歩行者混雑環境下におけるメタ強化学習を用いた移動ロボットナビゲーション
-第二報 Meta-Critic を利用した学習手法の検討と未学習環境での評価実験-

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

Mobile Robot Navigation in Crowded Environments with Pedestrians Using Meta Reinforcement Learning
-Investigation of a Meta-Critic Learning Approach and Evaluations in Unseen Environments-

○ Yuki Hyodo (Kyushu University), Kohei Matsumoto (Kyushu University),
Yuki Tomita (Kyushu University), Haruto Nagahisa (Kyushu University),
and Ryo Kurazume (Kyushu University)

Abstract: With the growing demand for mobile robots, ensuring their safe and efficient operation in dynamic environments that include pedestrians has become a critical challenge. Recent approaches using deep reinforcement learning have shown promise; however, a major limitation is the degradation of performance in previously unseen environments. Therefore, rapid adaptation to such situations is required. In this paper, we investigate whether meta-reinforcement learning enables performance improvement with only a small number of trials in unseen scenarios. Furthermore, we examine the effectiveness of employing our proposed method, which dynamically switches between learning-based and rule-based strategies, as a data collection mechanism. Specifically, we evaluate whether utilizing the data collected when switching to the rule-based strategy as unseen data for the learning-based model contributes to improved adaptability.

1. 緒言

今日の移動ロボットの需要に伴い, 歩行者を含む動的環境で移動ロボットを安全にかつ効率的に動作させることが重要な課題となっている. 特に歩行者混雑環境はロボット周囲の状況が動的に変化するため，ロボットは直面する状況に対して，適応的に行動する必要がある．近年では，深層強化学習を用いた手法が盛んに研究されており，学習した状況に対して安全でかつ効率的な動作が可能となっている．しかし，学習済みのモデルを未学習の環境に適用すると著しく性能が低下するという問題がある．この問題に対して，未学習の新しい環境への迅速な適応を目的としたメタ強化学習を用いたアプローチを考える．このアプローチによって，学習時と異なるシナリオでテストする場合に少数回数のオンライン更新で性能を向上させ，その環境に適応していくことを目指す．我々は第一報にて，メタ強化学習手法のひとつである MAML[1] に基づいた手法を提案し，メタ強化学習手法が強化学習のみで学習した手法に比べて，人数変化に対して汎用的であるという結果を得た．しかし，この手法では少数回数のオンラインファインチューニングによる性能向上は確認できなかった．

そこで本稿では，強化学習ベース手法の損失関数に未学習の状況の学習を促す損失関数である Meta-critic[2] を追加することで，新しい状況のデータに迅速に適応するメタ強化学習を行う手法を提案する．この損失関数は学習環境と異なる状況での学習方法を学習し，ロボットが迅速に未学習環境に適応していくことが期待される．本稿では，学習時と異なる歩行者の動作モデルを用いた環境で新たにデータを収集し，少数回数で性能向上が達成できるかをシミュレーションにて検証する．また，我々が過去に提案した学習ベースとルールベースを動的に切り替える手法[3](以下，切り替え手法という) をデータ収集手法として利用し, ルールベースに切り替わったデータを, 学習ベースが未学習のデータとして活用することの有効性についても検証する．

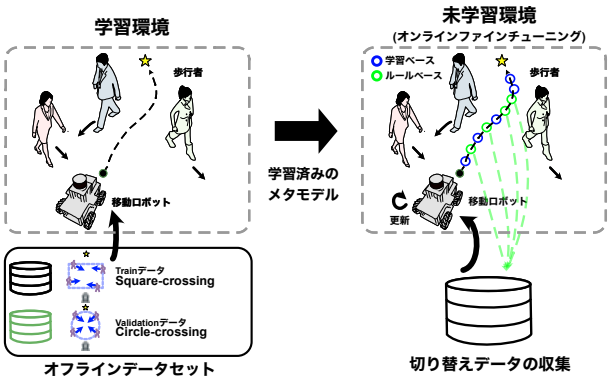


Fig. 1 Overview of meta-critic based learning method

2. 提案手法

2.1 モデルアーキテクチャ

本手法の概要図を図 2 に示す．本手法は，メタ強化学習手法の一つである，Meta-critic[2] に基づいている．また，あらかじめロボットと歩行者の軌跡のデータセットとして，$d_{\text{trn}}, d_{\text{val}}$ を用意し，オフラインのメタ強化学習手法としている．学習対象のモデルは Actor, Critic, Meta-critic の 3 つである．Actor では，入力である歩行者の位置・速度情報が Multi Layer Perceptron (MLP) と Graph Neural Network (GNN)[4] によりエンコードされ，最終的に行動の確率分布が出力される．Critic では，歩行者の位置，速度情報に加えて，Actor からサンプルされた行動を入力とし，Actor と同様に MLP と GNN のエンコーディングにより，行動に対する価値を出力する．Meta-critic は MLP で構築されており，歩行者の位置・速度情報に加えて，方策の内部表現とその方策から出力される行動を全て結合したデータを入力とする．Meta-critic で使用するデータセットは d_{val} とし，Actor と Critic で用いる d_{trn} とは異なる．これは Meta-critic が学習データと異なるデータに対しての評価を行う役割を持つためである．また，このモデルの出力は Actor に追加する損失関数そのものである．この損失関数が，未学習状況での性能向上を促進するように学習を行う．損失関数の詳細は次節で説明する．

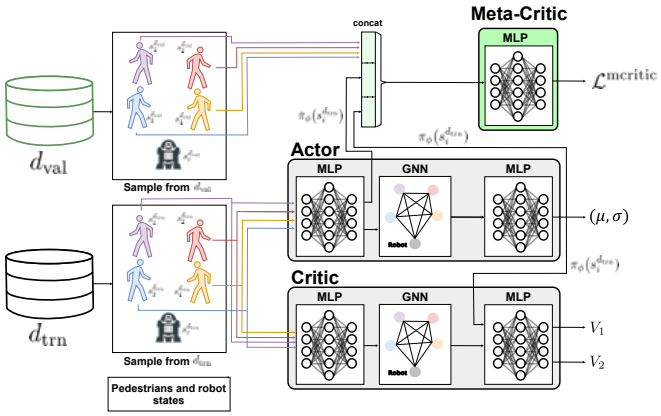


Fig. 2 Architecture of proposed method

2.2 学習アルゴリズム

次は学習アルゴリズムについてである．Algorithm 1 に提案手法のアルゴリズムを示す．学習は大きく 4 つのフェーズに分かれており，Critic の学習，メタ訓練，メタテスト，メタ最適化である．まずは Critic の学習である．Critic は学習の安定化を目的として，2 つの min-Double-Q の Q 関数 Q_{θ^1}，Q_{θ^2} とターゲットネットワーク y を持つ．ターゲットネットワークとモデルパラメータの更新は以下の式で表す．

$$y = r(s,a) + \gamma \mathbb{E}_{s',a'}\left[\min_{i=1,2} Q_{\theta^i}(s',a')\right] \tag{1}$$

$$\theta \leftarrow \arg\min_{\theta} \mathbb{E}_{d_{\text{trn}}}\left[(Q_\theta(s,a) - y)^2\right] \tag{2}$$

ここで，s, a は環境の状態とその状態におけるロボットの行動，s', a' は環境の次状態とその状態におけるロボットの行動を表す．次はメタ訓練である．ここでは Actor の更新のための損失関数を計算し，Actor の 2 段階のパラメータ仮更新を行う．Actor の損失関数は 2 つの要素の和からなり，一つはオフライン強化学習の手法の一つある，AWAC[5] による損失関数である． AWAC は学習方策を強化学習におけるアドバンテージ関数で方策の重み付き回帰を行うことで，オフラインのデータセットに近い行動を学習しやすい学習効率の良さがあるため採用している．他方は，Meta-critic が直接出力する損失関数である．ぞれぞれの損失関数は以下の式で表される．

$$L_{\text{actor}} = -\log \pi_\phi(\boldsymbol{s}, \boldsymbol{a}) \exp\left(\frac{1}{\lambda} W\right). \tag{3}$$

$$L_{\text{mcritic}}(d_{\text{trn}}; \phi) = \frac{1}{N} \sum_{i=1}^{N} f_\omega(\bar{\pi}_\phi(s_i), s_i, a_i). \tag{4}$$

ここで，L_{actor} における π_ϕ は，パラメータ ϕ を持つ Actor の方策であり，λ はハイパーパラメータである．また重み W は以下の式で表される．

$$W = \max\left(0, \min_{i=1,2} Q_{\theta'^i}(\boldsymbol{s}, \boldsymbol{a}) - \min_{i=1,2} Q_{\theta'^i}\left(\boldsymbol{s}, \pi_\phi(\cdot \mid \boldsymbol{s})\right)\right). \tag{5}$$

加えて $L_\omega^{\text{mcritic}}$ における $\bar{\pi}_\phi(s_i)$ は Actor の特徴抽出後の内部表現であり，Meta-critic のモデルに入力することで，Meta-critic は Actor による方策がどのように変化しているのかを確認できる．

Algorithm 1: Meta reinforcement learning algorithm using meta-critic

Input offline dataset $d_{\text{trn}}, d_{\text{val}}$
Initialize ϕ, θ, ω
for *each iteration* **do**
  for *each step* **do**
    Sample mini-batch d_{trn}^B from d_{trn}
    $y = r(s,a) + \gamma \mathbb{E}_{s',a'}[\min_{i=1,2} Q_{\theta^i}(s',a')]$
    $\theta \leftarrow \arg\min_\theta \mathbb{E}_{d_{\text{trn}}^B}\left[(Q_\theta(s,a) - y)^2\right]$
    $L_{\text{actor}} = \log \pi_\phi(a|s) \exp\left(\frac{1}{\lambda} W\right)$
    $L_{\text{mcritic}}(d_{\text{trn}}^B; \phi) \leftarrow \frac{1}{N} \sum_{i=1}^{N} f_\omega(\bar{\pi_\phi}(s_i), s_i, a_i)$
    $\phi_{\text{old}} = \phi - \eta \nabla_\phi L_{\text{actor}}$
    $\phi_{\text{new}} = \phi_{\text{old}} - \eta \nabla_\phi L_{\text{mcritic}}$
    Sample mini-batch d_{val}^B from d_{val}
    $L_{\text{meta}}(d_{\text{val}}^B; \phi_{\text{old}}, \phi_{\text{new}}) =$
      $\tanh\left(L_{\text{actor}}(d_{\text{val}}^B; \phi_{\text{new}}) - L_{\text{actor}}(d_{\text{val}}^B; \phi_{\text{old}})\right)$
    $\phi \leftarrow \phi - \eta(\nabla_\phi L_{\text{actor}} + \nabla_\phi L_{\text{mcritic}})$
    $\omega \leftarrow \omega - \eta \nabla_\omega L_{\text{meta}}$
  end
end

また L_{mcritic} において，f_ω はパラメータ ω を持つ，Meta-critic の学習モデルによる関数を表している．そしてここで，これらの損失関数を用いて，Actor のパラメータの 2 段階の仮更新を行う．仮更新の更新式は以下のよ

うに表される．

$$\phi_{\text{old}} = \phi - \eta \nabla_{\phi} L_{\text{actor}} \tag{6}$$

$$\phi_{\text{new}} = \phi_{\text{old}} - \eta \nabla_{\phi} L_{\text{mcritic}}. \tag{7}$$

この仮更新結果は次のメタテストで利用される．次にメタテストである．ここではメタ訓練で得た Actor の 2 段階の仮更新の結果得られたパラメータに基づいて，d_{val} に対して，その更新の良さを評価する．更新式は以下のように表される．

$$L_{\text{meta}} = \tanh\left(L_{\text{actor}}(d_{\text{val}}; \phi_{new}) - L_{\text{actor}}(d_{\text{val}}; \phi_{old})\right) \tag{8}$$

この評価はのちのメタ最適化で Meta-critic の更新の損失関数として Meta-critic に反映する．

最後にメタ最適化である．ここでは Actor と Meta-critic のモデルパラメータを更新する．それぞれの更新は次の式で表される．

$$\phi \leftarrow \phi - \eta(\nabla_{\phi} L_{\text{actor}} + \nabla_{\phi} L_{\text{mcritic}}) \tag{9}$$

$$\omega \leftarrow \omega - \eta \nabla_{\omega} L_{\text{meta}}. \tag{10}$$

この更新によって，Actor は価値の高い行動の確率を高くすると同時に補助的な Meta-critic による損失関数により，その更新を学習データと異なるデータに対しても価値が高くなるような方向に学習を加速させる．つまり，Meta-critic は Actor の更新方向に対して，学習データと異なるデータに対しても価値が高くなるような更新になっているかを評価し，その結果に基づいてその学習を促進させるかどうかを判断する役割を持つ．

2.3 オンラインファインチューニング

本稿では，メタ強化学習によって学習したモデルを未学習環境に適用し，オンラインファインチューニングによって性能が向上するかどうかを検証する．以下 Algorithm 2 には，オンラインファインチューニング時の学習アルゴリズムを示す．オンラインファインチューニングでは，データ収集，モデル更新の 2 つのフェーズに分かれる．データ収集はロボットが歩行者混雑環境下で指定のスタート地点からゴール地点までナビゲーションする 1 エピソードずつで行う．またデータ収集時の方策は切り替え手法によるものとする．さらにこの方策で得られたデータのうちルールベースに切り替わったデータは論文内で定義される Switching Administrator が入力データが学習データに対して尤度が低いと判断したものであり，未学習のデータをこの手法によって効率的に収集することができる．この方法で収集されたデータをもとにオンラインで Actor と Critic のモデルの追加更新を行う．この際，学習済みの Meta-critic は更新を行わない．

Algorithm 2: Online fine-tuning algorithm using meta-critic

Input learned parameters ϕ, θ, ω
Switching model trained on a dataset $\mathcal{D}$ consisting of 200 episodes in the square-crossing scenario using ORCA
Explore new environment using switching method and store 1 episode of switched data as d^B
for *each iteration* **do**
  for *each step* **do**
    $y = r(s,a) + \gamma \mathbb{E}_{d^B}\left[\min_{i=1,2} Q_{\theta^i}(s', a')\right]$
    $\theta \leftarrow \arg\min_{\theta} \ \mathbb{E}_{d^B}\left[(Q_{\theta}(s,a) - y)^2\right]$
    $L_{\text{actor}} = \log \pi_{\phi}(a|s) \exp\left(\frac{1}{\lambda} W\right)$
    $L_{\text{mcritic}}(d^B; \phi) \leftarrow \frac{1}{N} \sum_{i=1}^{N} f_{\omega}(\pi_{\phi}(s_i), s_i, a_i)$
    $\phi \leftarrow \phi - \eta(\nabla_{\phi} L_{\text{actor}} + \nabla_{\phi} L_{\text{mcritic}})$
  end
end

3. シミュレーション実験

本稿の提案手法により，学習環境と異なる未学習の歩行者混雑環境下で性能を向上させることができるのか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．メタ強化学習モデルが学習するデータセットとして，square-crossing シナリオと circle-crossing シナリオで歩行者 5 人とロボットが ORCA[8)] で動いたデータをそれぞれ 200 エピソード分用意する．前者は d_{trn} として，後者は d_{val} として扱う．またデータ収集時に用いる切り替え手法は，square-crossing シナリオで歩行者 5 人の環境でロボットと歩行者 5 人が ORCA で行動したデータ 200 エピソード分のデータセット $\mathcal{D}$ で学習している．

3.1 オンラインファインチューニングの数値評価

本稿で提案したメタ強化学習手法のオンラインファインチューニングの数値評価を行う．テスト環境は circle-crossing シナリオ，Social Force Model[9)] に基づいて行動する歩行者 5 人の環境である．オンラインファインチューニングは Algorithm2 に基づいて行う．比較手法は以下の 6 つである．

AWAC-LB: 切り替えなしで学習済み方策のデータに基づいて，強化学習モデル (AWAC) をオンラインファインチューニング．

AWAC-SW: 切り替え発生データのみを用いて，強化学習モデル (AWAC) をオンラインファインチューニング．

Mcritic-LB: 提案手法の学習済みモデルを切り替えなしで用いてメタ学習モデル (Meta-critic) をオンラインファインチューニング．

Mcritic-RB: ルールベース手法のみで収集したデータに基づいてメタ学習モデル (Meta-critic) をオンラインファインチューニング．

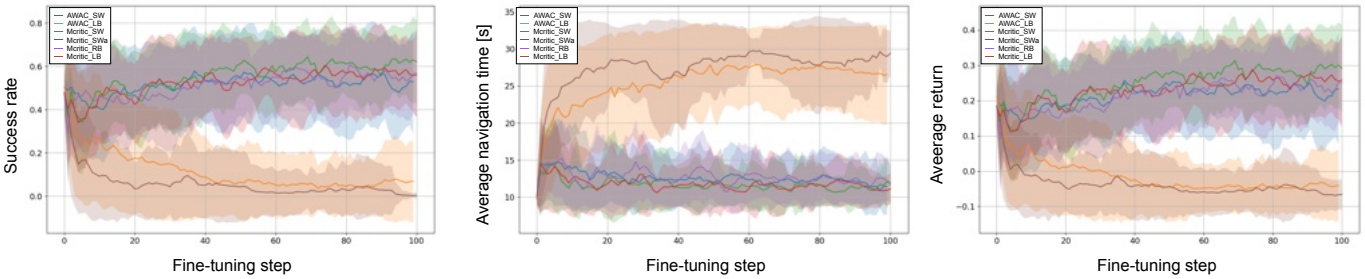
Mcritic-SWa: 切り替えで行動した際の 1 エピソード全体を用いてメタ学習モデル (Meta-critic) をオンラインファインチューニング．

Mcritic-SW: 切り替え発生データのみで収集したデータ全てに基づいてメタ学習モデル (Meta-critic) をオンラインファインチューニング．

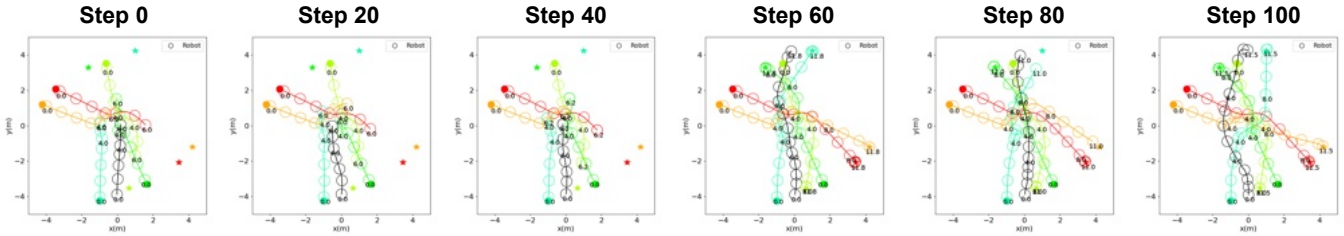
ここで，ルールベース手法とは，ORCA を指す．図 3(b) は上記 6 つの手法で 100 ステップのオンラインファインチューニングを行った際の成功率，平均到達時間，平均収益の推移である．1 ステップの更新ごとに 100 エピソードのテストを行っており数値は各手法に対して 5 つのモデル，10 個のシードでテストをした際の平均と標準

(a) Numerical comparison of performance improvement rates with the evolution of success rate，average arrival time, and average return during online fine-tuning

手法	成功率 [%]↑			平均到達時間 [s]↓			平均収益↑		
	0step	100step	変化率 [%]	0step	100step	変化率 [%]	0step	100step	変化率 [%]
AWAC-LB	50.4 ± 9.6	6.8 ± 18.9	−87.0	9.70 ± 0.07	26.52 ± 6.67	+173.4	0.211 ± 0.070	−0.041 ± 0.101	−119.4
AWAC-SW	50.4 ± 9.6	52.8 ± 23.9	+4.8	9.70 ± 0.07	11.40 ± 3.80	+17.5	0.211 ± 0.070	0.233 ± 0.157	+10.4
Mcritic-LB	47.9 ± 6.4	0.3 ± 1.7	−99.4	10.08 ± 1.26	29.43 ± 3.00	+192.0	0.184 ± 0.053	−0.066 ± 0.044	−135.9
Mcritic-RB	47.9 ± 6.4	56.9 ± 4.0	+18.9	10.08 ± 1.16	12.01 ± 2.65	+19.1	0.184 ± 0.053	0.251 ± 0.127	+36.4
Mcritic-SWa	47.9 ± 6.4	56.1 ± 20.0	+17.1	10.08 ± 1.16	11.09 ± 2.15	+10.0	0.184 ± 0.053	0.258 ± 0.127	+40.2
Mcritic-SW	47.9 ± 6.4	62.1 ± 19.4	+29.6	10.08 ± 1.16	11.68 ± 2.35	+15.9	0.184 ± 0.053	0.291 ± 0.123	+58.2



(b) Progression of success rate (left), average arrival time (center), and average return (right) during online fine-tuning



(c) Qualitative evaluation of robot and pedestrian trajectories at each update step

Fig. 3 Performance comparison table and results of online fine-tuning

偏差を表している．表 3(a) はオンラインファインチューニング時の各手法の 0 ステップ目と 100 ステップ目の成功率，平均到達時間，平均収益の数値結果とその変化率を表している．加えて，図中の網掛け部は，100 エピソード分の各テストにおける標準偏差を表す．この結果からまず，AWAC-LB，Mcritic-LB のように学習ベースによる方策のみでデータを収集した場合，いずれも性能が大幅に低下していることがわかる．このことから今回テストに用いた分布は学習データと分布が異なるデータであったことがわかる．そしてその他 4 つの手法で性能を比較すると，いずれも成功率と平均収益において，性能が向上した．提案したメタ強化学習手法はいずれも AWAC-SW と比較して成功率と平均収益で共に性能がより向上していた．特に Mcritic-SW は成功率で 29%，平均収益で 58%の性能向上となった．これは Meta-critic を利用した学習と切り替え手法でのデータ収集において，ルールベースに切り替わったデータのみを更新に用いることで効率良く未学習環境のデータを学習しやすくなった結果であると考える．しかし，平均到達時間に関して，いずれも 0 ステップに比べて遅い結果となった．これはオンラインファインチューニング初期で平均到達時間が長くなり，一度性能が下がってしまったことが原因であると考えられる．切り替え手法によるデータ収集により，従来の強化学習のファインチューニングにおける性能低下の問題を緩和し，さらにメタ強化学習として提案手法を用いることで，未学習環境でも性能を向上させることが可能であると考えられる．

3.2 オンラインファインチューニングの定性評価

本稿で提案したメタ強化学習手法のオンラインファインチューニングにおいて，一定更新ステップごとのロボットと歩行者の移動軌跡の定性評価を行う．テスト環境は circle-crossing シナリオ，Social Force Model に基づいて行動する歩行者 5 人が存在する環境とする．図 3(c) はオンラインファインチューニングによる更新 100 ステップにおける，20 ステップごとのロボットと歩行者の軌跡の結果である．この結果から 40 ステップまではロボットと歩行者の衝突が発生しているが，60 ステップ以降はロボットの軌跡は同じシナリオで変化しつつもいずれもロボットは歩行者を回避しながら目的地に到達できている．しかしながら 80 ステップにおける初期のロボットの減速，100 ステップにおけるゴール付近の迂回の現象が見られ，これらは数値評価における平均到達時間の性能低下の原因の一つであると考えられる．

4. 結言

本稿では，歩行者混雑環境下における移動ロボットナビゲーションに対して，メタ強化学習に基づいた手法を提案し，学習環境と異なる未学習環境での性能向上が見込めるかを検証した．メタ強化学習の学習方法として，深層強化学習における行動生成モデルの損失関数に補助的な損失関数である Meta-critic を追加し，その補助

損失そのものをニューラルネットワークで近似するものであった．Meta-critic を用いることで，学習データと異なるデータでも価値が高くなるような更新になるように学習を進めることが可能であり，未学習の環境でも性能向上を達成することができた．また過去に提案した学習ベースとルールベースの切り替え手法を用いた未学習環境を探索することで，未学習のデータを効率的に収集し，さらなる性能向上が見込めることを確認した．今後は人数の違いやシナリオの違いを含めた多様な環境でテストできるようにモデルを改良し，実環境でも性能向上が可能かの検証を行う．またシナリオが動的に変わる環境でも性能を維持または向上が可能な継続的な学習への拡張も検討する．

参考文献

[1] C. Finn, P. Abbeel, and S. Levine. Model-agnostic Meta-learning for Fast Adaptation of Deep Networks. *Proceedings of the International conference on machine learning*. (2017), pp. 1126–1135.

[2] W. Zhou, Y. Li, Y. Yang, H. Wang, and T. Hospedales. Online Meta-Critic Learning for Off-Policy Actor-Critic Methods. *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., (2020), pp. 17662–17673. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/cceff8faa855336ad53b3325914caea2-Paper.pdf.

[3] K. Matsumoto, Y. Hyodo, and R. Kurazume. Crowd-Aware Robot Navigation with Switching Between Learning-Based and Rule-Based Methods Using Normalizing Flows. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. (2024), pp. 4823–4830. DOI: 10.1109/IROS58592.2024.10802676.

[4] C. Chen, S. Hu, P. Nikdel, G. Mori, and M. Savva. Relational Graph Learning for Crowd Navigation. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. (2020), pp. 10007–10013. arXiv: 1909.13165.

[5] A. Nair, M. Dalal, A. Gupta, and S. Levine. Accelerating Online Reinforcement Learning with Offline Datasets. CoRR abs/2006.09359, (2020). arXiv: 2006.09359. URL: https://arxiv.org/abs/2006.09359.

[6] C. Chen, Y. Liu, S. Kreiss, and A. Alahi. 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, C. Liu, B. E. Shi, and M. Liu. 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] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha. Reciprocal n-Body Collision Avoidance. *Proceedings of the International Symposium of Robotic Research*. (2011), pp. 3–19. DOI: 10.1007/978-3-642-19457-3_1.

[9] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. Physical review E 51.5, p. 4282, (1995).