F-013

オリジナルゲームの強化学習型 CPU
Reinforcement Learning based CPU for an Original Game

中村　杏
Kyo Nakamura
法政大学情報科学研究科情報科学専攻
Email:kyo.nakamura.5h@stu.hosei.ac.jp

Abstract

In this research, I make an original game and propose a reinforcement learning based CPU for a player in the original game. The Monte Carlo method was used as a structure of the reinforcement learning and a filtering mechanism was applied to solve huge state space. The reinforcement learning based CPU player fights against a rule base CPU player in the original game. As a result of 2,000 times of play, the winning rate of the reinforcement learning based CPU reached an average of 59%.

1. 序論

対戦型ゲームにおける人工知能の研究は昔から盛んであり，1997 年に IBM のコンピュータであるディープブルーがチェスにおいて当時の世界チャンピョンに勝利を収めたのは有名な話である[1]．しかしながら，チェスや将棋のような昔から存在しており，戦術が確立されているものとは違い，一般的に市販されているそのメーカー独自の対戦型ゲームの CPU は強いものとは言いがたい．

これらの問題を解決するため，本論文では CPU 開発のアプローチとして強化学習型 CPU の実装を提案する．対戦結果から自己判断して学習する強化学習型 CPU ならば，戦術が確立されていなくてもよいため，通常用いられるルールベース型よりも強くなる可能性がある．状態空間に関しては，フィルタリングの技術を用いて解決を図る．

本研究では市販ゲームにあるようなオリジナルの対戦型ゲームを作成する．その後，ルールベースによる CPU の上に強化学習の仕組みを組み込み，強化学習型 CPU を作成する．強化学習の仕組みとしてモンテカルロ法を用いる．

2. オリジナルゲームのシステム

2.1. オリジナルゲームのルール

本オリジナルゲームは 30 枚のカードからなる「デッキ」，デッキからカードを引いてストックする「手札」，そしてその手札からユニットを召喚する場である「フィールド」によって構成される 2 人対戦型カードゲームであり，フィールド上に存在する相手マスターの HP を 0 にすると勝利となるものである．

相手マスターの HP を減らすためには、まず手札から「ユニット」を「フィールド」に出す。そのユニットをマスターの近くまで移動させ、攻撃する。このゲームのポイントはいかに自分のマスターを守り、相手マスターに攻撃するか，である．

2.2. オリジナルゲームの問題点

本オリジナルゲームを CPU 実装の観点から見ると，以下の注目するべき問題点が挙げられる．

- 膨大な状態空間
- 部分観測となる
- 学習に用いる教師が存在しない

以上の点を踏まえ，本研究では強化学習法を用いる．

3. 強化学習型 CPU 実装

3.1. 全体のサイクル

1) 場面取得
CPU 自身のターン開始時にゲームの場面状態を取得する．

2) 行動生成
場面取得時の各プレイヤの手札，デッキ，ユニット等の要素の組み合わせを現状態と定義する．この現状態から CPU が行うことが可能な全ての行動一つ一つを Order というオブジェクトとして生成する．状態に Order を適用することで状態は 1 ステップ進む．現状態を S，現在の時間を t とすると，次状態は以下の式で表せる．

$$S_{t+1} = Order(S_t) \quad (1)$$

3) 評価関数
場面取得にて取得した現状態と，その現状態に各 Order を適用させ，取得した次状態の二つをこの評価関数にて評価する．この評価関数によって得られた点数が高い Order が良い行動と定義する．これを式で表すと以下のようになる．

$$Score = f(myself) - f(rival) \quad (2)$$

$$f(player) = MasterHp(player) + PlayerStatus(player) + \sum_{i=0}^{n} UnitStatus(player.unit[i]) \quad (3)$$

式中の myself はプレイヤ、rival は対戦相手を示す．(2)の式は評価関数の大本の式で，評価関数による最終スコアはプレイヤの状態評価から対戦相手の評価を引いたものとなることを表している．(3)の式は状態評価の式であり，式中の n はプレイヤが持つユニット数となる．したがって，状態評価としては主に，プレイヤのマスターの HP の残量，ユニットの数およびその状態，プレイヤの状態（手札の数など）といった要素が評価に関わることとなる．

4) フィルタリング

Supervisor: Runhe Huang, Professor

フィルタリングについては次節にて説明する．

5) 行動実行

(1)から(4)までのサイクルにて決定した一連の行動を実際に行う．

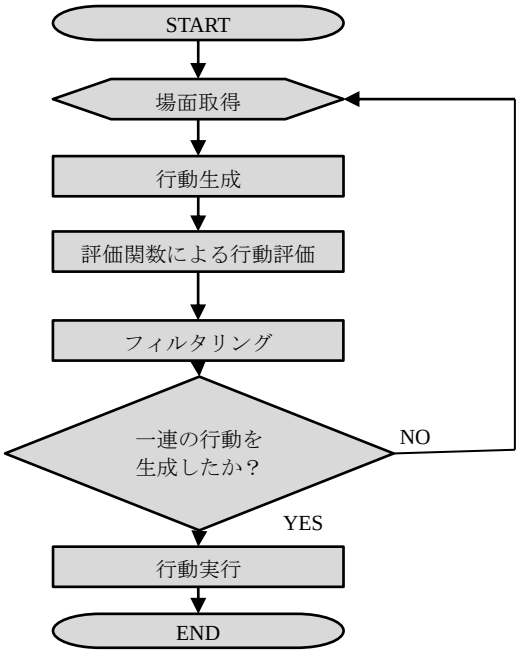

図 1　CPU の処理フローチャート

3.2. パーティクルフィルタ

本オリジナルゲームの膨大な状態空間に対処するため，フィルタリング技法としてパーティクルフィルタ[2]を応用したものを用いることにした．

まず，パーティクルをプレイヤの行動とみなす．次に，尤度計算をゲームの評価関数と考える．つまり，評価関数によって高い点数を出した Order を残し，低い値の Order を破棄する．このとき，残す Order は一つではなく複数のこすことで，次の一手で評価が大きく伸びる Order についても考慮できるようにする．このフィルタリングにおいて使用する点数は，次状態の点数から現状態の点数を引いた値となる．高い値の Order を適応した次状態 S_{t+1} において，取ることができる行動(Order)を CPU が作り，評価関数にて行動を適用し状態 S_{t+2} を取得する．そして再度フィルタリングを行う．このサイクルを繰り返すと，最も有効な一連の行動(Orders)が取得できる．その結果，1 ターンに行う一連の行動を決定できる．

3.3. 強化学習法

今回用いる強化学習法はモンテカルロ法を応用したものである．

まず，対戦結果やお互いのプレイヤの行動(Order）を記録したリプレイデータを読み取り，学習データベースにその情報を書き込む．無数のリプレイデータから作られた学習データベースを元に，勝利したプレイヤの行動評価を高く，敗北したプレイヤの行動評価を低くなるように評価関数を書き換えていく．評価関数によって得られる行動評価を Q(s, a)で表し（ここで a は状態 s で行う行動である），α を学習による点数とすると以下の式になる．

$$Q(s,a) \leftarrow Q(s,a) + \alpha \qquad (4)$$

4. 実験と結果

第 3 章にて作成したルールベースと強化学習法を組み合わせた CPU（以下強化学習型 CPU とする）をルールベースのみの CPU と対戦させ，オリジナルゲームにおいての強化学習による学習の有用性を評価する．

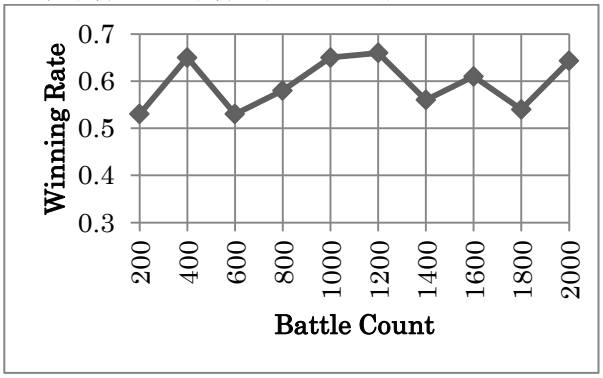

図 2　強化学習型 CPU の勝率の推移

上記の図は強化学習型 CPU とルールベース CPU を先に示した設定に基づいて対戦させた際の強化学習型 CPU の勝率の推移である．横軸は対戦したゲーム数であり，縦軸は強化学習型 CPU の勝率を表す．

グラフを見ると，初期の 200 ゲームの勝率ではおおよそ 53%であったのが 2000 ゲーム後には 64%ほどに上昇していることが確認できる．しかし，400 ゲーム時や 1800 ゲーム時などに勝率の振動が見られ，2000 ゲームほどの試行回数では勝率は収束しなかった．しかしながら，勝率は振動するものの，強化学習型 CPU は一度もルールベース CPU に負け越しておらず，強化学習による学習成果が出ていることが分かる．

5. 考察

本論文で提案する強化学習型 CPU はフィルタリングや評価関数を更新する強化学習法を用いることで，ルールベースのみで構築された CPU よりも強くなることが可能であることを示した．

今回，本オリジナルゲームのあらゆる行動一つ一つを Order と呼ばれるオブジェクトとして扱うことで，強化学習法の一種であるモンテカルロ法を応用し，強化学習機構を実装した．このように，一つの行動を抜き出すことが出来れば，モンテカルロ法は様々なジャンルのゲームに応用が可能である．

また，今回の実装では強化学習型 CPU の思考時間は学習回数の増加に伴い，増えていった．そのため，今後の課題として，CPU の思考速度の上昇などが挙げられるだろう．

6. 参考文献

[1] ブルース パンドルフィーニ，鈴木知道(訳) “ディープブルー vs. カスパロフ”河出書房新社 (1998)

[2] 北川源四郎 “モンテカルロフィルタおよび平滑化について” 統計数理, Vol.44, No.1, pp. 31-48(1996)