予測状態表現に基づく深層強化学習を用いた動的環境下における移動ロボットナビゲーション

○松本耕平　河村晃宏　安琪　倉爪亮（九州大学）

1. はじめに

近年，少子高齢化や地方の過疎化などの社会問題に対して，IoT，ロボット，人工知能，ビッグデータ等の新たな技術を導入して，解決を図る取り組みに注目が集まっている．特に，ロボティクス分野においては，自律ロボットによる運搬，警備，介護などの 3K タスクの代替が期待されている．これら日常生活環境での自律移動を伴うロボットの実現には，歩行者の行き交うような動的環境における安全で効率的なナビゲーション手法が重要となる

歩行者が多数存在する環境でロボットが適切に動作するためには，ロボットが歩行者の行動を理解して行動を選択する必要がある．歩行者の行動は，環境や個人の興味などによって様々で不確定であり，また，ロボット側も機構の制約のため，必ずしも理想的な行動ができるとは限らない．さらに，ロボットの行動によって歩行者の行動が変化する可能性もある．

このような歩行者の不確定な行動を考慮した手法として，部分観測マルコフ決定過程を用いて，歩行者の最終的な目的地を予測することで歩行者が行き交う環境で安全なナビゲーションをおこなう手法が提案されている [1, 2]．しかし，これらの手法は事前に作成された歩行者モデルを用いて人の行動を予測しているため，環境に依存するような歩行者の行動の変化に対応するのは困難である．また，ロボットの行動は離散的な加減速として制限されている．

本研究では，部分観測な問題を取り扱う手法の一つである，予測状態表現 (Predictive State Representation : PSR) に基づいた深層強化学習である Recurrent Predictive State Policy (RPSP) Network を移動ロボットのナビゲーションに適用する．PSR に基づく本手法を用い，連続的なロボットの行動と環境の時系列変化を学習することで，歩行者による動的環境において，安全な経路生成を行う．

2. 背景

2.1 Predictive State Representation

予測状態表現 (Predictive State Representation : PSR)[3] は，考え得る全てのパターンのテストを行った場合に予想される結果が全て分かっていれば，動的システムを完全に把握できているという考え方に基づいており，観測可能な情報を用いて状態を表現することで，事前の知識なしに部分的に観測可能な動的システムをモデル化することができる．有限な観測 $\mathcal{O} = \{o_1, o_2, \ldots, o_k\}$ と行動 $\mathcal{A} = \{a_1, a_2, \ldots, a_k\}$ のセットを持つ離散的なシステムの場合，時刻 t におけるシステムの状態表現は，現在までの履歴を条件としたテストの発生確率で構成されるベクトルである．ここで，テストは時刻 $t+1$ から始まる行動と観察のシーケンスであり，時刻 t における履歴は，時刻 t までの，行動と観察のシーケンスである．履歴 h に対する長さ m のテスト τ の成功確率，つまり，τ の一連の行動を取った際に，τ の一連の観測を得る確率は $p(\tau \mid h) = p(h, \tau)/p(h) = \prod_{i=1}^{m} \Pr(o_i \mid h, a_i)$ と定義される．

一部のテストの成功確率を把握することで，他のテストの成功確率を把握できる場合があり，テストセット $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_k\}$ が与えられた場合に，予測ベクトル $p(\mathcal{T} \mid h) = [p(\tau_1 \mid h)\, p(\tau_2 \mid h) \ldots p(\tau_k \mid h)]$ が，任意のテスト τ_l に対して，$p(\tau_l \mid h) = f_t(p(\mathcal{T} \mid h))$ となる関数 f_t が存在する場合，$\mathcal{T}$ はコアテストと呼ばれ，予測ベクトル $p(\mathcal{T} \mid h)$ は全てのテストを予測するにあたり十分な統計量であり，PSR の状態を表す．

2.2 Recurrent PSR

基本的な PSR モデルは離散的な観測・行動からなるシステムのみに適応できる．これまでに，PSR を連続的なシステムに対応できるようにした手法が提案されている．本稿では，これらの手法を総称して Recurrent PSR (RPSR) と呼ぶ．RPSR において状態 q_t は，将来の行動 $a_{t:t+k-1}$ によって条件付けられた将来の観測 $o_{t:t+k-1}$ の条件付き分布である．RPSR のアーキテクチャを図 1 に示す．

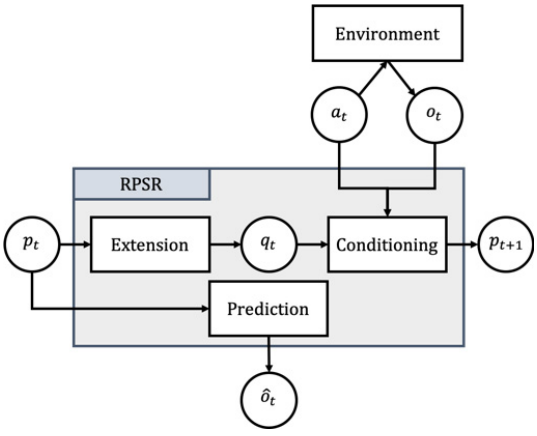

図 1　RPSR のアーキテクチャ

RPSR の状態の更新は 2 つの手順で行われる．

- Extension : 状態 q_t に線形写像 W_{ext} を適用し，拡張状態 p_t を得る．拡張状態 p_t は，拡張された行動 $a_{t:t+k}$ によって条件付けられた拡張された観測 $o_{t:t+k}$ の条件付き分布である．また，W_{ext} は学習によって最適化されるパラメータである．

$$p_t = W_{\text{ext}} q_t \tag{1}$$

- Conditioning： 時刻 t における行動 a_t と観測 o_t に．既知の条件付け関数 f_{cond} により，以下のように状態が更新される．

$$q_{t+1} = f_{\mathrm{cond}}\left(p_t, a_t, o_t\right) \tag{2}$$

離散的なシステムにおいて，q_t と p_t は条件付き確立テーブルで表され，f_{cond} はベイズ則を適用する．これらを，連続的なシステムに応用するために，分布のヒルベルト空間埋め込み [4] とカーネルベイズ則 [5] を用いる．

本研究では，RPSR のモデルとして RFF-PSR [6] を用いる．RFF-PSR では，観測と行動のデータに RBF カーネルによる写像を適用後，ランダムフーリエ特徴 [7] を抽出し，ランダム主成分分析 [8] を用いて次元削減したものを，それぞれ観測と行動データの特徴量として用いる．この特徴量を抽出する関数を ϕ で表す．これを用いて，観測予測関数 f_{pred} によって以下のように時刻 t における観測が予測される．

$$\begin{aligned}\hat{o} &= f_{\mathrm{pred}}\left(q_t, \phi\left(a_t\right)\right) \\ &= W_{\mathrm{pred}}\left(q_t \otimes \phi\left(a_t\right)\right)\end{aligned} \tag{3}$$

ここで，W_{pred} は学習で最適化される線形写像であり，$\otimes$ はクロネッカー積を表す．

3. 提案手法

3.1 PSR の移動ロボットナビゲーションへの適用

本研究では PSR の構造を応用した，深層強化学習を複数の歩行者が行き交う動的環境下での移動ロボットのナビゲーションに適用する．以下のように移動ロボットナビゲーションと PSR の要素を対応づける．

- 観測 : ロボットと n 人の歩行者に関して，それぞれの位置情報を観測とする．具体的には 2 次元空間におけるロボットの位置 (x_r, y_r) と歩行者の位置 (x_p, y_p) からなるベクトル $(x_r, y_r, x_p^1, y_p^1 \cdots x_p^n, y_p^n)$ を観測とする．
- 行動 : 本研究ではホロノミックな全方向移動ロボットを想定し，2 次元空間におけるロボットの x 軸方向の入力速度 v_x と y 軸方向の入力速度 v_y からなる 2 次元ベクトル (v_x, v_y) を行動とする．

また，環境から得られる報酬を以下のように設定する．

$$R_t = \begin{cases} -0.25 & \text{if } d_t < 0 \\ -0.1 + d_t/2 & \text{else if } d_t < 0.2 \\ 1 & \text{else if } p_t = p_g \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

d_t はロボットと周囲の歩行者の最小距離であり，p_t はロボットの位置，p_g はナビゲーションの目標位置を表す．

3.2 ネットワークアーキテクチャ

本研究では，深層強化学習として RPSR に応用した Recurrent Predictive State Policy (RPSP) Network [9] を拡張し，Actor-Critic [10] の構造を取り入れた手法を用いる．提案するアーキテクチャを図 2 に示す．Actor-Critic には．決定的方策を Actor として持つ Deterministic Actor-Critic [11] を用い，Actor と Critic は RPSR の状態 q_t を入力とし，それぞれ行動と価値を出力する．

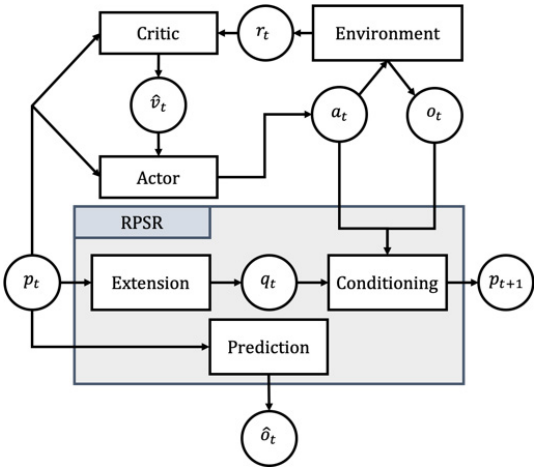

図 2　RPSR Actor-Critic のアーキテクチャ

4. 学習手法

学習は PSR の初期化と方策の学習の 2 段階で行われる．

4.1 PSR の初期化

提案するネットワークを学習するために，PSR の初期化を行い PSR のパラメータ $\theta_{\mathrm{PSR}} = \{q_0, W_{\mathrm{ext}}, W_{\mathrm{pred}}\}$ の初期値を定める．この初期化は，ランダムな行動を取る方策を用いてデータを収集したのちに，Two-stage Regression [12] を用いて行う．

4.2 方策の最適化

方策の最適化には Twin Delayed DDPG (TD3) [13] を基に，PSR パラメータの学習を組み込んだ手法を用いる．アルゴリズム 1 に最適化のアルゴリズムを示す．

5. シミュレーション実験

シミュレーション実験を行い，提案した手法が歩行者による動的環境下での移動ロボットナビゲーションタスクにおいて，安全な経路を生成できることを確認する．

5.1 シミュレーション環境

シミュレーション環境を図 3 に示す．本環境は CrowdNav [14] 環境を基にしており，図中の黄色の円がロボットを表し，その他の円は歩行者を表す．ロボットはゴール地点 $(x, y) = (0, 4)$ の位置を目指して進み，歩行者は群衆シミュレーション手法の ORCA [15] に従い行動し，環境の中心を経由して向かい側を目指す．歩行者の初期位置はエピソードごとにノイズを与えて初期化される．また，本環境での歩行者の人数は 5 人である．

5.2 学習の設定

RFF-PSR に関して，ランダムフーリエ特徴の数は 1000 に設定し，ランダム主成分分析による削減後の次元数は 40 に設定した．

Algorithm 1: 提案ネットワークの最適化

PSR パラメータ θ_{PSR} を Two-stage Regression により初期化し，Critic $Q_{\theta_1}, Q_{\theta_2}$ と Actor π_ϕ をランダムパラメータ θ_1, θ_2, ϕ で初期化
ターゲットをネットワークを $\theta_1' \leftarrow \theta_1, \theta_2' \leftarrow \theta_2 \phi' \leftarrow \phi$ により初期化
リプレイバッファ $\mathcal{B}$ を初期化
for $i = 1$ **to** T **do**
  探索方策に従い行動 a を選択し，報酬 r と観測 o を取得
  状態 q にを更新し，次状態 q' を取得
  (q, o, a, r, q', o') をリプレイバッファ $\mathcal{B}$ に格納

  リプレイバッファ $\mathcal{B}$ からミニバッチをサンプルし，N 組みのデータ (q, o, a, r, q', o') を取得
  $\theta_{\mathrm{PSR}} \leftarrow \mathrm{argmin}_{\theta_{\mathrm{PSR}}} N^{-1} \sum \left(o - W_{\mathrm{pred}}\left(q_t \otimes a_t\right)\right)^2$
  $\tilde{a} \leftarrow \pi_{\phi'}\left(q'\right) + \epsilon, \quad \epsilon \sim \mathrm{clip}(\mathcal{N}(0, \tilde{\sigma}), -c, c)$
  $y \leftarrow r + \gamma \min_{i=1,2} Q_{\theta_i'}\left(q', \tilde{a}\right)$
  $\theta_i \leftarrow \mathrm{argmin}_{\theta_i} N^{-1} \sum \left(y - Q_{\theta_i}(q, a)\right)^2$ により Critic を更新
  if t *mod* d **then**
    決定的方策勾配法により，ϕ を更新
    $\nabla_\phi J(\phi) = N^{-1} \sum \nabla_a Q_{\theta_1}(q, a)\big|_{a=\pi_\phi(q)} \nabla_\phi \pi_\phi(q)$
    ターゲットネットワークを更新
    $\theta_i' \leftarrow \tau\theta_i + (1-\tau)\theta_i'$
    $\phi' \leftarrow \tau\phi + (1-\tau)\phi'$
  end
end

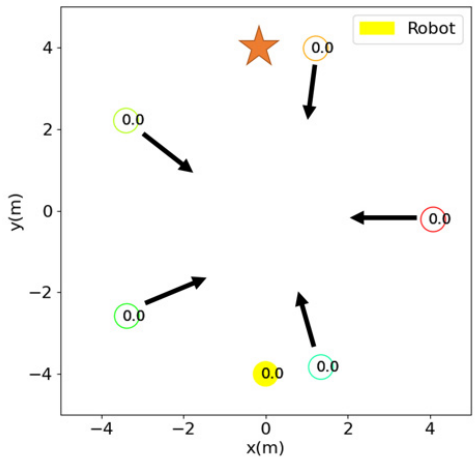

図 3　シミュレーション環境

また，学習に関して，TD3 における探索ノイズは $\mathcal{N}(0, 0.1)$ とし，ターゲットネットワークから生成される行動には，$\mathcal{N}(0, 0.1)$ に従う $(-0.5, 0.5)$ の範囲でクリッピングしたノイズを与える．全体の繰り返し回数 T は 1000000 とし，遅延方策更新パラメータ d を 2 に設定する．

探索は，ゴールに到達する有効なサンプルを得るために，繰り返し回数 i が $i < T/4$ の場合は 50%，$T/4 \leq i < T/3$ では 30%，$T/3 \leq i < T/2$ では 10%の確率で ORCA に従う方策で探索を行う．

5.3　実験結果

提案手法によるシミュレーション実験の結果を図 4 に示す．また，500 エピソード中の，提案手法と群衆

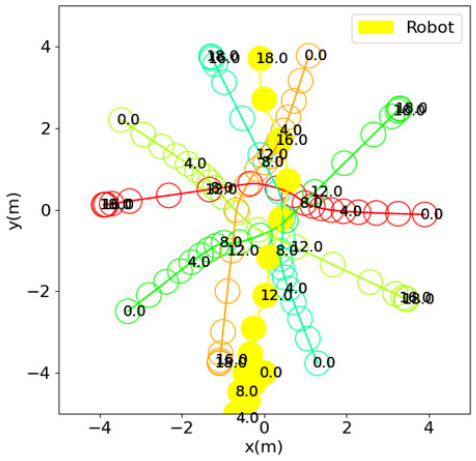

図 4　実験結果の軌跡

シミュレーション手法 ORCA に従う方策を用いた場合のナビゲーションタスクの成功率，衝突率，未達成率，実行時間を表 1 に示す．この比較より，提案手法の方

手法	成功率 [%]	衝突率 [%]	未達成率 [%]	実行時間 [s]
ORCA	94.0	2.8	3.2	14.7
提案手法	96.0	1.0	3.0	14.9

表 1　提案手法と ORCA に従う方策の結果の比較

がより成功率が高く，衝突率が低いことが確認できた．

6.　まとめと今後の予定

本研究では，予測状態表現に基づく深層強化学習手法を，歩行者による動的環境下での移動ロボットナビゲーションタスクに適用した．

また，シミュレーション環境で実験を行い，提案手法が周囲の歩行者を回避して目的地まで到達する安全な経路を生成可能であることを確認した．

今後は，より複雑なシミュレーション環境での実験や，学習時のパラメータの変化による結果への影響を考慮し，ハイパーパラメータの最適化を行う．

また，本手法は，歩行者による動的環境に特化した構造を持っているわけではない．そこで，これまでに提案されている歩行者の行動予測などに用いられている手法等を参考にし，歩行者による動的環境に特化した構造を模索する予定である．

参　考　文　献

[1] H. Bai, S. Cai, N. Ye, D. Hsu, and W. S. Lee, “Intention-aware online POMDP planning for autonomous driving in a crowd,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 454–460, IEEE, may 2015.

[2] Y. Luo, P. Cai, A. Bera, D. Hsu, W. S. Lee, and D. Manocha, “PORCA: Modeling and Planning for Autonomous Driving Among Many Pedestri-

ans," *IEEE Robotics and Automation Letters*, vol. 3, pp. 3418–3425, oct 2018.

[3] S. Singh, M. James, and M. Rudary, "Predictive State Representations: A New Theory for Modeling Dynamical Systems," in *20th The Conference on Uncertainty in Artificial Intelligence*, 2004.

[4] B. Boots, A. Gretton, and G. J. Gordon, "Hilbert space embeddings of predictive state representations," in *Uncertainty in Artificial Intelligence - Proceedings of the 29th Conference*, pp. 92–101, 2013.

[5] K. Fukumizu, L. Song, and A. Gretton, "Kernel Bayes' rule: Bayesian inference with positive definite kernels," *Journal of Machine Learning Research*, vol. 14, pp. 3753–3783, 2013.

[6] A. Hefny, C. Downey, and G. Gordon, "An efficient, expressive and local minima-free method for learning controlled dynamical systems," in *32nd AAAI Conference on Artificial Intelligence*, pp. 3191–3198, feb 2018.

[7] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Advances in Neural Information Processing Systems 20*, 2008.

[8] N. Halko, P. G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Review*, vol. 53, no. 2, pp. 217–288, 2011.

[9] A. Hefny, Z. Marinho, W. Sun, S. S. Srinivasa, and G. Gordon, "Recurrent predictive state policy networks," in *35th International Conference on Machine Learning*, vol. 5, pp. 3104–3119, 2018.

[10] V. R. Konda and J. N. Tsitsiklis, "Actor-Critic Algorithms," in *Advances in Neural Information Processing Systems 12*, pp. 1008–1014, 2000.

[11] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *31st International Conference on Machine Learning, ICML 2014*, vol. 1, pp. 605–619, 2014.

[12] A. Hefny, C. Downey, and G. J. Gordon, "Supervised learning for dynamical system learning," in *Advances in Neural Information Processing Systems*, vol. 2015-Janua, pp. 1963–1971, 2015.

[13] S. Fujimoto, H. Van Hoof, and D. Meger, "Addressing Function Approximation Error in Actor-Critic Methods," in *35th International Conference on Machine Learning, ICML 2018*, vol. 4, pp. 2587–2601, 2018.

[14] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, "Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2019-May, pp. 6015–6022, 2019.

[15] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Springer Tracts in Advanced Robotics*, vol. 70, pp. 3–19, 2011.