

## 状況に応じて手加減を行うゲームエージェントの構築 Construction of a Game Agent Which Can Hold Back Depending on the Situations

内田 純平<sup>†</sup> 穴田 一<sup>†</sup>  
Junpei Uchida Hajime Anada

### 1. はじめに

ゲーム AI の研究が精力的に行われており、将棋や囲碁は人間のトッププロに迫る強さになりつつある。これらのゲームは完全情報ゲームであり、プレイヤーは互いに全ての情報を手にいれることができる。一方、一部の情報が得られないゲームは不完全情報ゲームと呼ばれる。Long ら[1]によると不完全情報ゲームは大きく 2 種類に分けている。1 つはゲームが進み徐々に情報が明らかになるトリック型ゲームである。2 つ目は、ポーカー型ゲームと呼ばれ、勝利決定までのプロセスでは明らかな情報開示のないゲームである。

トリック型不完全情報ゲームの研究も行われており、人狼知能[2]等でゲームログのデータ解析などが行われている。このような不完全情報ゲームの一つとして、トランプゲームの大貧民がある。大貧民をコンピュータにプレイさせる大会が UEC コンピュータ大貧民大会 (UEC-da) として、2006 年より開催されている[3]。また、アルゴリズム[4][5]の提案も行われている。

本研究では、多人数不完全情報ゲームであり、トリック型ゲームの一種である大貧民を取り扱う。UEC-da で提案されているエージェントは、人間と戦うことを想定していないため、エージェントは人間に適した強さでプレイすることが難しい。そこで本研究では、状況に応じて手加減を行う機能を構築し、UEC-da で提案されているエージェントに導入することで、相手に合わせたプレイを可能にした。

大貧民において人間のプレイヤーがカードを提出したときの既出カード、場のカード、各プレイヤーの手札数などの状況データから提出カードが勝負手かどうかを動的に判断し、その判断に応じて手加減を行える手法の構築を目的としている。

### 2. UEC 標準ルール

コンピュータ大貧民は、カードゲームの 1 種である大貧民を計算機上で行うゲームである。電気通信大学において大会が毎年開催されており、統一ルールや各種プログラム、開発環境が公開されている。本研究でもこの枠組みを利用する。

### 3. 提案手法

本研究では、メタヒューリスティックな戦略を基に提出を行う kou2[6]をベースに手加減を行うエージェントの構築を行った。それに伴い、手札や場のカード、既に出たカードなど環境による情報から推定対象プレイヤーの評価を行う評価器の構築をした。また、評価器の推定結果に基づいて、

手加減対象プレイヤーを相対的に評価する方法を構築し、それに基づいて手加減の判断、手加減をした提出を行う。

#### 3.1 評価器

本研究における評価器は 大富豪、富豪を富裕クラス、平民を庶民クラス、貧民と大貧民を貧困クラスとした 3 つのクラスを推定する Feed Forward Neural Network であり、Soft Max 関数による確率を出力とする。

##### 3.1.1 入力データ

本研究では、UEC コンピュータ大貧民大会出場エージェント計 5 体による 2000 回の対戦ログデータを使用した。

推定対象者それぞれの状態を表す情報として、前回の順位、手札数、総合順位、自分の番がきた回数を使用している。また、共通した情報として場のカード、既出カード、場の役、場のマークを使用している。自分の番がきた回数についてはステージスの公式に基づいて Binning したものを特徴量とした。

##### 3.1.2 パラメーター

評価器には、3 層の Feed Forward Neural Network を使用した。以下の表 1 に使用したパラメーターを示す。

表 1 評価器のパラメーター

|            |                     |
|------------|---------------------|
| 入力次元数      | 118                 |
| 出力次元数      | 3                   |
| 中間層数       | 1                   |
| 中間ユニット数    | 236                 |
| 活性化関数      | Swish 関数            |
| 出力次元の活性化関数 | Soft Max 関数         |
| 損失関数       | Cross Entropy Error |
| 学習方法       | SGD                 |
| 学習率        | 0.001               |
| epoch      | 10000               |

#### 3.2 カード提出方法

カード提出は、手加減提出と通常提出がある。通常提出では、kou2 をベースにしたメタヒューリスティック戦略を用いた提出を行う。一方、手加減提出では、3.2.1 で説明する評価式に基づいた提出を行うことで、手加減を行う。

##### 3.2.1 評価式

評価器によって出力された各クラスの確率を用いて、得点の期待値を表す Rank を算出する。Rank は、富裕層になる確率と 3.5、庶民層になる確率と 2.0、貧困層になる確率と 0.5 の積を計算し、和を取った値であり、評価式で利用する。

手加減対象のプレイヤーを  $i$  とするとき、手加減対象プレイヤー  $i$  の評価を次式で行う。

$$Rate_i = \Delta Rank_i - \frac{1}{4} \sum_{j=1, j \neq i}^5 \Delta Rank_j \quad (1)$$

<sup>†</sup> 東京都市大学大学院 総合理工学研究科 情報専攻  
Tokyo City University, Department of Information Science,  
Graduate School of Science and Engineering

ここで,  $\Delta Rank_i$  は手加減対象プレイヤー  $i$  のカード提出前後の手加減対象プレイヤーの Rank の変化を表している. 一方,  $\Delta Rank_j$  は手加減対象プレイヤー  $i$  のカード提出前後での手加減対象プレイヤー以外のプレイヤーの Rank の変化を表している.

3.2.2 手加減提出

手加減提出は, 手加減対象プレイヤーの Rate を算出し, 閾値を超えたときに行う. 以下図1にカード提出のフローチャートを示す.

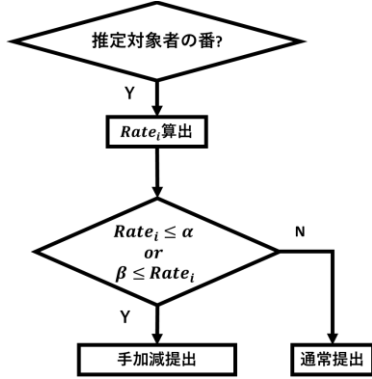


図1 カード提出のフローチャート

手加減を行うカード提出では, 提出可能なカード組を提出したときの Rate を全てで計算し, 最も Rate が高いカードを提出する. Rate は, カード提出によって相対的に手加減対象プレイヤーが有利になるかどうかを算出した値であり, これを最大化するような提出を行うことで手加減した提出を行う.

4. 結果

本研究では, 大会サンプルエージェント(default)1 体と提案エージェント(Sontaku)4 体で 1000 試合行い, UEC 標準ルールに従った獲得得点で提案手法の有効性を確認した. また, 提案エージェントが大会サンプルエージェントに対して状況に応じた手加減を行えているかどうかを確認するために, 常に手加減を行わない場合を設定 A, 常に手加減を行う場合を設定 B, 状況に応じた手加減を行う場合を設定 C として 3 パターンの実験を行った. また, それぞれの実験で  $\alpha = -0.025, \beta = 0.05$  のパラメータを使用した. 以下の図 2 から図 4 にそれぞれの実験の結果を示す. さらに, 表 2 に各設定におけるエージェント毎の得点と標準偏差を示す.

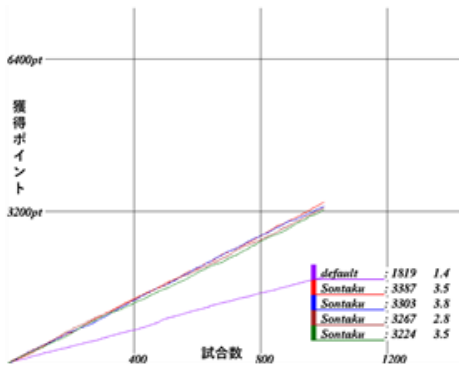


図2 常に手加減を行わない場合 (設定 A)

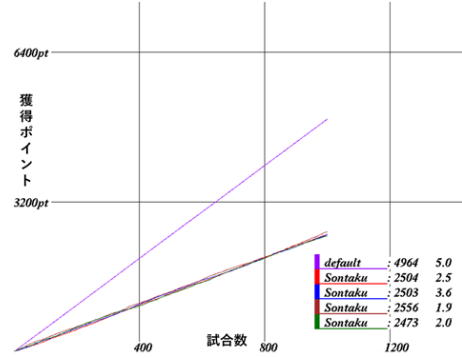


図3 常に手加減を行う場合 (設定 B)

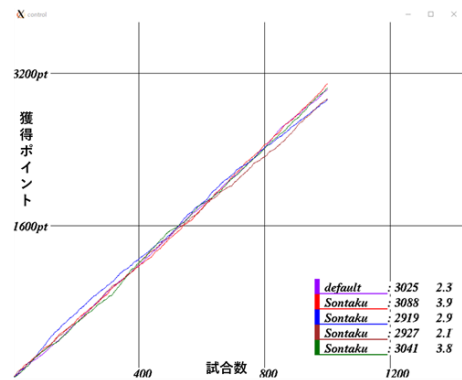


図4 状況に応じた手加減を行う場合 (設定 C)

表2 エージェントの得点

| 設定 | Default | Sontaku1 | Sontaku2 | Sontaku3 | Sontaku4 | 標準偏差        |
|----|---------|----------|----------|----------|----------|-------------|
| A  | 1819    | 3387     | 3303     | 3267     | 3224     | 592.9       |
| B  | 4964    | 2504     | 2503     | 2556     | 2473     | 982.4       |
| C  | 3025    | 3088     | 2919     | 2967     | 3041     | <b>66.2</b> |

表2の標準偏差を見ると状況に応じた手加減を行う場合が最も小さいことがわかる. また, 図4より提案エージェントは手加減対象エージェントの強さに合わせたプレイを行っていることが分かる. このことから, 状況に応じた手加減を行うことで, 接戦出来ることが確認できた.

参考文献

- [1] Long, J., Sturtevant, N.R., Buro, M. and Furtak, T.: Understanding the Success of Perfect Information Monte Carlo Sampling in Game Tree Search, Proceeding of the 24<sup>th</sup>. AAAI conf., AAAI, pp. 134–140 (2010).
- [2] 人狼知能プロジェクト Artificial Intelligence based Werewolf, <http://www.aiwolf.org/> (2020).
- [3] UECda 運営委員会: UEC コンピュータ大貧民大会, <http://uecda.nishino-lab.jp/> (参照 2019).
- [4] 飯田伸也, 藤田 悟: 大貧民におけるシミュレーション・バランシングを用いた方策学習, 第 77 回全国大会講演論文集, 人工知能と認知科学, pp. 93-95 (2015).
- [5] 大渡勝己, 田中哲朗: 方策勾配を用いた教師有り学習によるコンピュータ大貧民の方策関数の学習とモンテカルロシミュレーションへの利用, 情報処理学会研究報告, 2016-GI-35, No. 10, pp. 1-8 (2016).
- [6] 田頭幸三, 但馬康宏: コンピュータ大貧民におけるヒューリスティック戦略の実装と効果, 情報処理学会論文誌, vol157, no11, pp. 2403-2413 (2016)