キュリオシティドリブンを用いた格闘ゲーム AI の提案

Proposal of fighting game AI with Curiosity driven

井上　秀保 † 高野　喜名 ‡ 欧陽　文文 † 伊藤　卓 ‡
Hideyasu Inoue Yoshina Takano Wenwen Ouyang Suguru Ito
ターウォンマット　ラック † 原田　智広 †
Ruck Thawonmas Tomohiro Harada
ruck@is.ritsumei.ac.jp

1. はじめに

これまで，ゲーム AI の分野において様々な強化学習の手法が利用されてきた．しかし，強化学習を利用する際の課題として，膨大な学習時間が必要になる場合があることや，学習の報酬設計が難しいことが挙げられる．これらの課題を回避するために，人間などの好奇心による自発的行動から発想を得たキュリオシティドリブンという手法が考案され，好奇心を定式化した．[1][2]

本論文ではこれを受け，キュリオシティドリブンを格闘ゲーム AI に適用し，従来手法を用いた格闘ゲーム AI と比べて有用性があるアルゴリズムを提案する．

2. 関連研究

2.1 キュリオシティドリブン (Curiosity driven)[1]

キュリオシティドリブンとは，人間などが持つ「モチベーション」や「好奇心」によって，環境の探索や目標へのアプローチを模索する概念を指す．機械学習におけるキュリオシティドリブンは，エージェントの行動から予測された状態と異なる状態に到達した場合に，大きな内部報酬 r_t^i を得ると定義されている．Policy based アルゴリズムにおける環境から得られる報酬を外部報酬 r_t^e と定義し，内部報酬 r_t^i と外部報酬 r_t^e の和を用いて方策 π を学習させる．内部報酬を生成する手法として，Intrinsic Curiosity Module が挙げられ，これを用いたゲーム AI の有用性が提唱されている．[1]

3. 提案

本章では，Policy based アルゴリズムの一つである Actor-Critic アルゴリズムをニューラルネットワークを用いて実装した格闘ゲーム AI に格闘ゲーム AI 用 ICM を実装し，外部報酬 r_t^e と内部報酬 r_t^i の和 r_t を用いて，得られる報酬の総和を最大化するよう方策 π を学習する手法を提案する．なお，使用する格闘ゲームは FightingICE version:4.30 とする．

3.1 提案アルゴリズム

図 1 に Actor-Critic アルゴリズムのニューラルネットワークの概要を示し，図 2 に提案アルゴリズムの概要を示す．Actor-Critic アルゴリズムは，政策反復法における Value の計算を Critic による Value の推定に置き換え，政策改善の判定を TD-error という確率変数を用いた判定に置き換えたものと考えられる．[3] この TD-error を求める際に利用する報酬 r_t を，ICM で生成した内部報酬 r_t^i と環境から得られる外部報酬 r_t^e の和とすることで，外部報酬 r_t^e のみを利用した場合と比べて多様な探索を行わせることを目標とする．

次に，各変数について説明する．状態 s の要素は，自分と相手キャラクターそれぞれの HP，エネルギー，X 座標，Y 座標，X 座標方向の移動速度と方向，Y 座標方向の移動速度と方向，一つ前の行動，相手キャラクターの次の行動開始までの残りフレーム数，波動拳コマンドによる当たり判定とダメージの計 141 要素とする．

外部報酬 r_t^e は，自分キャラクターの与ダメージ数を正の値，被ダメージ数を負の値とし，その合計とする．

θ は Actor のニューラルネットワークのパラメータとする．

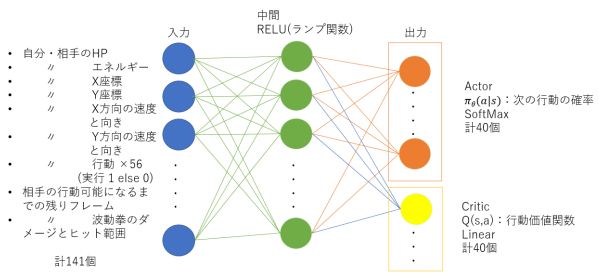

図 1: ニューラルネットワークの構成

以下に，提案アルゴリズムの処理手続きを示す．

1. エージェントは環境の状態 s_t を観測する．Actor は政策 $\pi_\theta(a|s)$ に従い行動 a_t を実行する．

† 立命館大学情報理工学部, Ritsumeikan University of Information Science and Engineering
‡ 立命館大学大学院情報理工学研究科, Graduate School of Information Science and Engineering, Ritsumeikan University

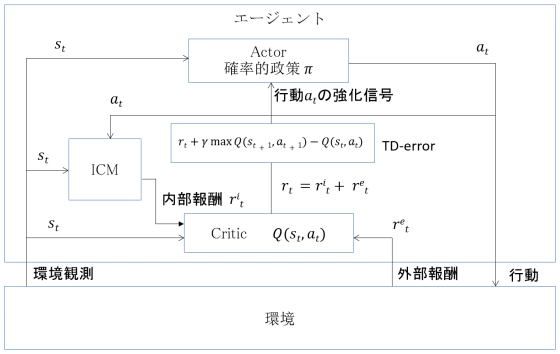

図 2: 提案アルゴリズムの構成

2. **ICM** は環境の状態 s_t，行動 a_t を受け取り順モデルで次の状態の予測 $\hat{s}_{t+1}$ を計算する．次の状態 s_{t+1} を受け取り，$\hat{s}_{t+1}$ と s_{t+1} のユークリッド距離を内部報酬 r^i_t とし，$\hat{s}_{t+1}$ が s_{t+1} に近づくよう順モデルを更新する．

3. **Critic** は内部報酬と外部報酬の和 $r_t = r^i_t + r^e_t$ を受け取る．次の状態 s_{t+1} を観測し，**Actor** への強化信号として以下の **TD-error** を計算する．

   (TD-error)$= r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)$

   $\gamma(0 \leq \gamma \leq 1)$ は割引率．

4. **TD-error** を用いて **Actor** の政策 π_θ を更新する．**Actor** のコスト関数は以下の通りである．

$$J(\theta) = \log(\pi_\theta(a_t|s_t)) * (TD - error) \quad (1)$$

   Actor のネットワークの更新は，$J(\theta)$ が最大になるように θ を更新する．

5. **TD** 法を用いて **Critic** のネットワークを更新する．

6. 規定ラウンド数まで **1.** から繰り返す．

3.2 格闘ゲーム AI 用 ICM

図 3 に従来の ICM の概要を示し，図 4 に格闘ゲーム AI 用 ICM の概要を示す．従来の ICM は各状態の画像データから抽出した特徴ベクトル $\phi(s_t)$，$\phi(s_{t+1})$ が，行動 a_t と関連性があるかを検証するために逆モデルを導入している．[1][2] しかし，FightingICE では内部データから状態のデータを直接受け取ることが可能であるため，状態の画像データから特徴ベクトルを抽出するステップを省略できる．よって，本論文で提案する格闘ゲーム AI 用 ICM では順モデルのみを実装している．

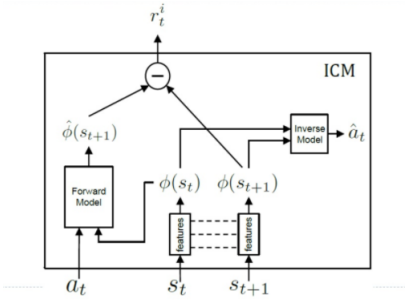

図 3: ICM の構成（[1] より引用）

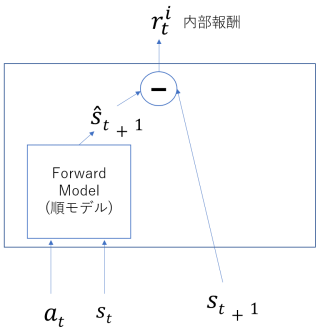

図 4: 格闘ゲーム AI 用 ICM の構成

4. おわりに

本論文ではキュリオシティドリブンを用いた格闘ゲーム AI の提案に留まり，実験による有用性の検証まで達成できなかった．今後は提案アルゴリズムで格闘ゲーム AI を実装し，キュリオシティドリブンを用いない格闘ゲーム AI と比較実験を行い，キュリオシティドリブンの有用性について検証する予定である．

参考文献

[1] Pathak D, et al. ”Curiosity-driven exploration by self-supervised prediction.” International Conference on Machine Learning (ICML). Vol. 2017. 2017.

[2] 加納由希夫, 鶴岡慶雅. ”内部報酬を自動生成する強化学習による一人用 RPG の自動攻略.” ゲームプログラミングワークショップ 2017 論文集 2017 (2017): 219-225.

[3] 木村元, 小林重信. ”Actor に適正度の履歴を用いた actor-critic アルゴリズム: 不完全な value-function のもとでの強化学習.” 人工知能学会誌= Journal of Japanese Society for Artificial Intelligence 15.2 (2000): 267-275.