直交ランダム特徴量を用いた予測状態表現に基づく移動ロボットナビゲーション

○松本 耕平（九州大学）　倉爪 亮（九州大学）

歩行者が行き交う動的な環境での自律移動は，人間の周囲でサービスを行うロボットシステムの実現に向けて重要な課題である．我々はこれまでに，行動による環境の変化を学習できる予測状態表現に基づいた深層強化学習ベースの移動ロボットナビゲーション手法に取り組んできた．本研究では，予測状態表現モデルの性能を向上することが報告されている直交ランダム特徴量を導入し，ベンチマークデータセットとシミュレーション環境での予測性能及び，タスクの成功率を比較することでその有効性を検証する．

1. はじめに

生活環境で動作するサービスロボットにとって，歩行者が行き交う動的な環境での自律移動は重要な課題である. このためには，ロボットは歩行者の行動を予測し，歩行者の行動に合わせて自身の行動を決定する必要がある. しかし, 人間の行動は意図や環境の影響など, 事前に直接観測したりモデル化したりすることが困難な要因によって影響を受ける可能性があるため，歩行者の行動を事前に予測することは容易ではない. また, 歩行者の行動はロボットの行動に影響され変化する可能性もある.

我々はこれまでに, 行動による環境の変化を学習する, 予測状態表現に基づいた深層強化学習ベースの移動ロボットナビゲーション手法を提案してきた [1, 2, 3, 4]. この予測状態表現にはランダムフーリエ特徴量が利用されているが，このランダムフーリエ特徴量の代わりに直交ランダム特徴量を利用することで予測状態表現に基づく時系列モデリング手法の性能が改善することが報告されている [5]．そこで，本研究では，直交ランダム特徴量を予測状態表現を用いた深層強化学習ベースの移動ロボットナビゲーション手法に適用する．

2. 予測状態表現

予測状態表現 (Predictive State Representation : PSR)[6] は，観測・行動から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，将来の観測と行動の予測によって状態を表現する．

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用できる手法であるが，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 [7, 8]．本稿では，これらの手法を総称して Recurrent PSR (RPSR) と呼称する．

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 e_t を得る．拡張状態 e_t は，1 ステップ先までの行動列 $a_{t:t+k}$ によって条件付けられた 1 ステップ先までの観測列 $o_{t:t+k}$ 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．

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

- 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．

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

q_t と e_t は条件付き分布であり，f_{cond} はベイズ則を適用する．連続的なシステムに応用するために，これらに対して，ヒルベルト空間埋め込み [7] とカーネルベイズ則 [9] を用いる．

観測及び，行動から特徴量を抽出する関数を ϕ で表すと，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．

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

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. 直交ランダム特徴量

観測及び行動の特徴量として，RPSR のベースとして利用する RFF-PSR[8] では ϕ にはランダムフーリエ特徴量 (RFF) が利用される．RFF の特徴抽出関数は以下の式で表される．

$$\phi(\boldsymbol{x}) = \sqrt{\frac{2}{D}} \cos(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}) \tag{4}$$

ここで，$\boldsymbol{W}$ は正規分布からサンプルされ，$\boldsymbol{b}$ は一様分布からサンプルされる．また，D はサンプル数を表す．

一方で，直交ランダム特徴量 (ORF) では，$\boldsymbol{W}$ は以下のようにサンプルされる．

$$\boldsymbol{W}_{\text{ORF}} = \boldsymbol{S}\boldsymbol{Q} \tag{5}$$

ここで，$\boldsymbol{Q}$ は正規分布からサンプルされた値を要素に持つ行列 $\boldsymbol{G}$ に対して QR 分解を行うことで得られる．$\boldsymbol{S}$ は対角行列であり，自由度 d の χ 分布からサンプルされる．

4. RPSR を用いた移動ロボットナビゲーションのための深層強化学習モデル

RPSR を用いた移動ロボットナビゲーションのための深層強化学習モデルのアーキテクチャを図 1 に示す. このモデルは Feature extractor，State updater，Observation predictor, State integrator, Value estimator により構成される．それぞれの説明を以下に示す.

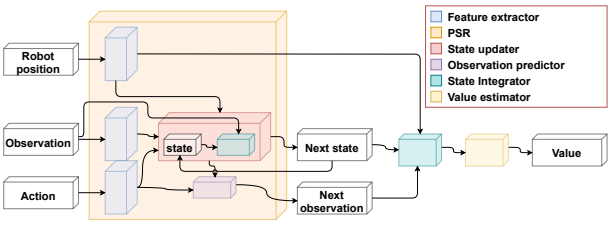

図 1 RPSR を用いた移動ロボットナビゲーションのための深層強化学習モデルのアーキテクチャ.

- Feature extractor：入力情報から特徴を抽出する．本研究では RFF と ORF を利用して比較を行う．
- State updater：観測と行動から抽出された特徴量を用いて，状態を更新する．
- Observation predictor：行動から抽出された特徴量と状態から，行動を行った後の観測を予測する．
- State integrator：各歩行者に対応する状態情報を統合する．本研究ではグラフ畳み込みによる状態統合 [4] を用いる．
- Value estimator：統合された状態とロボットの情報から，行動の価値を推定する．

4.1 学習手法

学習は 2 つのステップで行われる．最初のステップでは，RPSR パラメータを学習する．この学習は，ORCA [10] に従った行動方針による探索でデータを収集した後，two-stage regression [11] によって予測の学習を行い，Value estimator を模倣学習することで行われる．その後，アルゴリズム 1 に従い，Value estimator を強化学習し，教師あり学習で予測モデルを学習させる．

Algorithm 1: RPSR を用いた移動ロボットナビゲーションのための深層強化学習モデルの学習

Value estimator のターゲットネットワーク $\hat{f}_v$ を初期化
for $i = 1$ **to** E **do**
  探索方策に従い a_t を選択し，報酬 r_t，観測 o_t，ロボットの位置 p_t を取得
  エピソード終了後，軌跡 (o_t, a_t, r_t, p_t) をバッファ $\mathcal{B}$ に保存
  $\mathcal{B}$ から M 個の軌跡をサンプルする
  for $j = 1$ **to** M **do**
    以下の軌跡を取得する
      観測 $\boldsymbol{o}^j = \left\{o_1^j, o_2^j, \ldots, o_T^j\right\}$,
      ロボットの位置 $\boldsymbol{p}^j = \left\{p_1^j, p_2^j, \ldots, p_T^j\right\}$,
      行動 $\boldsymbol{a}^j = \left\{a_1^j, a_2^j, \ldots, a_T^j\right\}$
    以下の軌跡を計算する
      状態 $\boldsymbol{q}^j = \left\{q_1^j, q_2^j, \ldots, q_T^j\right\}$,
      価値の目標値 $\boldsymbol{y}^j = \left\{y_1^j, y_2^j, \ldots, y_T^j\right\}$,
      統合された状態 $\boldsymbol{s}^j = \left\{s_1^j, s_2^j, \ldots, s_T^j\right\}$
  end
  $L_{\text{pred}} = \text{MSE}(f_p^o(\boldsymbol{q}, \boldsymbol{a}), \boldsymbol{o})$ を最小化するように f_p を更新
  $L_{\text{value}} = \text{MSE}(f_v(\boldsymbol{s}, \boldsymbol{p}), \boldsymbol{y})$ を最小化するように f_v を更新
  if i mod d **then**
    $\hat{f}_v \leftarrow f_v$ によって $\hat{f}_v$ を更新
  end
end

5. 比較実験

5.1 ベンチマークデータでの比較

先行研究で用いられているベンチマークデータセット [7, 8] を利用して，two-stage regression 後の 1 ステップオンライン予測における性能を評価した．このデータは式 (6) で表される．入力 $a(t)$ は -0.5 から 0.5 の間に一様に分布する白色雑音として生成され，予測する出力は $o(t)$ である．100 個の出力と入力からなる 10 通りの軌跡で学習し，別の 10 通りの軌跡でテストした，学習データとテストデータの軌跡のサンプルを，それぞれ図 2 と図 3 に示す．比較は RFF と ORF の特徴量のサンプル数を 10~50 まで 10 ずつ変化させながら行った．

$$\begin{aligned}\dot{x}_1(t) =& x_2(t) - 0.1\cos\left(x_1(t)\right)\left(5x_1(t) - 4x_1^3(t) + x_1^5(t)\right) \\ & - 0.5\cos\left(x_1(t)\right)a(t) \\ \dot{x}_2(t) =& -65x_1(t) + 50x_1^3(t) - 15x_1^5(t) - x_2(t) - 100a(t) \\ o(t) =& x_1(t)\end{aligned} \tag{6}$$

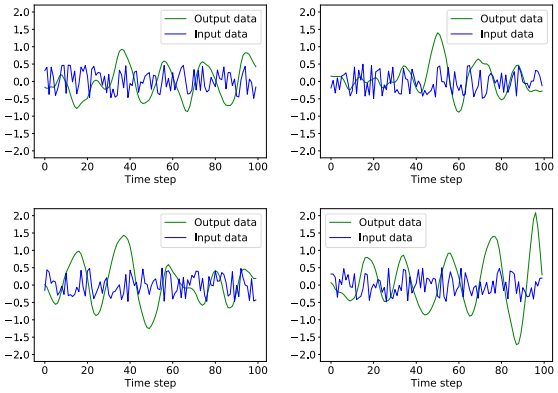

図 2 学習データの軌跡のサンプル.

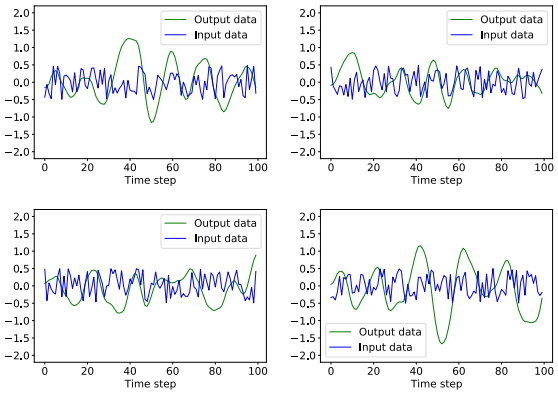

図 3 テストデータの軌跡のサンプル.

結果を図 4 に示す．この図より，一貫して ORF の方が誤差が小さくなっていることが分かる．また，RFF，ORF ともにサンプル数が増加するほど誤差が小さくなっていることが分かる．

5.2 シミュレーション環境での比較

シミュレーション環境には，CrowdNav 環境の circle crossing シナリオを利用する [12, 13]．このシナリオで

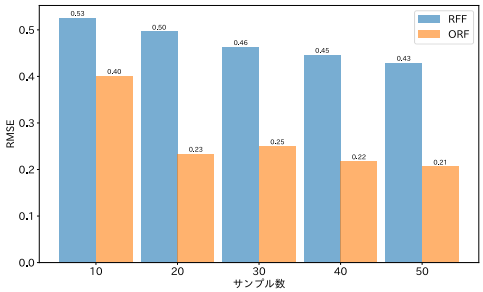

図 4　ベンチマークデータでの予測誤差の比較.

は，ロボットは初期位置 $(x, y) = (0, -4)$ からゴール地点 $(x, y) = (0, 4)$ を目指して進む．歩行者の行動は ORCA によって生成されており，歩行者の初期位置はエピソードごとに半径 4m の円上にランダムに設定される．歩行者は周囲の歩行者とロボットから影響を受け，位置や速度によって挙動が変化する．このシミュレーション環境のロボットと歩行者の動作例を図 5 に示す．評価は 500 パターンのテストケースを用いて行い，RFF と ORF の特徴量のサンプル数を 10~50 まで 10 ずつ変化させながら，予測誤差とタスク成功率についての比較を行う．RPSR を構成する観測と行動は，以下のように設定した．

- 観測：歩行者とロボットの位置と速度データを観測として利用する．各歩行者，ロボットの観測はベクトル $(p_x^i, p_y^i, v_x^i, v_y^i)$ であり，i 番目の歩行者またはロボットにおいて，(p_x^i, p_y^i) は位置，(v_x^i, v_y^i) は速度を表す．
- 行動：ホロノミックな全方位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) を用いる．

5.2.1　予測誤差の比較

この比較実験では，事前学習後の予測性能を比較する．条件を揃えるために，ロボットが ORCA に従って行動した場合の性能を比較する．結果を図 6 に示す．この結果においても，一貫して ORF の方が誤差が小さくなっていることが分かる．サンプル数が増加するほど手法間の性能差は縮まるものの，サンプル数が少なくなるにつれて RFF 利用時の誤差は ORF 利用時に比べて大きく増加し，サンプル数が 10 の場合は RFF の方が 2 倍近い誤差がある．このことから，特にサンプル数が少ない場合は ORF が有効であることが分かる．

5.2.2　タスク成功率の比較

この比較では，強化学習後のタスク成功率を比較する．結果を図 7 に示す．この結果より，ほとんどの場合で，ORF のほうが成功率が高いことが分かる．また，RFF の方が成功率の振れ幅が大きく，サンプル数の変化に対して不安定な結果となった．サンプル数が 20 の場合，RFF の結果は他のサンプル数での結果と比較して最も高い性能を示しており，同サンプル数での ORF

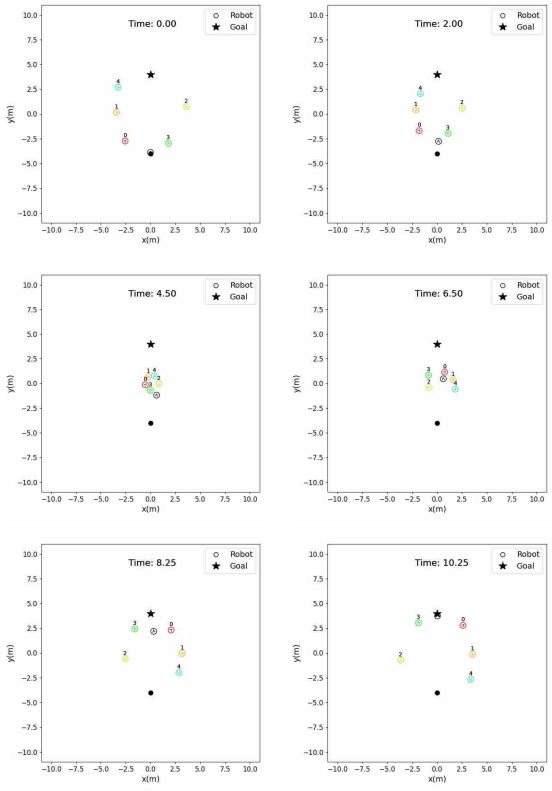

図 5 シミュレーション環境の動作例．黒色の円がロボットを表しており，その他の円は歩行者を表している．

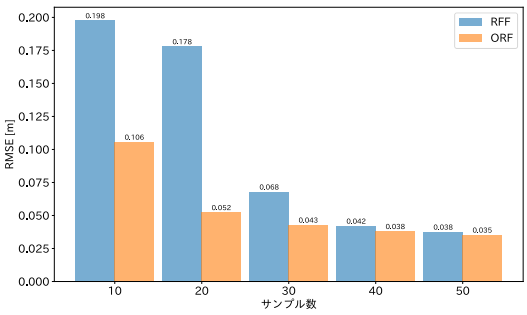

図 6　シミュレーション環境での予測誤差の比較.

の結果を上回っている．これは，強化学習時の探索などのランダム性による影響があるものと考えられる．

6.　結論

本研究では，直交ランダム特徴量を予測状態表現を用いた深層強化学習ベースの移動ロボットナビゲーション手法に導入した．比較実験により，直交ランダム特徴量を利用した場合のモデルの予測性能は，今回の実験設定において，一貫してランダムフーリエ特徴を利用した場合より上回ることが分かった．特に，直交ランダム特徴量を利用した場合はサンプル数が少ない場合でも，サンプル数が多い場合に対して性能の低下が少ないことが示された．これは，実ロボットに対して本手法を実装する場合に，歩行者の検出や自己位置推定などに使用する計算リソースやメモリを確保するた

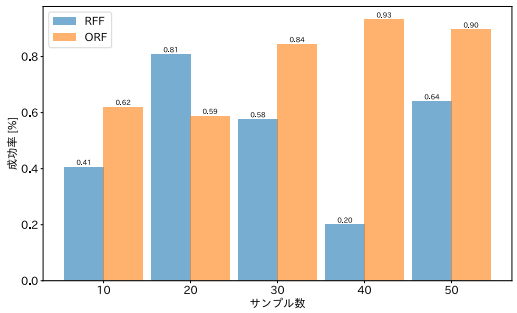

図 7　シミュレーション環境でのタスク成功率の比較.

めに有効に働く可能性がある．今後，実環境における実機での実験に取り組んでいく．

強化学習後のタスク成功率に関しては，ほとんどの場合で直交ランダム特徴量を利用した方が性能が高いことが示された．しかしながら，探索などのランダム性の影響により，ランダムフーリエ特徴量を利用した場合の結果が不安定になっていることが考えられるため，乱数のシード値を変えた場合の安定性の比較にも今後取り組んでいく．

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] 松本 耕平，河村 晃宏，安 琪，倉爪 亮，“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”，第 38 回日本ロボット学会学術講演会，pp. 3A3–04, 2020.

[2] 松本 耕平，河村 晃宏，安 琪，倉爪 亮，“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”，第 39 回日本ロボット学会学術講演会，pp. 3D1–06, 2021.

[3] 松本 耕平，河村 晃宏，安 琪，倉爪 亮，“グラフ畳み込み構造を持つ予測状態表現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”，第 40 回日本ロボット学会学術講演会，pp. 4G1–06, 2022.

[4] K. Matsumoto, A. Kawamura, Q. An, and R. Kurazume, “Mobile robot navigation using learning-based method based on predictive state representation in a dynamic environment,” in Proceedings of the IEEE/SICE International Symposium on System Integration (SII), pp. 499–504, 2022.

[5] K. Choromanski, C. Downey, and B. Boots, “Initialization matters: Orthogonal predictive state recurrent neural networks,” in Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[6] S. Singh, M. James, and M. Rudary, “Predictive State Representations: A New Theory for Modeling Dynamical Systems,” in Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI), pp. 512–519, 2004.

[7] B. Boots, A. Gretton, and G. J. Gordon, “Hilbert space embeddings of predictive state representations,” in Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI), pp. 92–101, 2013.

[8] A. Hefny, C. Downey, and G. Gordon, “An efficient, expressive and local minima-free method for learning controlled dynamical systems,” in Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), pp. 3191–3198, 2018.

[9] 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.

[10] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in Proceedings of the International Symposium of Robotic Research, pp. 3–19, 2011.

[11] A. Hefny, C. Downey, and G. J. Gordon, “Supervised learning for dynamical system learning,” in Advances in Neural Information Processing Systems (NeurIPS), pp. 1963–1971, 2015.

[12] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, “Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning,” inProceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 6015–6022, 2019.

[13] C. Chen, S. Hu, P. Nikdel, G. Mori, and M. Savva, “Relational graph learning for crowd navigation,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 10007–10013, 2020.