2Q-1

RBF ネットワークと強化学習的手法による評価関数を用いたゲーム AI

成田卓也 †　　佐藤晴彦　　小山聡　　栗原正仁

北海道大学大学院情報科学研究科 ‡

1　はじめに

近年，技術の発達とともにビデオゲームは著しく発展を続け，またその種類も多種多様と化してきた．また，それに付随してゲームを行うプレイヤーの能力や趣向も多様化している．そうした中で，ライトユーザを如何にして確保，そして繋ぎ止めるかということが大きな課題の一つとなっている．特に携帯電話を中心としたソーシャルゲーム，カジュアルゲームの市場が拡大しており，ライトユーザの多くがそちらへと流れることとなってしまっている．そこで，一度確保したライトユーザを如何にしてコアユーザへと変化させていくかということが大切になってくる．

本研究では，ビデオゲームの主要なジャンルの一つである SRPG(Simulation Role Playing Game) を対象としてこの課題への一つの解決策を提案する．まず，SRPG においてライトユーザがコアユーザへとなっていくには，プレイヤー自身の上達が必要不可欠である．しかし，現在使われているゲーム AI のほとんどがライトユーザを上達させることを踏まえた設計を行っていない．そのため，多くのライトユーザがコアユーザへとならずにそのまま去って行ってしまっていると考えられる．そこで，ライトユーザの上達を促すための行動選択を行うゲーム AI を設計する必要がある．またこの行動選択を行うためにはゲーム中の各状態の評価値を利用することが必要であると考えられる．しかし、一般的な SRPG において，その状態の総数は古典的なゲームと比べても非常に多く，またゲーム自体もより複雑であるため，人手で有意な評価関数を設計することは非常に困難である．そこで本研究では機械学習の手法を用いて評価関数を自動的に学習していく方法を提案する．

以降，2 章では評価関数の学習に関しての関連研究をまず挙げる．3 章では対戦を行いながら自動的に学習する評価関数の設計について提案する．最後に 4 章でまとめと今後の展望について述べる．

Game AI with Evaluation Function based on RBF Network and Reinforcement Learning
†Takuya NARITA
‡Graduate School of Information Science and Technology, Hokkaido University

2　関連研究

2.1　TD-GAMMON

TD-GAMMON は Tesauro によるコンピュータバックギャモンのシステム [1][2] である．特徴として，評価関数に三層のシグモイド型ニューラルネットワークを利用し，その学習に強化学習の手法の一つである $TD(\lambda)$ が使われている．学習は自分自身を対戦相手として行われ，それを繰り返すことで人間のトッププレイヤーとも渡り合えるようになっている．

2.2　RBF ネットワーク

Radial Basis Function(RBF) ネットワークはニューラルネットワークの一種で，中間層に RBF を用い，出力層は各 RBF ユニットの重み付き和になる．この RBF は中心点からの距離にのみ依存する関数であり，主にガウス関数 $\phi_i(\mathbf{x}) = \exp\left(-\frac{|\mathbf{c}_i-\mathbf{x}|^2}{\sigma_i^2}\right)$ がよく使われている．RBF ネットワークは中間層のユニット数が十分に多ければ次のような形で任意の連続関数を近似することが可能である．

$$f(\mathbf{x}) = \sum_{i=1}^{N} w_i \phi_i(\mathbf{x}) \tag{1}$$

また RBF としてガウス関数を用いた場合には，各ユニットの出力は自身の中心点と入力点の距離が小さい場合のみ大きな値になる．

3　提案手法

評価関数の学習には強化学習の手法の一つである $TD(\lambda)$ と RBF ネットワークを組み合わせて行う．$TD(\lambda)$ とニューラルネットワークを組み合わせるという点では TD-Gammon と同じではあるが，学習が進んだ後では，シグモイド型ニューラルネットワークの中間層に比べると RBF ネットワークのほうが二次利用することが容易であることから，シグモイド型では無く RBF ネットワークを用いることとした．本研究におけるネットワークの入力ベクトルはゲーム中の各状態の特徴ベクトルであり，各 RBF ユニットの平均は入力ベクトル空間内の点を表しているので，それらの点をゲームを代表する特徴点とみなすことが出来る．これらの代表点

は今後 AI の行動選択について考えていくうえで利用が可能である．

それでは具体的な評価関数の学習について述べていく．ある時刻 t での状態 s_k の評価関数 $V_t(s_k)$ を次のように定義する．

$$V_t(s_k) = \sum_i w_i \exp\left(-\frac{|\mathbf{c}_i - \mathbf{x}_k|^2}{\sigma_i^2}\right) \tag{2}$$

ここで，$\mathbf{x}_k$ は状態 s_k の特徴ベクトルであり，ゲームに関するヒューリスティックを基にして決定している．$\mathbf{c}_i$ と σ_i^2 はそれぞれ中間層の RBF ユニット ϕ_i の平均と分散であり，w_i は各ユニット ϕ_i の重みになっている．各時刻 t でのこれらの各パラメータの更新は次の式にしたがって行う．

$$\Delta w_i = \alpha\delta_t \sum_{k=1}^{t} \left\{\lambda^{t-k}\phi_i(\mathbf{x}_k)\right\} \tag{3}$$

$$\Delta c_{ij} = \alpha\delta_t \sum_{k=1}^{t} \left\{\lambda^{t-k}\phi_i(\mathbf{x}_k)\cdot\frac{c_{ij}-x_{kj}}{\sigma_i^2}\right\} \tag{4}$$

$$\Delta \sigma_i = \alpha\delta_t \sum_{k=1}^{t} \left\{\lambda^{t-k}\phi_i(\mathbf{x}_k)\cdot\frac{|\mathbf{c}_i-\mathbf{x}_k|^2}{\sigma_i^3}\right\} \tag{5}$$

ただし $\delta_t = r_{t+1}+\gamma V_t(s_{t+1})-V_t(s_t)$，$\phi_i(\mathbf{x}_k) = \exp\left(-\frac{|\mathbf{c}_i-\mathbf{x}_k|^2}{\sigma_i^2}\right)$ である．ここでの δ_t は $TD(\lambda)$ における TD 誤差であり，r_{t+1} は対戦に勝利した場合には +1，敗北した場合には −1，それ以外の場合は 0 となる．α，γ，λ は $[0:1]$ の定数であり，それぞれ学習率，報酬の減衰率，過去の状態の評価値をどれだけ考慮するかの値となる．

本研究において中間層のユニットの数を学習前に予め決定することは困難である．そこで，minimal resorce-allocating network(M-RAN)[3] の手法を参考にし，中間層のユニットを動的に追加および削除を行う方法を提案する．まず追加の手法について述べる．初めに次の基準を定義する．

$$I_i = \begin{cases} 1 & |\mathbf{c}_i - \mathbf{x}_k|^2 \le (d_c\sigma_i)^2 \\ 0 & else \end{cases} \tag{6}$$

d_c は距離定数であり，ユニット ϕ_i の平均 $\mathbf{c}_i$ を中心とした半径 $(d_c\sigma_i)$ の超球の中に $\mathbf{x}_k$ が入っているかどうかを判定している．これを基に次の基準を満たす場合ユニットを追加する．

$$\delta_t > d_{th} \text{かつ} \sum_i I_i = 0 \tag{7}$$

d_{th} は TD 誤差に関する閾値である．このとき最近のユニットの平均を $\mathbf{c}_{nr}$ として，追加されるユニット ϕ^* に関するパラメータを次のようにする．

$$\mathbf{c}^* = \mathbf{x}_k \tag{8}$$

$$\sigma^* = \frac{1}{d_c}|\mathbf{c}_{nr} - \mathbf{x}_k| \tag{9}$$

$$w^* = \delta_t \tag{10}$$

続いて削除の手法について述べる．各ユニット ϕ_i の重み付き出力を $o_i = w_i\phi_i(\mathbf{x}_k)$ として，各ユニットの出力への寄与率を $r_i = \left|\frac{o_i}{o_{max}}\right|$ とし，寄与率に関する閾値を r_{th} とする．ここで N 回連続で $r_i < r_{th}$ となったユニット ϕ_i を削除する．

以上の手法を使い，ϵ-greedy で手を選択しながら対戦を行うことで評価関数を学習する．

4 まとめと今後の展望

対戦を行いながらプレイヤーを上達させるための AI の行動選択を実現するための一環として，RBF ネットワークと $TD(\lambda)$ を組み合わせ，対戦を行いながら評価関数を自動的に学習していく方法を提案した．しかし，学習した評価関数の性能評価はまだ出来ていないため，他の手法で作成した AI と比較しての性能評価が今後の課題となる．また，学習した評価関数を用いた，プレイヤーを上達させるための行動選択方法や，学習後の RBF ユニットの具体的な二次利用の方法も今後の課題として考えていくべき点である．

参考文献

[1] Gerald Tesauro. Practical issues in temporal difference learning. *Machine Learning*, Vol. 8, pp. 257–277, 1992.

[2] Gerald Tesauro. Temporal difference learning and td-gammon. *Commun. ACM*, Vol. 38, pp. 58–68, 1995.

[3] Lu Yingwei, N. Sundararajan, and P. Saratchandran. A sequential learning scheme for function approximation using minimal radial basis function neural networks. *Neural Computing*, Vol. 9, pp. 461–478, 1997.

[4] 原田智寛. Orthogonal least squares 法を用いた逐次学習法. Master's thesis, 北海道大学大学院情報科学研究科, 2006.