予測状態表現に基づく歩行者行動予測を用いた深層強化学習によ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

歩行者が多い環境での円滑な自律移動は，生活環境で動作するサービスロボットにとって不可欠である．これには，ロボットが歩行者の行動を把握し予測することが重要であるが，人間の行動は，意図や環境への影響など，事前に直接観察またはモデル化できない要因の影響を受ける可能性があり，さらにロボット自身の行動の影響を受けて変化する可能性もある．しかし，これまでに提案されている手法は，予め用意された歩行者の行動のモデルを用いており，特にロボットの行動による歩行者の行動の変化を考慮していない．

本研究では，行動による影響を考慮して環境の変化を予測する予測状態表現に基づく深層強化学習法を，動的環境における移動ロボットナビゲーションに適用することで，ロボットの行動による周囲の歩行者の行動の変化を考慮できるナビゲーション手法を提案する．さらに，占有地図に基づいて歩行者の情報を統合することで，歩行者数の変化に対応することを検討する．

2. 背景

2.1 Predictive State Representation

予測状態表現 (Predictive State Representation : PSR)[1] 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 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．

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

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 [2] とカーネルベイズ則 [3] を用いる．

本研究では，RPSR のモデルとして RFF-PSR [4] 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 [5] を抽出し，ランダム主成分分析 [6] 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．

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

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．

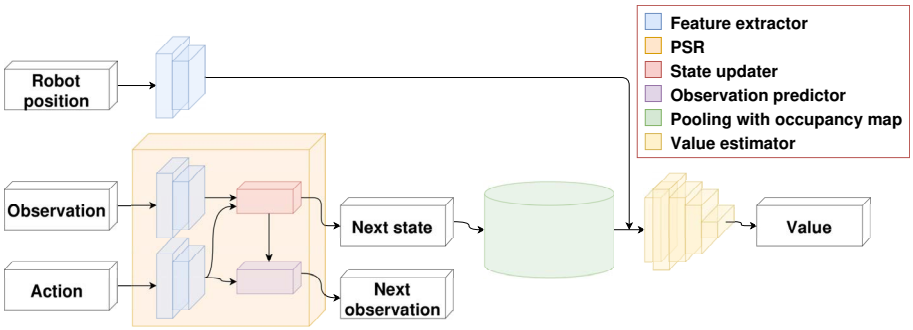

図 1　提案手法のアーキテクチャ

- 観測：ロボットの位置を中心とした 2 次元座標系における，n 人の歩行者に関して，それぞれの位置情報 (x^p, y^p) と速度情報 (v_x^p, v_y^p) 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．

これによって，本研究で用いる PSR はロボットの位置を中心とした座標系における歩行者の位置と，ロボットの入力速度を入力として次のステップの歩行者の位置を予測することができるモデルになる．この予測の流れを図 2 に示す．

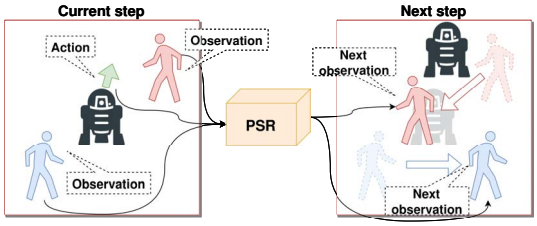

図 2 移動ロボットナビゲーションにおける PSR の予測の流れ

3.2　提案手法

提案するアーキテクチャを図 1 に示す．提案手法は，Feature extractor，State updater と Observation predictor から構成される PSR，Pooling with occupancy map，Value estimator によって構成される．それぞれの説明を以下に示す．

- Feature extractor：それぞれの入力から特徴抽出を行う．
- PSR ：観測と行動から抽出された特徴を用いて State updater により状態を更新し，Observation predictor で観測の予測を行う．
- Pooling with occupancy map：占有地図に基づいてそれぞれの歩行者に対応する PSR 内の状態を統合する．
- Value estimator：占有地図の情報とロボットの位置情報から価値を推定する．

4.　占有地図を用いた状態の統合

本研究では，環境内の歩行者の人数が変化する場合に対応するために，ロボットを中心とした円形の占有地図を用いた状態の統合を用いる．これによって，環境内の人数に関係なく Value estimator が受け取る情報の量は一定になるため，歩行者数が変化した場合でも価値を推定することが可能になる．この状態の統合の流れを図 3 に示す．PSR から取得される各歩行者の状態は，それぞれの歩行者の位置に応じて，占有する占有地図のセルに格納される．この際に，複数の歩行者が同一のセルを占有する場合はそれらの状態の平均値をセルに格納する．

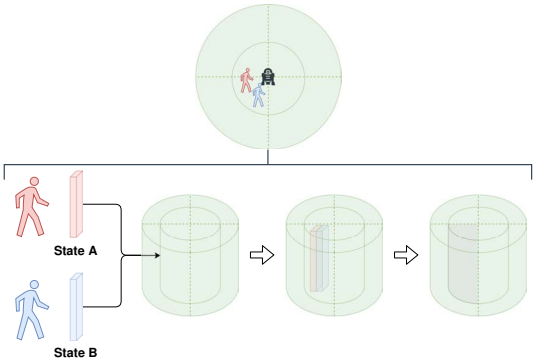

図 3　占有地図を用いた状態の統合の流れ

5.　行動の生成

行動は学習された Value estimator f_v と PSR モデル f_p の State updater f_p^q と Observation predictor f_p^o を用いて，行動空間 A のうちから最大の価値を得られる行動を式 (4) に従って選択することで生成される．ここで，γ は割引率を表し，m_t は時刻 t における占有地図の情報を表し，p_t は時刻 t におけるロボットの位置を表す．また，$R(o_t)$ は時刻 t に取得される報酬であり，式 (5) に従って報酬を取得する．

$$\begin{aligned} a_t \leftarrow \mathrm{argmax}_{a_t \in A} R(\hat{o}_{t+1}) + \gamma^{\Delta t} f_v(m_t, p_t) \\ \text{where } \hat{q}_{t+1} = f_p^q(q_t, a_t, o_t), \\ \hat{o}_{t+1} = f_p^o(q_t, a_t) \end{aligned} \tag{4}$$

$$R(o_t) = \begin{cases} -0.25 & \text{if } d_t < 0 \\ -0.1 + d_t/2 & \text{else if } d_t < 0.2 \\ 1 & \text{else if } p_t = p_g \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

式 (5) における d_t はロボットと周囲の歩行者の最小距離であり，p_g はナビゲーションの目標位置を表す．

6. 学習手法

6.1 事前学習

提案手法全体を学習するために，PSR の事前学習を行い PSR のパラメータを初期化する．またこの際に，収集したデータから割引報酬を計算し，占有地図からこの割引報酬を推定できるように Value estimator を学習する．この事前学習には，ORCA[7] に基づいて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 [8] を用いて行う．

6.2 提案手法の学習

Algorithm 1 に提案手法の学習アルゴリズムを示す．

Algorithm 1: 提案手法の学習

Two-stage Regression により PSR f_p，Value estimator f_v を初期化
ターゲット Value estimator $\hat{f}_v$ を初期化
for $i = 1$ **to** E **do**
  探索方策に従い行動 a_t を選択し，報酬 r_t，観測 o_t，ロボットの位置 p_t 取得
  エピソードが終了した場合 (o_t, a_t, r_t, p_t) の軌跡をバッファ $\mathcal{B}$ に格納
  バッファ $\mathcal{B}$ からミニバッチをサンプルし，N 組みの軌跡データを取得
  軌跡データから状態の軌跡 $\mathbf{q} = \{q_1, q_2, \ldots, q_T\}$，$\hat{f}_v$ による価値のターゲット $\mathbf{y} = \{y_1, y_2, \ldots, y_T\}$，歩行者の状態が格納された占有地図 $\mathbf{m} = \{m_1, m_2, \ldots, m_T\}$ を取得
  $L_{\text{pred}} = \text{MSE}(f_p^o(q_t, a_t), o_{t+1})$ を最小化するように f_p を更新
  $L_{\text{value}} = \text{MSE}(f_v(m_t, p_t), y_t)$ を最小化するように f_v を更新
  $\hat{f}_v \leftarrow \mu f_v + (1-\mu)\hat{f}_v$ により $\hat{f}_v$ を更新
end

7. シミュレーション実験

7.1 シミュレーション環境

シミュレーション環境を図 4 に示す．本環境は CrowdNav [9] 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 [7] 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中心 $(x, y) = (0, 0)$，半径 4m の円上にランダムに配置され，エピソードごとにノイズを与えて初期化される．

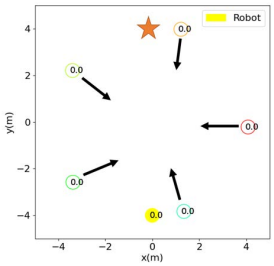

図 4　シミュレーション環境

7.2 実験設定

提案手法を用いてロボットが歩行者に衝突せずにゴールに到達することが可能であるか，環境内の歩行者数が変化しても対応できるかどうかを確認する．

実験 1 では歩行者数 5 人の環境で学習した提案手法を用いて，歩行者が 5 人のテスト環境での性能評価を行う．実験 2 では歩行者数 5 人の環境で学習した提案手法を用いて，歩行者が 1~10 人で変化数テスト環境での性能評価を行う．各実験では歩行者の初期位置を変えながら 500 エピソード分評価を行う．

7.3 実験 1

提案手法によるシミュレーション実験の結果の軌跡のうちの 4 つのサンプルの結果を図 5 に示す．

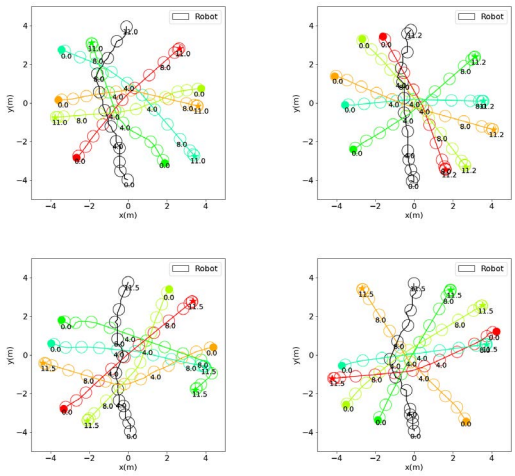

図 5　実験結果の軌跡のサンプル

また，歩行者の初期位置を変えながら 500 エピソード実行した場合の成功率，衝突率，未達成率，実行時間を表 1 に示す．

成功率 [%]	衝突率 [%]	未達成率 [%]	実行時間 [s]
85.4	14.2	0.4	12.6

表 1　提案手法の数値評価

この実験により，提案手法を用いて学習時とテスト時の歩行者数が同じ場合に約 85%の割合で目的を達成できることを確認した．

7.4 実験 2

提案手法によるシミュレーション実験の結果の軌跡のうち，歩行者が 1~10 人の場合のサンプルを 1 つずつ図 6 に示す．

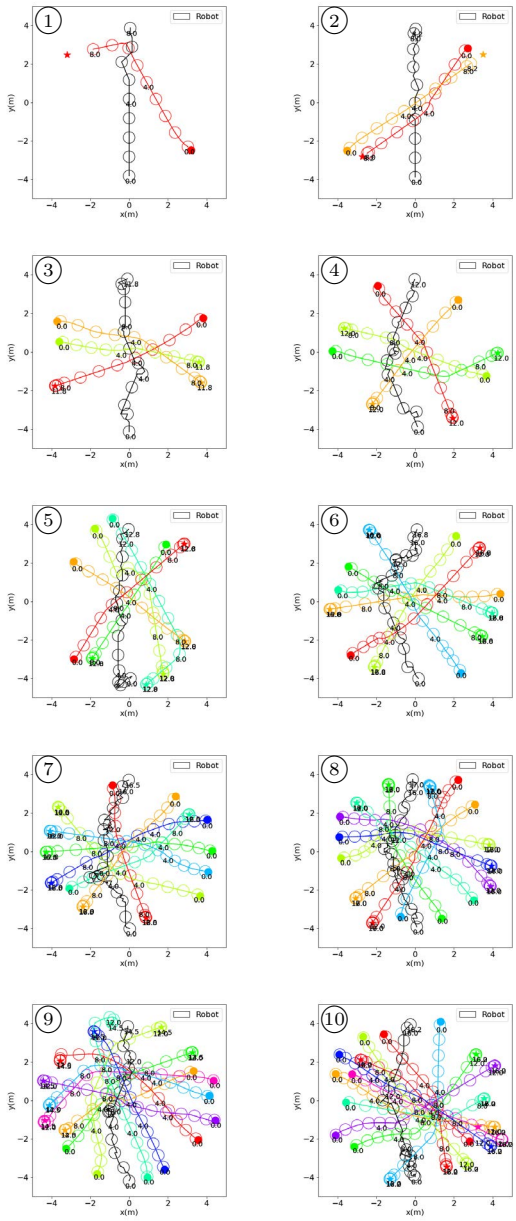

図 6　実験結果の軌跡 (各図左上の数字は歩行者数)

さらに，歩行者の初期位置を変えながら 500 エピソード実行した場合の平均の成功率，衝突率，未達成率，実行時間を表 2 に示す．

成功率 [%]	衝突率 [%]	未達成率 [%]	実行時間 [s]
56.2	42.4	1.4	12.5

表 2　歩行者数が変化する環境での提案手法の数値評価

この実験により，テスト時に学習時と歩行者数が異なる状況を含む場合に，学習時とテスト時の歩行者数が同じ場合に比べて性能が下がるものの，約 56%の割合で目的を達成できることを確認した．

8.　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，学習時とテスト時の歩行者数が同じ場合に提案手法が約 85%の割合で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．加えて，学習時とテスト時の歩行者数が異なるような場合にも，性能は低下するものの，目的を達成可能であることを確認した．

今後はさらなる性能向上のためのパラメータや学習手法の調整，歩行者同士の関係性を表現できる構造を取り入れることなどを検討していく．また，学習時とテスト時の歩行者数が異なる場合に，学習時の人数とテスト時の人数の関係が性能に与える影響や，学習時に人数を変化させることで性能に影響があるかなどの検討も行い，学習時とテスト時の歩行者数が異なる場合の性能の改善を試みる．さらに，現在はシミュレーション内での実験に留まっているが，実世界での実験も取り組む予定である．

謝　辞　本研究の一部は JSPS 科研費 JP20H00230 の助成を受けたものです．

参　考　文　献

[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

[6] 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.

[7] 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.

[8] 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.

[9] 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.