動的環境における学習ベースおよびルールベースの切り替え手法を用いた移動ロボットナビゲーション
-第三報 学習ベースおよび切り替えモデルの追加学習の検討-

○兵頭侑樹（九州大学），松本耕平（九州大学），富田湧（九州大学），倉爪亮（九州大学）

Mobile Robot Navigation with Switching Learning-Based and Rule-Based Method in Dynamic Environments
-Incremental learning of learning-based model and swiching model-

○ Yuki Hyodo (Kyushu University), Kohei Matsumoto (Kyushu University),

Yuki Tomita (Kyushu University), and Ryo Kurazume (Kyushu University)

Abstract: In recent years, the demand for autonomous mobile robots has been increasing, and autonomous robot navigation is essential for realizing social robot navigation especially in dynamic environments with pedestrian. We have proposed a navigation method that switches between learning-based and rule-based methods using a switching model based on normalizing flows, and have conducted experiments in simulation and real environments. In this paper, we investigate incremental learning of the learning-based model and switching model to further improve the performance of mobile robot navigation using the switching model. In particular, we verify whether adding data from situations where switching has occurred is effective in improving performance.

1. 緒言

人々が生活する日常環境にロボットを安全に導入するには，歩行者が行き交う混雑した環境でもスムーズに移動できる移動ロボットナビゲーション手法の実現が不可欠である．我々は第一報[1]で，このような環境で移動ロボットのナビゲーションを行うために，学習ベースの行動とルールベースの行動を切り替える手法を提案し，学習ベースが学習していない環境に対してルールベースへの切り替えを行うことで，安全かつ短時間で目的地に到達できることをシミュレーション環境上で確認した．具体的には，限られたデータセットで強化学習させた学習ベース手法と衝突回避アルゴリズムからなるルールベース手法を，Switching Administrator と呼ぶグラフ正規化フロー[2]を用いた切り替えモデルによる尤度推定を基準に切り替える手法を提案した．さらに第二報[3]で，実環境で実験を行うための実機実装を行い，実験を通して切り替え手法の実環境への適用可能性を示した．

本稿では先行研究の学習モデルに対して追加学習を行い，実環境で歩行者を含むような複雑な環境に対して，より柔軟に適応できる学習モデルの構築を目指す．この目的を達成するための前段階として，第一報で提案した手法に含まれる深層強化学習モデルと Switching Administrator を追加学習することによってナビゲーション性能が向上するかをシミュレーションにおいて検証する．図 1 に本稿での追加学習の概要を示す．まず学習済みの深層強化学習モデルと Switching Administrator を用いて，適用させたい環境で切り替え手法によりデータを収集する．そして，このデータセットを用いて学習モデルを追加学習する．本稿では追加学習方法に加えて追加学習する際に用いるデータにも注目する．Switching Administrator がルールベースへの切り替えを行った際のデータは，深層強化学習モデルがまだ学習ができていないデータとみなすことができる．このデータのみを使って学習モデルの性能を向上をさせることができれば，追加するデータ量が減少し，効率的に性能を向上させることが可能になると考えられる．今回用いる方法によって，ルールベースへの切り替えが発生したデータを効率的に用いて追加学習できることを示す．

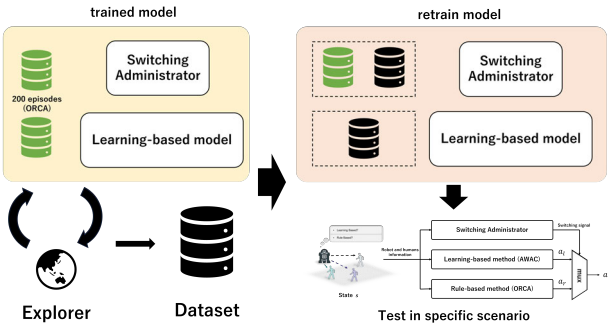

Fig. 1 Overview of incremental learning method.

2. 深層強化学習モデルの追加学習

図 2 に第一報の提案手法における深層強化学習手法のアーキテクチャを示す．学習手法は AWAC[4] を用いており，アドバンテージ関数を用いた方策の重み付き回帰学習を行うことにより，学習データセットに近い方策を学習することができる．ここで，アドバンテージ関数 $A(s,a)$ はある状態 s で行動 a を選んだときの価値が，その状態で期待される価値よりもどれだけ良いかを示す指標であり，本稿では以下の式で表される．

$$A(s,a) = \max\left(0, \min_{i=1,2} Q_{\phi^i}(\boldsymbol{s},\boldsymbol{a}) - \min_{i=1,2} Q_{\phi^i}(\boldsymbol{s},\pi_\theta(\cdot \mid \boldsymbol{s}))\right). \quad (1)$$

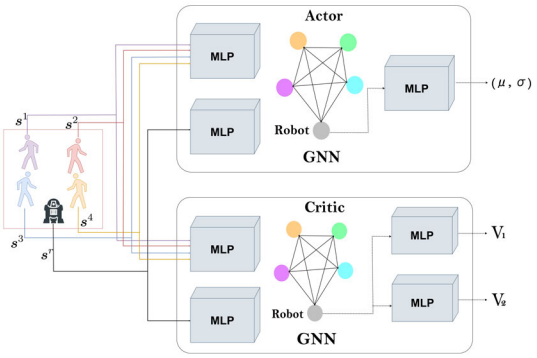

Fig. 2 Architecture of actor-critic part of learning-based method.

Q_{ϕ^i} は min-Double-Q の Q 関数を表しており，π_θ は Actor の確率的方策を表している．そして Actor は以下の損失関数を最小化するように学習を行う．

$$L_{\text{actor}} = -\log \pi_\theta(\boldsymbol{s}, \boldsymbol{a}) \exp\left(\frac{1}{\lambda} A\right) \tag{2}$$

ここで, λ はハイパーパラメータとする．

本稿ではこの深層強化学習モデルを追加学習するために，アドバンテージ関数による優先度に基づいたサンプリングを行いながら追加学習を行う．実環境でのデータをサンプリングして追加学習を行う場合，データが十分に集まらないことが懸念される．一方, 提案手法はサンプルしたデータの中でも良いデータを効率的に用いることができる．本手法は ODPR[5] に基づいており，オフライン強化学習の学習方法に対して，用意したデータセット内の各データの優先度を更新しながら学習を行うことで，良いデータを効率的に活用し，学習を進めることができることが示されている．データサンプル時の優先度を表す重み w は方策 π_θ に基づいたアドバンテージ関数 A^{π_θ} 用いて以下の式で更新される．

$$w(A^{\pi_\theta}) = C(A^{\pi_\theta}(s,a) - \min_{(s,a)\in\mathcal{D}} A^{\pi_\theta}(s,a)) \tag{3}$$

C は定数であり，データセット内での重みの合計が 1 となるように設定される．アドバンテージ関数の最小値を引くのは，データセット内での優先度は正であるという条件に基づいているためである．

3. 切り替えモデルの追加学習

切り替えモデルは歩行者の位置，速度情報を入力とし，グラフ正規化フローに基づいて，学習データセットに対する入力データの尤度推定を行う．切り替えモデルは第一報における深層強化学習モデルと同じデータセットの歩行者情報を学習しているため，深層強化学習モデルが学習したデータにどれだけ近いかを切り替えモデルによる尤度推定で判断している．本稿ではこの切り替えモデルの追加学習方法としては，学習ベース手法と同じデータセットで学習のやり直しを行う．これにより，追加学習を行った学習ベース手法が学習したデータセットに対する尤度推定を行うことができる．

4. 追加学習のアルゴリズム

Algorithm1 にモデルの追加学習のアルゴリズムを示す．まず学習済みの深層強化学習モデルと切り替えモデルを用意し，適用させたい環境で探索を行う．この時学習済みモデルは square-crossing にて 200 エピソード分のデータセット $\mathcal{D}$ で学習を行っており，ロボットと歩行者はルールベース手法で行動している．探索時に得た n エピソード分のデータのうち，ルールベースへの切り替えが発生したデータをデータセット $\mathcal{D}'$ に格納する．そして，元の学習データセットと新しく得たデータセットを用いて，Switching Administrator の追加学習を行う．次にデータセット $\mathcal{D}'$ の優先度をすべて 1 に更新する．そして深層強化学習モデルに対して，データセット $\mathcal{D}'$ から優先度に基づいてバッチをサンプルし，このデータで，学習モデルの追加学習を行う．その後，データセット $\mathcal{D}'$ 内のデータすべてに対する優先度を計算し，データセット内の優先度を更新する．データのサンプルから重みの更新までを複数回行い，データセット内の価値が高いデータの重みを大きくしながら，追加学習を行う．

Algorithm 1: Incremental learning algorithm for learning-based model and switching model

Prepare learned policy π_θ and switching model with a dataset $\mathcal{D}$ including 200 episodes of data in square-crossing scenario using ORCA
Explore new environment using switching method and store n episodes of switched data in dataset $\mathcal{D}'$
Retrain switching model with dataset $\mathcal{D} \cup \mathcal{D}$‘
Initialize priorities $w_i = 1$
for $i = 1$ **to** K **do**
    Sample a batch $\mathcal{B} = \{(\boldsymbol{s}, \boldsymbol{a}, r, \boldsymbol{s}')\}$ from the dataset $\mathcal{D}'$ with priority w^i
    Update policy π_θ
    Evaluate A^i of behavior policy π^i for all data in dataset $\mathcal{D}$‘
    $w^i = w^i A^i(s_i, a_i)$
end

5. シミュレーション実験

本稿で示した追加学習方法で，ルールベースへの切り替えが発生したデータを用いた追加学習により，ナビゲーションの性能が向上することを検証するためにシミュレーションにて実験を行った．実験環境は第一報と同じく CrowdNav[6), 7)] を用いる．追加学習に用いるデータセットは，切り替え手法を用いて circle-crossing シナリオで 10 エピソード分探索した際に収集した成功データのうち，ルールベースへの切り替えが発生した際のデータである．重み更新の繰り返し回数は 5 回とする．本実験におけるすべての学習，テストで歩行者の人数は 5 人である．

5.1 深層強化学習モデルの性能評価

まず，追加学習により深層強化学習モデルのみを用いたナビゲーションの性能が向上するかを数値評価と定性評価で確認する．表 1 は追加学習前と追加学習後の深層強化学習モデルのみのナビゲーションの成功率，衝突率，平均到達時間を示している．追加学習後の中で，切り替えが発生したデータのみを用いている場合を SO

(Switching data Only)，ODPR を用いている場合を ODPR と表している．この結果から，まず深層強化学習モデルを探索時の成功データを用いて追加学習させることで，ナビゲーションの性能が向上することが分かった．そして，成功データのうちルールベースへの切り替えが発生したデータのみを追加して追加学習を行うと，さらにナビゲーションの性能が向上し，加えて，アドバンテージ関数に基づいた優先度付き経験再生を用いることで，成功率は追加学習前後で 63.8%から 94.4%に，平均到達時間は追加学習前後で 8.70s から 8.14 秒まで向上した．これは探索時の成功データのうちルールベースへの切り替えが発生したデータは，もともと学習ベースが学習できていないデータとみなすことができ，このデータのうち価値が高いものを優先的に学習させることで，成功率と平均到達時間の両方の結果を向上させることができたと考えられる．次に図 3 に追加学習前と追加学習後（SO + ODPR）に関して，ロボットと歩行者の軌跡の例を示す．ロボットの軌跡は黒で示している．この結果から，追加学習前で対応衝突が発生していたエピソードについて，追加学習によりナビゲーションに成功していることを確認することができた．

Table 1 Numerical comparison in circle-crossing using only RL methods

Method	Success [%]	Collision [%]	Exec. time [s]
追加学習前	63.8	36.2	8.70
追加学習	79.2	20.8	8.65
追加学習+SO	90.6	9.4	8.63
追加学習+SO+ODPR	94.4	5.6	8.14

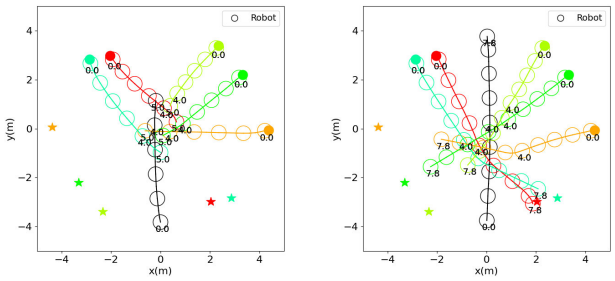

Fig. 3 Comparison of trajectories before and after incremental learning of deep reinforcement learning model

5.2 切り替えモデルを含めた全体の性能評価

ここでは追加学習後の切り替えモデルと Switching Administrator を用いた切り替え手法により，ナビゲーションの性能が向上するかどうかを数値評価と定性評価で確認する．表 2 は本研究における切り替え手法において，追加学習前と追加学習後の 2 つについて成功率，衝突率，平均到達時間，ルールベースへの切り替えが発生した割合を示したものである．深層強化学習モデルはルールベースへの切り替えが発生したデータを優先度付き経験再生を用いて追加学習したモデルを用いている．結果は成功率は追加学習前後で 97.8%から 96.4%に下がったものの，平均到達時間は 8.84s から 8.30s となり，ルールベースへの切り替え割合は 60.3%から 33.9%になった．このことから，学習モデルの追加学習を行うと，Switching Administrator が追加学習前より学習ベース手法を多く選択したうえで，追加学習前と同程度の成功率を維持し，より早く目的地に到達できるようになった．成功率が下がった原因としては，深層強化学習モデルは追加学習により，平均到達時間が早くなっており，直進しながら目的地を目指す行動をしやすくなっていることが挙げられる．ルールベースへの切り替えにより，ロボットの方向が変わったときに学習ベース手法で対応できるとは限らず，そのような場面で学習ベース手法の行動を選択して衝突が発生したと考えられる．

また図 4 に追加学習後の切り替え手法について，それぞれの切り替えの様子の違いを示す．図において，ルールベースへの切り替えが発生した箇所は色が濃く示されている．また図内に示されている数値はその位置におけるロボットの到達時間を示している．この結果から，追加学習前に切り替えが必要だった場面で，学習ベース手法を選択し，追加学習前より早く目的地に到達していることが分かり，本稿の追加学習がナビゲーション性能向上に有効であることを示している．

Table 2 Numerical comparison in circle-crossing with switching method

Method	Success [%]	Collision [%]	Exec. time [s]	Switching rate
追加学習前	97.8	2.2	8.84	60.3
追加学習後	96.4	3.6	8.30	33.9

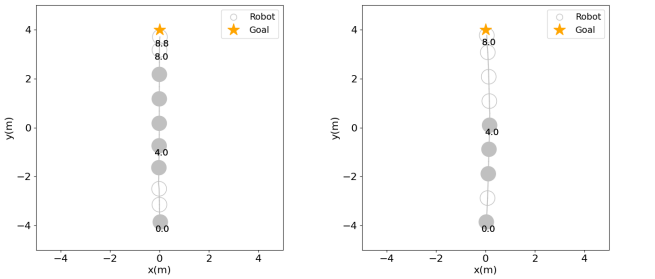

Fig. 4 Comparison of the timing of switching between before and after Incremental learning

5.3 同じ切り替え割合での比較

5.2 節の実験結果から，深層強化学習モデルと Switching Administrator の両方のモデルを追加学習した際の切り替え割合が 33.9%であることが分かった．追加学習後の切り替えモデルによる切り替えのタイミングが適切であるかどうかを確認するために，追加学習せずに切り替え割合を 33.9%に固定した場合，追加学習後に 33.9%の割合でランダムにルールベースへ切り替える手法，追加学習後の 3 つの手法で比較を行う．表 3 に上記 3 つの手法について成功率，衝突率，平均到達時間を示している．ランダムに切り替える手法の結果は 100 個のシード値の平均値である．この結果から追加学習前は成功率が 97.8%から 81.4%まで減少しており，同じ切り替え割合では，追加学習後の方が，適切に対応できていることがわかる．またランダムに切り替える方法と追加学習後の性能を比較すると，追加学習後の方が成功率が高いが，大きな差はみられなかった．これは学習ベース手法のナビゲーション成功率が高くなったことにより，ランダムに切り替えたとしても学習ベース手法とルールベース手法の両方がその場の状況に対応できるようになったためであると考えられる．今後，追加学習に限らず，学習ベース手法の学習方

法や, 切り替えモデルの別の追加学習手法を検討する必要があると考えられる.

Table 3 Numerical comparison between before incremental training and the method of random switching

Method	Success [%]	Collision [%]	Exec. time [s]
追加学習前	81.4	18.6	8.66
ランダム	96.0	4.0	8.22
追加学習後	96.4	3.6	8.30

6. 結言

本稿では，第一報で提案した学習ベースとルールベースの切り替え手法における，深層強化学習モデルと切り替えモデルの追加学習手法を検討した．2 つの学習モデルに対して，適応させたいシナリオの成功データを追加学習させることで，ナビゲーションの性能が向上することが分かった．またルールベースへの切り替えが発生した箇所のデータをアドバンテージ関数による優先度付き経験再生を用いて追加学習させることで，深層強化学習モデルによるナビゲーションの性能向上を達成することができた．今後は第二報で行った実機実験で収集したデータを利用して，実環境に適用できるように更なる追加学習手法を検討し，実験を行っていきたい．また，本稿で提案した切り替えモデルの追加学習方法は，学習のやり直しに相当し，時間的，計算資源的な側面からコストが大きいため，別の正規化フローの再学習手法を適用させる必要がある．具体的には, [8] で提案されている手法を用いるなどして，学習モデルの学習済みデータの忘却を防ぎつつ，適応させたいシナリオのデータを追加学習できるような手法を検討する．また，学習ベース手法の方策を考慮しつつ，切り替えモデルと同時に学習させていくことで，新しい状況に対してさらに柔軟に対応できるような学習方法も検討していきたい.

謝辞

本研究の一部は，JSPS 科研費 JP20H00230 の助成を受けたものである.

参考文献

[1] 兵頭侑樹，松本耕平，倉爪亮.「動的環境における学習ベースおよびルールベースの切り替え手法を用いた移動ロボットナビゲーション」. 第 *24* 回計測自動制御学会システムインテグレーション部門講演会 *(SI)*. (2023), pp. 275–278.

[2] J. Liu et al. Graph Normalizing Flows. *Advances in Neural Information Processing Systems (NeurIPS)*. (2019), pp. 13556–13566.

[3] 兵頭侑樹，松本耕平，富田湧，倉爪亮.「動的環境における学習ベースおよびルールベースの切り替え手法を用いた移動ロボットナビゲーション 第二報 実機実装と実環境での動作実験」. 第 *42* 回日本ロボット学会学術講演会. (2024).

[4] A. Nair et al. AWAC: Accelerating online reinforcement learning with offline datasets. arXiv preprint arXiv:2006.09359, (2020).

[5] Y. Yue et al. Decoupled Prioritized Resampling for Offline RL. arXiv preprint arXiv:2306.05412, (2023).

[6] C. Chen et al. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. (2019), pp. 6015–6022. DOI: 10.1109/ICRA.2019.8794134. arXiv: 1809.08835.

[7] Y. Chen et al. Robot Navigation in Crowds by Graph Convolutional Networks with Attention Learned from Human Gaze. IEEE Robotics and Automation Letters 5.2, pp. 2754–2761, (2020). DOI: 10.1109/LRA.2020.2972868. arXiv: 1909.10400.

[8] S. Malnick, S. Avidan, and O. Fried. Taming Normalizing Flows. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. (2024), pp. 4644–4654.