが新しい作業を効率的に学習する手法を新たに提案した。提案手法を用いたロボットは、過去の経験を活かして新しい作業を短時間で学習する。挿入方向の異なるペグインホールの実験を行い、提案手法の有効性を確認した。この手法によって、ロボットが新しい作業を短時間で学習できるようになり、ロボットを製造やサービスの新しい用途に利用できるようになる。

Adaptation to a New Setup by Combining Skills

TANAKA Kazutoshi

The robotic system integrator programs the robot to automate the work requested by the customer. Robotic work for new parts that require minor changes in production requires a large amount of effort to programs the robot. If the robot can learn new tasks autonomously, this effort can be eliminated. For this reason, we have proposed a method for robots to learn new tasks. The robot using the method learns new tasks in a short period of time by utilizing its experience. To evaluate the usefulness of the method, we conducted experiments on a peg-in-hole task with different insertion directions. The method enables the robot to perform new tasks in a short time, which will lead to new ways of using robots for manufacturing and service.

1. まえがき

本稿では、ロボットが新しい作業を学習する必要性について解説し（2章）、ロボットが新しい作業を学習する既存手法について検討し（3章）、筆者らが開発した過去の経験を活用してロボットが新しい作業を高速に学習する手法（4章）と実験結果（5章）を紹介する。本稿は、筆者らが発表した論文¹⁾を参照し、過去の学習経験を組み合わせることで適応するロボットについて解説したものである。

2. ロボットによる新規作業学習の必要性

本章では製品生産を例として、産業用ロボットがどのように使われるかを紹介し、ロボットが新しい作業を学習する必要性について述べる。

2.1 ロボットシステムの構築方法²⁾

産業用ロボットは、半製品と呼ばれる。すなわち、産業用ロボットのみ購入しただけでは、使えない。ロボットエンジニア、あるいはロボットのシステムインテグレータ

（以後、SIerと表記）と呼ばれる人間が、ロボットを他の機械と組み合わせることで、はじめてロボットシステムは完成し、利用できるようになる。

ロボットシステムを導入する手順として、RIPS (Robot system integration process standard) が定義されている。この手順における各フェーズ（各作業工程）とSIerの仕事は以下のとおりである。

1. 準備フェーズ（引合、企画構想）：顧客からの自動化要望の取りまとめ
2. 設計フェーズ（仕様定義、基本設計、詳細設計）：要件化、システム仕様の定義、設計
3. 製造フェーズ（製造、内部テスト）：加工、組立、配線、プログラミング
4. テストフェーズ（統合テスト、ユーザテスト）：テスト
5. 稼働後（運用保守サポート）：保守

設計フェーズにおいてSIerは、操作対象物体と物体に対する操作を定義して、ロボット、エンドエフェクタ、セン

Contact : TANAKA Kazutoshi kazutoshi.tanaka@sinicx.com

サなどを選択し、ロボット用の架台、スライダ、治具、ポジショナーを設計する。その後、ロボットの動作をプログラミングする。

ロボットへのプログラミングは主に以下の2方式がある。

- ・ティーチングプレイバック方式
- ・マニュアル数値入力方式

ティーチングプレイバック方式では、移動させたい位置をロボットに教示し、記憶させ、その位置へどのように移動するかを指示する。一般的に産業用ロボットはティーチングプレイバック方式（教示再生方式）でプログラミングする。マニュアル数値入力方式では、ロボットの位置姿勢を表す数値を直接入力する。

ティーチングプレイバック方式は、さらに以下の3方式に細分される。

- ・ダイレクトティーチング
- ・オンラインティーチング
- ・オフラインティーチング

ダイレクトティーチングでは、ロボットを直接掴んで動かしながらプログラミングする。オンラインティーチングでは、ティーチングペンダントを使ってロボットを動かしながらプログラミングする。オフラインティーチングでは、ロボットを動かさず、部品の幾何的情報を用いたシミュレーションによってプログラミングする。これら3つの方式はいずれもロボットプログラミングにおいて多くの労力を要する。

2.2 新規の物体や作業に対するロボット自動適応

生産ラインでは、製品のマイナーチェンジなどがあり、それにともなうロボットの作業をプログラミングし直す必要がある²⁾。加えて、現在行われているような単一品種大量生産ではなく、ロボットによる多品種少量生産を目指す場合、新しい部品の作業動作を、短時間でロボットに実装する必要がある。

新しい部品を扱う新しい作業におけるロボットの動作を、オフラインティーチングによって自動で生成する方向も検討されている³⁾。オフラインティーチングは部品や作業の変化のうち幾何的な変更に対応するのに使われる。他の、部品組立作業などの接触を多く含む動的な作業などに使える方法として、機械学習を用い、試行錯誤を通じて、ロボットの動作を獲得する方法もある。以下では、このロボットによる作業学習について検討する。

3. ロボットによる新しい作業の学習

代表的なロボット作業である部品組立作業を例として、ロボットによる新しい作業の学習について考察する。特に、部品組立作業における基礎的な動作であるペグインホールに着目し、新しい傾きの穴への挿入を学習することを考える。続いて、そのようなロボットによる新しい作業の学習を可能にする手法について既存研究を概観するとともに、筆者らの提案する新たな学習手法について解説する。

3.1 ロボット学習課題としての製品組立作業の新規物体操作

ロボットやエージェントが試行錯誤を通じてタスクの解き方を学ぶ技術として、強化学習⁴⁾がある。強化学習は、ロボットに作業を自動で獲得させることを可能にする。現在も強化学習を用いたロボットの作業学習に関する研究は活発に行われている。しかし、ロボットが既に学習した内容（経験）を活用して新しいタスクを短時間で学ぶことは未だ挑戦的な課題である。

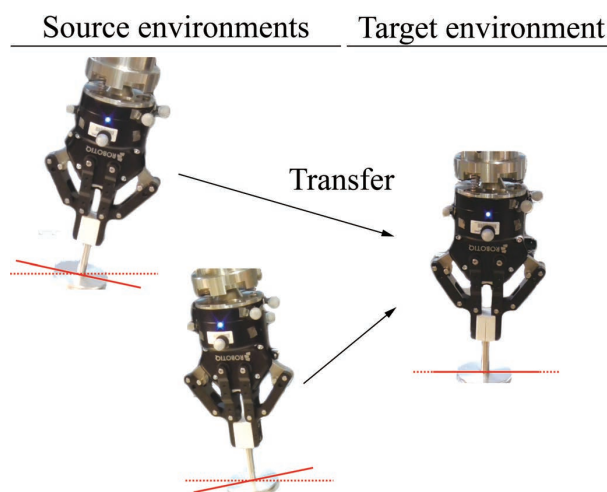


図1 本論文における問題設定：過去の経験を活用した新しい傾きの穴へのペグインホールの学習

ロボット作業の基礎的な動作であるペグインホールについて考える。特に既にペグの挿入傾きが異なる場合のペグインホールを学習したロボットが、同じ部品（ペグ）を未知の挿入傾きでペグインホールを学習することを想定する（図1）。これはペグ挿入の傾きが異なる一連のペグインホールタスクであり、実際の製造において生じる製品の軽微な変更を想定している。

この新しい傾きの穴へのペグインホール学習は、転移強化学習の課題として捉えられる。新しい傾きの穴へのペグインホールを転移強化学習と捉える場合、過去実施した傾きの穴へのペグインホールはソース環境と呼ばれ、新しい傾きの穴へのペグインホールはターゲット環境と呼ばれる。ペグインホールの転移強化学習は具体的には、以下の

問題として定式化できる。

1. ソース環境とターゲット環境は同じタスクだが、状態遷移ダイナミクスは異なる。
2. 正確な環境のダイナミクスは未知である。
3. ターゲット環境における学習時に、ソース環境との通信は不可能である。

ソース環境とターゲット環境の状態遷移ダイナミクスが異なる場合、ペグと穴が接触する位置も異なるため、穴の方向の小さな変化がロボットの状態遷移に大きな影響を及ぼす。すなわち、同じ状態で同じ行動を取っても、次の状態であるペグの位置や姿勢も変わる。ゆえに、穴の向きに応じた制御器が必要である。状態遷移ダイナミクスの完全なパラメータは未知か、不正確か、動的に変化するものである。ここでいうパラメータは、例えばリンクの重量、慣性テンソル、関節の粘性摩擦などがある。

さらに、実際の製造において、工場のラインにロボットは分散して配置されており、ロボット間の通信チャンネルはほとんどない。このため、ロボット同士が情報を大規模に収集、交換するのは難しい。

3.2 製品組立作業のロボット学習

強化学習のうち、モデルベース強化学習は、サンプル効率が高く、ロボットの学習に適している。近年では、複雑なダイナミクスを扱うために、深層ニューラルネットワークが用いられている⁵⁾。

転移強化学習⁶⁾も、他のタスクで得られた経験を使うことでサンプル効率を高められる。転移強化学習は、報酬関数や、状態遷移ダイナミクスや、状態行動空間が既知のソース環境と新しい環境が異なる強化学習として分類される。方策、Q 関数、ダイナミクスモデルを転用する。転移強化学習に関する多くの成果があるが、状態遷移ダイナミクスの異なる環境間での転用に関する研究は限られている⁷⁻¹¹⁾。特に、モデルフリー強化学習の手法が多く、サンプル効率の高いモデルベース強化学習には使えない。

他の関連する研究領域はメタ学習である。メタ学習は、未知のタスクにロボットエージェントが高速に適応する方法である¹²⁻¹⁶⁾。メタ学習の多くはモデルフリー強化学習であるため、実用的なロボットの製品組立作業に使うのは難しい。

4. ダイナミクスモデルを集約するモデルベース転用強化学習¹⁾

上述した問題を標準的な強化学習の問題として、あらためて定式化する。ロボットの部品組立作業をマルコフ決定過程としてモデル化する。状態と行動の空間 S と A 、状態遷移 T 、報酬関数 R からなるタプルでマルコフ決定過程は

定義される。

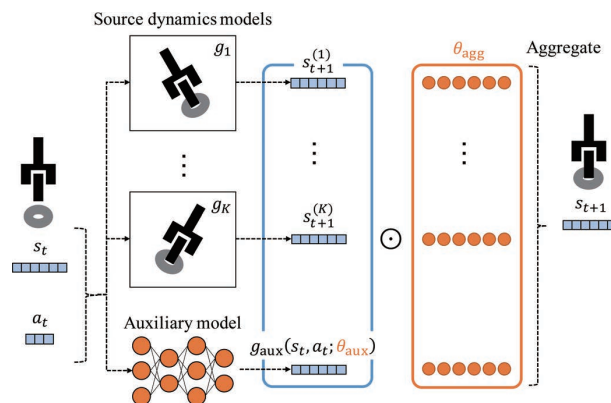


図2 提案手法 既知モデルの集約による新規モデルの学習

強化学習における目標は、報酬関数の時間総和である収益を最大化する最適な制御器を得ることである。提案手法には、サンプル効率の優れたモデルベース強化学習を用いた。提案手法は、実際のダイナミクスから収集したデータを用いて、状態遷移 T を関数 g のモデルで近似する。モデルは行動の系列から状態の系列を予測するため、予測に基づいて最適な行動を選択できる。

転移強化学習の設定を用いた。ダイナミクスモデル g_k で表現された K 個の関連する既知環境 (source environment) を考える。このモデルは、パラメタライズされてなくても、訓練できなくても良い。例えば、学習したニューラルネットワークでも、システムを近似したヒューリスティックな規則を手動で設計したシミュレータでも良い。モデルは、固定された訓練するパラメータのないブラックボックス関数でも良い。新しい環境における目標は、未知の状態遷移を近似したダイナミクスモデル g^{target} を短時間で学べるよう $G = \{g_k\}$ を活用してモデルベース強化学習のサンプル効率でさらに改善することである。

提案手法である TRANS-AM は、過去のモデルの出力の線形和として、新しいモデル出力を計算する。それに加えて、線形和の残差を補償するため、残差モデルを学習する (図2)。

すなわち、提案手法は既知のダイナミクスモデルの出力を適応的に集約し、同時に、集約結果の残差を予測し、補償する補助モデル (auxiliary model) を学習する。既知モデルの出力 $s_{t+1}^{(k)} = g_k(s_t, a_t)$ は以下のように結合される。

$$S_{t+1} = \left[\left(s_{t+1}^{(1)} \right)^T, \dots, \left(s_{t+1}^{(K)} \right)^T \right] \in \mathbb{R}^{K \times D}$$

ここで、 D は状態空間 S の次元である。新規環境におけるダイナミクスモデルを以下のように表現する。

$$g^{target} = \mathbf{1} \left(\theta_{agg} \odot S_{t+1}, g_{aux}(s_t, a_t; \theta_{aux})^T \right)$$

ここで、 $\theta_{agg} \in \mathbb{R}^{(K+1) \times D}$ は訓練可能な補助重みパラメータ

の行列、 $\odot$ は要素ごとの積、 1 は $K+1$ の大きさの全ての要素が 1 の列ベクトルである。この θ_{agg} は正則化しない。

提案手法は、モデルベース強化学習のアルゴリズムで、ネットワークのパラメータ $\theta = (\theta_{agg}, \theta_{aux})$ を訓練する。Cross entropy method を用いて、モデルに基づく行動を生成する。

提案手法の利点をまとめる。提案手法はモデルベース強化学習であり、サンプル効率が低い。提案手法では収集した過去のダイナミクスモデルを利用する。この過去のダイナミクスモデルとして、ニューラルネットワークだけでなく、他の非線形関数による収集したデータの近似関数も利用できる。他にも、あらゆるブラックボックスが利用できる。例えばヒューリスティックに状態遷移が決まるシミュレータ、物体の先験的知識を用いた概形とパラメータの同定によって作ったモデルなどが利用できる。

5. 実験¹⁾

提案手法の実用性を評価するため、異なる穴の傾きのペグインホールタスクを行った (図3)。実験には、ロボットアーム UR5 (Universal Robots) を用いた。ロボットアームには筆者らが開発した柔軟手首¹⁷⁾、グリップ (2F-85, ROBOTIQ)、6軸力覚センサ (FT300, ROBOTIQ) を搭載した。グリップの位置と姿勢は6個のモーションキャプチャカメラ (FLEX13, OptiTrack) で計測した。ペグとして、ステンレス製の、直径 10 mm のものを用いて、グリップに固定した。ペグと穴の公差は H7/h7 であった。異なるペグの傾きを試した。 $\phi \in \{-10^\circ, 0^\circ, 10^\circ\}$ と変更した。

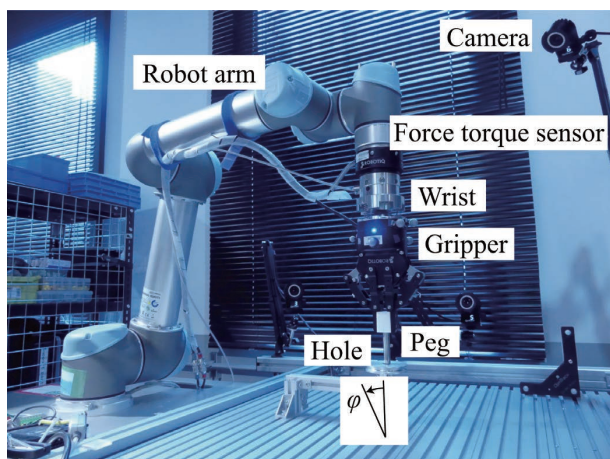


図3 実験に用いたロボットシステムの概観

モデルベース強化学習における行動として、ロボットの手先 (アームの先端) 速度の水平方向成分を用いた。ただし手先の傾きは一定に維持すると仮定する。行動として決められた目標手先速度から、ロボット制御のフレームワークである MoveIt¹⁶⁾ を用いて目標関節角速度を計算し、5 Hz でロボットに送った。手先の位置、接触力の並行成分、

グリップの位置と姿勢を用いて状態を定義した。水平成分の位置と力は3変数で、グリップの姿勢はオイラー角の正弦と余弦からなる6変数で表現した。この際に、グリップの位置は目標位置を原点とする相対位置で表現して正規化し、力も特定の大きさを正規化した。また報酬は、目標位置までの距離と接触力の線形和で定義する。

ダイナミクスモデルはモデル予測制御で制御し、取得したデータから学習した。ネットワークは Adam で最適化し、学習率は、0.01 に設定した。 $\phi=10^\circ$ から $\phi=0^\circ$ への転移 ($K=1$) と、 $\phi=\{-10^\circ, 10^\circ\}$ から $\phi=0^\circ$ への転移 ($K=2$) の TRANS-AM を試した。

学習の1セッションは20エピソード、1エピソードは100タイムステップとした。過剰な接触力が生じるか、穴からペグが過剰に離れた場合、ロボットや部品の破損を防ぐため、エピソードを失敗したとして、終了した。最初の2エピソードでは、ランダムに行動を決定した。異なる5つのランダムシードを試し、5, 10, 15, 20 エピソードでの成功率を計算した。

表1 初回成功エピソードの平均と標準偏差
各エピソードにおける成功率

	First Success	e = 5	10	15	20
Baseline	9.25 ± 5.49	0.2	0.4	0.6	0.8
TRANS-AM (K=1)	8.20 ± 5.34	0.4	0.6	0.8	1.0
TRANS-AM (K=2)	7.20 ± 3.37	0.4	0.6	1.0	1.0

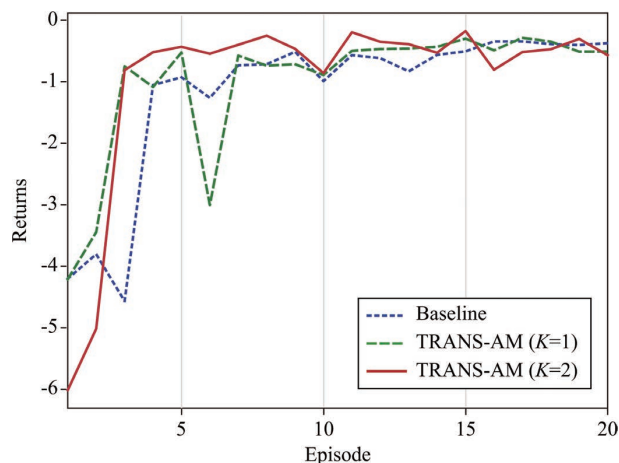


図4 収益比較

表1に成功率と初回成功の平均と標準偏差を示す。TRANS-AMがソースダイナミクスモデルを用いて、先見知識なしに学習するベースラインよりも、初回の成功を早めたことが分かる。図4から、TRANS-AMの方が、早いエピソードで高い報酬を得たことが分かる。図5に成功したエピソードのスナップショットを示した。以上のように

に、TRANS-AM が短期間で適応を可能にすることが分かった。

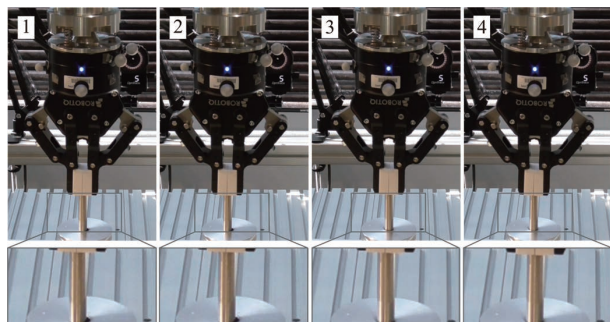


図5 学習したペグインホルルの外観

6. むすび

本稿では、新しい作業をロボットが学習する必要性について解説し、筆者らの提案するロボットの学習方法を紹介した。

提案手法によって、ロボットが過去の経験を活用することで、新しいタスクに対して高速に作業を学習することが可能になった。この手法を発展させて使うことで、生産する製品の軽微な変更に対して自動で新しいロボット動作を作成できる。加えて、ロボットによる多品種少量生産が可能になり、一人ひとりに合った製品を届けることが可能になる。さらに、ロボットプログラミングのコストを下げることで、製品組立作業にとどまらず、様々な単純手作業のロボットによる代替が現実的な選択肢になる。

多数のモデルを既に学習していた際に、どのモデルを学習するのか、また、将来においてダイナミクスを再利用することを考慮して、使いやすい形でモデルを学習する方法は、この研究において残された課題であり、筆者たちが今取り組んでいる課題である。

参考文献

- 1) Tanaka, K.; Yonetani, R.; Hamaya, M.; Lee, R.; Drigalski, F.; Ijiri, Y. TRANS-AM: “Transfer Learning by Aggregating Dynamics Models for Soft Robotic Assembly”. IEEE International Conference on Robotics and Automation. 2021.
- 2) 経済産業省, 日本ロボット工業会. ロボットインテグレータのスキル読本. 2018, 87p.
- 3) 小沢邦昭, 熊本健二郎, 明石吉三, 中田英樹. オフラインロボット教示における高速干渉チェックの一方式. 日本ロボット学会誌. 1986, Vol.4, No.2, p.79–88.
- 4) Sutton, R. S.; Barto, A. G. Reinforcement Learning: An Introduction, MIT Press, 1998, 344p.
- 5) Chua, K.; Calandra, R.; McAllister, R.; Levine, S. “Deep reinforcement learning in a handful of trials using probabilistic dynamics models”. Advances in Neural Information Processing Systems, 2018, p.4754–4765.
- 6) Taylor, M. E.; Stone, P. Transfer learning for reinforcement learning domains: A survey. J. Mach. Learn. Res. 2009, Vol.10, No.7, p.1633–1685.
- 7) Chen, T.; Murali, A.; Gupta, A. “Hardware conditioned policies for multi-robot transfer learning”. Advances in Neural Information Processing Systems. 2018, p.9333–9344.
- 8) Barekatin, M.; Yonetani, R.; Hamaya, M. “Multipolar: Multi-source policy aggregation for transfer reinforcement learning between diverse environmental dynamics”. International Joint Conference on Artificial Intelligence. 2020, p.3108–3116.
- 9) Song, J.; Gao, Y.; Wang, H.; An, B. “Measuring the distance between finite markov decision processes”. International Conference on Autonomous Agents and Multiagent Systems. 2016, p.468–476.
- 10) Yu, W.; Liu, C. K.; Turk, G. “Policy transfer with strategy optimization”. International Conference on Learning Representations. 2019.
- 11) Rajendran, J.; Lakshminarayanan, A. S.; Khapra, M. M.; Prasanna, P.; Ravindran, B. “Attend, adapt and transfer: Attentive deep architecture for adaptive transfer from multiple sources in the same domain”. International Conference on Learning Representations. 2017.
- 12) Vanschoren, J. Meta-learning: A survey. arXiv:1810.03548, 2018, (参照 2021-10-01).
- 13) Nagabandi, A.; Clavera, I.; Liu, S.; Fearing, R. S.; Abbeel, P.; Levine, S.; Finn, C. “Learning to adapt in dynamic, real-world environments through meta-reinforcement learning”. International Conference on Learning Representations. 2019.
- 14) Sæmundsson, S.; Hofmann, K.; Deisenroth, M. “Meta reinforcement learning with latent variable gaussian processes”. Conference on Uncertainty in Artificial Intelligence. 2018, Vol.34, p.642–652.
- 15) Clavera, I.; Rothfuss, J.; Schulman, J.; Fujita, Y.; Asfour, T.; Abbeel, P. “Model-based reinforcement learning via meta-policy optimization”. Conference on Robot Learning. 2018, p.617–629.
- 16) Schoettler, G.; Nair, A.; Ojea, J. A.; Levine, S.; Solowjow, E. “Meta-reinforcement learning for robotic industrial insertion tasks”. International Conference on Intelligent Robots and Systems, 2020. p.9728–9735.
- 17) Tanaka, K.; Drigalski, F.; Hamaya, M.; Lee, R.; Nakashima, C.; Shibata, Y.; Ijiri, Y. “A Compact, Cable-driven, Activatable Soft Wrist with Six Degrees of Freedom for Assembly Tasks”. IEEE/RSJ International Conference on Intelligent Robots and Systems. 2020, p.8752–8757.
- 18) Chitta, S.; Sucan, I.; Cousins, S. Moveit! IEEE Robotics & Automation Magazine. 2012, Vol.19, No.1, p.18–19.

執筆者紹介



田中 一敏 TANAKA Kazutoshi
オムロン サイニックス株式会社
リサーチアドミニストレイティブディビジョン
専門：知能ロボット工学
所属学会：日本ロボット学会
博士（情報理工学）

本文に掲載の商品の名称は、各社が商標としている場合があります。

