The 32nd Annual Conference of the Japanese Society for Artificial Intelligence, 2018

Generative Adversarial Imitation Learning に タスク達成報酬を付加した動作の学習

Learning actions using
Generative Adversarial Imitation Learning added task achievement reward

黄瀬 輝 *1　　谷口 忠大 *1
Akira Kinose　　Tadahiro Taniguchi

*1 立命館大学
Ritsumeikan University

In manipulation tasks, it is extremely costly for a human to manually create a program such as the complex motion like a human arm. On the other hand, learning manipulation tasks using machine learning has a huge range of actions, so it is difficult for robots to learn without demonstrations by a human. For these reasons, it is required for robots to learn motions smoothly using human demonstrations. In this research, based on Generative Adversarial Imitation Learning (GAIL) which is a representative method of Learning from Demonstrations (LfD) in recent years, we propose the method that can learn manipulation actions in the simulation environment by newly using task achievement rewards. As a result, the proposed method succeeded in learning the policy of manipulation task which was difficult to learn in the existing models.

1. はじめに

近年，ロボットは研究の発展とともに利用価値が高まり，日常生活環境や災害現場などで人の代わりに作業を行うことが期待されている．ロボットアームの制御によって物体の把持，移動を行うマニピュレーションは，ロボットが作業を行う上で最も基本的な技術の一つである．しかし，人間の腕の動作のような，複雑な制御を伴うプログラムを手作業で作成することは，非常に手間がかかる．一方，機械学習を用いたマニピュレーション制御は行動の取りうる範囲が膨大であるため，ロボットが人間による教示無しで学習を行うことは困難である．このような理由から，人間の教示によってロボットの動作を獲得することが必要である．本研究では，これらの問題に対して，タスクのデモンストレーションをもとに動作の学習を行う Learning from Demonstration (LfD) に焦点を当てる．

近年の LfD に代表される手法として，Ho らによる Generative Adversarial Imitation Learning (GAIL)[Ho 16] が挙げられる．GAIL は環境のダイナミクスと人間による特徴量の設計が不要かつ，高次元環境でのモデルフリーの優れた LfD の手法である．従来，逆強化学習によって推定した報酬関数に基づいて強化学習を行うことで，タスクのエキスパートの動作を学習していたのに対し，GAIL は Generative Adversarial Networks (GAN) [Goodfellow 14] を適用させることでこれらを統一的に扱っている．GAIL の特徴として，環境からの報酬を受け取らず，エキスパートの軌道から推論した報酬関数のみを用いて学習を行っていることが挙げられる．しかし，本研究で想定するような，タスクのゴールが存在するマニピュレーションタスクでは，タスク達成の要因となるオブジェクトやゴールの初期位置が変化するため，環境からの報酬を用いずに学習することは困難であると考えられる．

本研究では，タスクのゴールが存在するマニピュレーションタスクにおいて，LfD の代表的手法である GAIL のモデルにタスク達成報酬を導入することで，パフォーマンスの向上を図る．また，その実験として，MuJoCo シミュレーション環境 [Todorov 12] のマニピュレーションタスクにおいて，GAIL と提案手法の性能評価を行う．

連絡先: 黄瀬輝, 立命館大学情報理工学研究科, 滋賀県草津市野路東 1 - 1 - 1 ,akira.kinose@em.ci.ritsumei.ac.jp

(a) テレビゲーム Atari2600[Bellemare 13]

(b) ロボットによるマニピュレーション

図 1: タスクの例

2. GAIL へのタスク達成報酬の導入

2.1 強化学習と逆強化学習

一般的な強化学習では，エージェントが環境から現在の状態における行動の報酬を得ることによって最適方策を学習する．近年の強化学習の代表例として Deep Q-network [Mnih 13, Mnih 15] があり，図 1(a) のようなテレビゲームでは，ゲームのスコアを報酬に設定することで学習を行っている．また，碁やチェス，将棋などのボードゲームでは，ゲームの勝利を報酬に設定することで学習を行う．

しかし，現実のタスクにおいて，この報酬を定義することが困難な場合がある．例として，図 1(b) のようなロボットによる「ボトルの飲料をコップに移す」といったマニピュレーションタスクを考える．現実の環境では行動空間が膨大になるため，タスク達成報酬のみでは学習が難しく，タスク達成に繋がる要素を考慮した報酬が必要になる．深層強化学習によるマニピュレーションの学習に成功している Gu らの研究においても，学習にはタスク達成に繋がる要素を考慮した報酬を必要とし，タスク達成のみを報酬にした場合，学習は困難になることが述べられている [Gu 17]．タスク達成に繋がる要素とは，マニピュレーションタスクにおいて「ロボットの腕をどの位置

へ移動させ，何度傾けるのが好ましいか」などにあたる．しかし，「ロボットの腕をどの位置へ移動させ，何度傾けるのが好ましいか」を考慮した報酬を人間が明確に定義することは困難である．

このような問題に対して，逆強化学習は任意のタスクにおける最適な方策を持つエージェントであるエキスパートの行動から，人間によって定義が困難な報酬関数を推定する．そして，逆強化学習によって推定した報酬関数に従って強化学習を行うことで，報酬関数の定義が困難なタスクの最適方策を求めることができる．この LfD の手法は一般に人間が設計する特徴量が必要であり，高次元連続環境では計算量が膨大になるといった問題がある．

2.2 GAIL

Ho らが提案した GAIL は，課題点であったダイナミクスと人間による特徴量の設計が不要かつ，高次元環境でのモデルフリーの LfD の手法である．Ho らは同論文で逆強化学習に強化学習を用いた 2 段階最適化問題を 1 段階最適化問題に定式化できることを示している．その後，定式化された関数において，Goodfellow らによって提案された GAN の目的関数に一致するように正則化関数を定義することで，報酬関数を推定することなく，方策を直接推定ことができる GAIL を提案している．GAIL は式 (1) を解くことによって学習を行う．

$$\min_{\pi}\max_{D} \quad \mathbb{E}_{\pi_\theta} \quad [\log(D(s,a))] \\ + \quad \mathbb{E}_{\pi_E} \quad [\log(1-D(s,a))] - \lambda H(\pi) \qquad (1)$$

$D(s,a)$ はその状態行動のペア (s,a) がエキスパートから生成されたものである確率を示す．

GAIL モデルにおいて，Discriminator の役割は Generator によって生成された方策 π_θ とエキスパート方策 π_E を識別することである．Discriminator は勾配ベクトル:

$$\mathbb{E}_{\tau_i}[\nabla_w \log(D_w(s,a))] + \mathbb{E}_{\tau_E}[\nabla_w \log(1-D_w(s,a))]$$

に対して勾配降下法を行い，エキスパートから生成されたものか正しく識別できるように学習する．w は Discriminator のパラメータを示す．D は $D(s,a) \in [0,1]$ の値を取り，擬似的なコスト関数の役割をする．

Generator の役割はエキスパート方策に似るように方策を生成することである．Generator は勾配ベクトル:

$$\mathbb{E}_{\tau_i}[\nabla_\theta \log \pi_\theta(a|s) Q(s,a)] - \lambda \nabla_\theta H(\pi_\theta)$$
$$\text{where } Q(s,a) = \mathbb{E}_{\tau_i}[\log(D_{w_{i+1}}(s,a)|s_0 = s, a_0 = a)]$$

に対して方策勾配法を行い，Discriminator によって定義されるコストを最小化するような方策を学習する．θ は方策のパラメータを示す．この方策更新は強化学習手法 TRPO [Schulman 15] で勾配計算を行う．

2.3 タスク達成報酬を導入した GAIL

本研究では，タスク達成条件を満たした状態行動対に対して，従来手法では利用していなかったタスク達成報酬を導入することでパフォーマンスの向上を図る．

GAIL の学習則の特徴として，エキスパート軌道から推論した報酬関数のみを用いて強化学習を行うため，環境からの報酬を受け取らず，エキスパート軌道のみから学習を行っていることが挙げられる．この性質は，エージェントが自分の行った動作がタスクの達成にどのような影響を及ぼしたかがわからないという問題を引き起こす．よって，マニピュレーションタスクのように，タスクのゴールやオブジェクトの位置座標が変化するような環境下において，GAIL による学習は困難である．そこで，エキスパートだけでなく環境からも報酬を受け取ることができれば，GAIL の学習性能を上げることが期待できる．しかし，GAIL がエキスパートからの推論によって得られる報酬のみを利用している理由として，学習のための報酬関数の設計は非常に手間が掛かり，技術や知識を要することが挙げられる．以上のことから，本手法では設計に手間がかからないような報酬関数を利用することで学習を行う GAIL のモデルを考える．

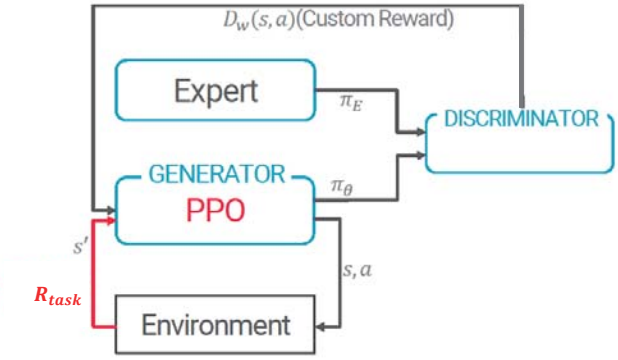

図 2: TRGAIL のモデル図

学習を行うタスクとして，本論文の研究対象としているマニピュレーションタスクを想定する．物体を指定した場所へ移動させることを目指すように学習を行うマニピュレーションタスクでの学習では，タスクのゴールが明確に存在するため，タスク達成条件を明示的に定義することが可能である．

そこで，本手法では，状態と行動を入力として，タスク達成条件を達成したかというバイナリ値を出力とする環境の報酬関数として定義する．その後，エキスパート軌道から推論される報酬とタスク達成バイナリ報酬の両方の報酬を用いることでタスク達成が可能な方策の学習を行う．ここで，バイナリ報酬とは，一般的に連続環境の複雑なタスクで用いられるような学習を円滑化させる工夫を用いた複雑な報酬とは異なり，ある状態行動対がタスクを達成しているかが 0 または 1 の値で与えられるものであるとする．

GAIL にタスク達成報酬を利用したモデルを本稿では Task Reward oriented Generative Adversarial Imitation Learning (TRGAIL) と呼称する．図 2 に TRGAIL のモデル図を示す．TRGAIL において定義される報酬関数の式を以下に示す．

$$Q(s,a) = \mathbb{E}_{\tau_i}[\alpha \log(D_{w_{i+1}}(s,a) + (1-\alpha) R_{task}(s,a)] \qquad (2)$$

ここで，α は報酬関数においてタスク達成報酬が報酬関数を占める割合を決めるパラメータである．本稿では $\alpha = 0.5$ を採用している．$R_{task}(s,a)$ はバイナリ値で与えられるタスク達成報酬を示す．GAIL において $D(s,a)$ は報酬ではないため，ここにタスク達成報酬を加えることはヒューリスティクスなアプローチであると言える．また，TRGAIL では Generator に強化学習手法である PPO[Schulman 17] を用いる．TRGAIL のアルゴリズムを Algotithm 1 に示す．

TRGAIL によるエキスパート軌道の学習では，タスク達成に近づくが完全なタスク達成までに至らない場合，タスク達成

Algorithm 1 TRGAIL

1: **Input:** エキスパート軌道 $\tau_E \sim \pi_E$, 方策と Discriminator の初期パラメータ θ_0, w_0
2: **for** $i = 0, 1, 2, \ldots$ **do**
3: 学習軌道をサンプリング $\tau_i \sim \pi_{\theta_i}$
4: Discriminator の勾配ベクトル:

$$\mathbb{E}_{\tau_i}[\nabla_w \log(D_w(s,a))] + \mathbb{E}_{\tau_E}[\nabla_w \log(1 - D_w(s,a))]$$

に対して勾配降下法を行うことでパラメータを w_i から w_{i+1} へ更新
5: Generator の勾配ベクトル:

$$\mathbb{E}_{\tau_i}[\nabla_\theta \log(\min(r_t(\theta), \mathrm{clip}(r_t(\theta), 1+\epsilon, 1-\epsilon))\, Q(s,a))] - \lambda \nabla_\theta H(\pi_\theta)$$

$$Q(s,a) = \mathbb{E}_{\tau_i}[\alpha \log(D_{w_{i+1}}(s,a) + (1-\alpha) R_{task}(s,a)]$$

に対して PPO の方策勾配法を行うことでパラメータを θ_i から θ_{i+1} へ更新
6: **end for**

報酬を環境から得ることによって，エージェントがエキスパート軌道の学習とタスクの達成軌道の学習を両立できると考えられる．

このアプローチを強化学習の観点から考えると，タスクのエキスパートによる手本を得ることで，膨大な探索範囲を絞り込めるという見方をすることができる．また，タスクのエキスパートとタスク達成報酬を準備する代わりに，従来の強化学習の課題点であったタスク達成に繋がる要素を考慮した複雑な報酬関数の設計が不要になるという利点が考えられる．

3. 実験

3.1 実験目的

本実験では，エキスパートの軌道からマニピュレーション動作の獲得をする実験を行う．本実験の目的は，既存の GAIL モデルとタスク報酬を導入した GAIL のモデルのタスクのパフォーマンスを比較し，その妥当性を検証することである．比較手法として，既存の GAIL，提案手法，LfD 手法のベースラインとして Behavior Cloning を用いる．

3.2 実験条件

実験環境は物理エンジンソフトウェアである MuJoCo [Todorov 12] を用いたシミュレーション環境で実験を行う．各タスクは OpenAI Gym [Brockman 16] によって定義されたモデルの状態空間，行動空間を利用して行う．本実験では，Pusher，Striker，Thrower の 3 タスクについてマニピュレーション動作獲得の実験を行う．それぞれのタスクの図を図 3，図 4，図 5 に示す．

各タスクにおいて，1 回の試行を合計 100 タイムステップとする．イテレーション数は 3000 であり，各イテレーションのタイムステップ数が 2400 になるようにバッチをサンプリングした．ベースラインとして，同タスクで Behavior Cloning で 10000 イテレーションの学習を行ったものを利用した．実験は 3 タスク，3 手法に対してシード値を変えてそれぞれ 3 試行ずつ行った．エキスパート軌道は実験条件の簡易化のため，連続環境における state-of-the-arts の手法の一つである強化学習手法の PPO によって学習を行った最適方策による軌道を用いた．PPO による強化学習は人による複雑なタスク達成に繋がる要素を考慮した報酬関数の設計を行い，最適軌道から 500 軌道をサンプリングした．

それぞれのタスクにおいて，行動空間は 7 次元の連続値でアームの各 7 関節に与えるトルクの値を示す．状態空間は 23 次元の連続値で 7 関節の角度角速度，オブジェクトゴール手先の xyz 座標の値を示す．以下に各タスクの設定について述べる．

Pusher

Pusher タスクは，エージェントの手先が U の字の型をしたアームを用いて白い円柱のオブジェクトをゴールである赤い円内へ押し込むように移動させるタスクである．環境の初期状態はエージェントの姿勢とゴールの座標が固定値で，オブジェクトの座標がランダム値で決まる．タスク達成条件はオブジェクトの中心の xy 座標がゴール範囲内に入ることとする．

Striker

Striker タスクは，エージェントの手先が T の字の型をしたアームを用いて，白いボールをテーブル上のゴールである離れた白い柵付きの円内へ打ち込むタスクである．環境の初期状態はエージェントの姿勢とオブジェクトの座標が固定値で，ゴールの座標がランダム値で決まる．タスク達成条件はオブジェクトの中心の xy 座標がゴール範囲内に入ることとする．

Thrower

Thrower タスクは，エージェントのアームを用いて白いボールを離れたテーブル上の四角の箱のゴールへ投げ入れるタスクである．環境の初期状態はエージェントの姿勢とオブジェクトの座標が固定値で，ゴールの座標がランダム値で決まる．タスク達成条件はオブジェクトの中心の xy 座標がゴール範囲内に入り，ボールが箱の底面に接触していることとする．

3.3 評価指標

軌道スコア

各試行の最終タイムステップにおける状態に対する評価を行い，スコアを算出する．この評価指標は時間的要因を考慮しないため，学習した軌道の試行における最終的な状態に対して評価を行う．よって，この評価値からは学習した方策が各試行においてタスク達成に近づいたかがわかる．Pusher タスクは最終タイムステップのオブジェクトの中心座標とゴール範囲の距離，Striker タスクは最終タイムステップのオブジェクトの中心座標とゴール範囲の距離，Thrower は最終タイムステップまでに最も近づいたオブジェクトの中心座標とゴール範囲の距離を軌道スコアと定義する．

タスク成功率

この評価指標は，各試行の最終タイムステップにおける状態を参照し，その時点でタスクの達成条件を満たしているかの判定を行い，そのイテレーションにおけるタスク成功率を示す．よって，この評価値が高い場合，タスクの達成条件を正確に満たす方策を学習できたということがわかる．

3.4 実験結果

実験結果を表 1 に示す．TRGAIL の実験結果について，この手法による評価指標は全手法において総合的に最も良いパフォーマンスを示した．実験結果の軌道と成功率を確認すると，従来までの手法と比較して，エキスパート軌道の学習に加えてタスクのゴールを目指すような方策を学習していることが確認できた．また，Pusher，Thrower の軌道スコア，成功率において，エキスパートのスコアと同等のパフォーマンスを示していることがわかる．

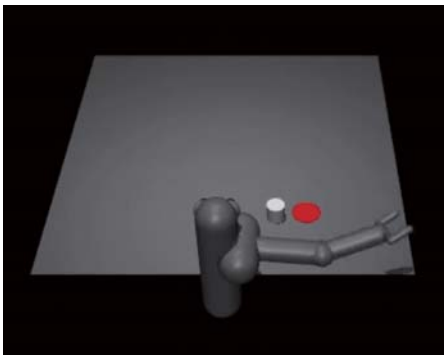

図 3: Pusher タスク

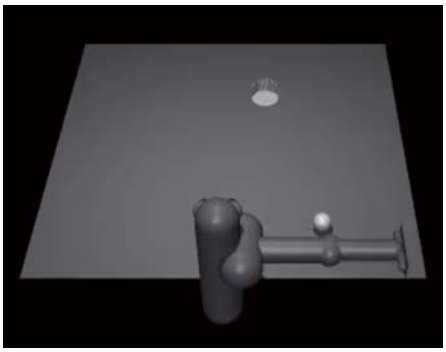

図 4: Striker タスク

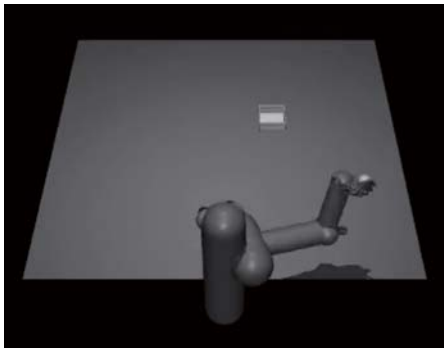

図 5: Thrower タスク

表 1: 実験結果

	Pusher		Striker		Thrower	
手法	軌道スコア	成功率	軌道スコア	成功率	軌道スコア	成功率
Expert Policy	-0.0006	93.2	-0.0095	87.1	-0.0077	79.3
Random Policy	-0.1675	0.5	-0.6365	0.0	-0.8586	0.2
Behavior Cloning	-0.1418	25.0	-0.1346	33.0	-0.0092	81.0
GAIL	-0.1975	9.4	-0.6466	6.9	-0.0535	28.0
TRGAIL	**-0.0004**	**92.3**	**-0.0270**	**79.7**	**-0.0082**	**91.6**

4. おわりに

本稿では，GAIL にタスク達成報酬の導入を行い，タスクのゴールが存在するシミュレーション上のマニピュレーションタスクにおいて，エキスパート軌道からタスク達成動作の獲得を行う TRGAIL を提案した．実験結果より，提案手法がタスクのゴールが存在するマニピュレーションにおいて高いパフォーマンスを示す方策を学習したことが示された．

GAIL はエキスパートとなる軌道と似たような軌道を生成する方策を学習するという問題設定だが，タスク達成までの過程がより複雑になるようなタスクの学習は困難であると考えられる．本研究の今後の課題として，階層型強化学習の理論を適応させることで，タスクを階層化し，階層的な報酬関数の推定を行うことで，より複雑なタスクの学習を行うことが期待される．

参考文献

[Bellemare 13] Bellemare, Marc G., et al. "The Arcade Learning Environment: An evaluation platform for general agents." J. Artif. Intell. Res.(JAIR) 47 (2013): 253-279.

[Brockman 16] Brockman, Greg, et al. "Openai gym." arXiv preprint arXiv:1606.01540 (2016).

[Goodfellow 14] Goodfellow, Ian, et al. "Generative adversarial nets." Advances in neural information processing systems. 2014.

[Gu 17] 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.

[Ho 16] Ho, Jonathan, and Stefano Ermon. "Generative adversarial imitation learning." Advances in Neural Information Processing Systems. 2016.

[Mnih 13] Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." in Deep Learning, Neural Information Processing Systems Workshop, 2013.

[Mnih 15] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." Nature 518.7540 (2015): 529.

[Schulman 15] Schulman, John, et al. "Trust region policy optimization." International Conference on Machine Learning. 2015.

[Schulman 17] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint arXiv:1707.06347 (2017).

[Todorov 12] Todorov, Emanuel, Tom Erez, and Yuval Tassa. "Mujoco: A physics engine for model-based control." Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. IEEE, 2012.