線形関数近似によるトリックテイキングゲームのQ学習

齋藤 雄太[1,a)]　鶴岡 慶雅[2]

概要：事前知識を用いない多人数不完全情報ゲームの AI の学習は、人工知能を現実世界の問題に応用する上で非常に重要な課題の一つである。本研究では、多人数不完全情報ゲームの一種であるトリックテイキングゲームの行動価値観数を線形関数で近似し、Q 学習を行った。その結果、トリックテイキングゲームに Q 学習を適用することで単純なルールベースのプレイヤに勝る結果が得られること、自己対戦による学習を行うことで、ランダムプレイヤによる学習を行った時よりも学習結果が向上することを示した。

Q-learning for Trick-Taking Card Games with linear function approximation

Yuta Saito[1,a)]　Yoshimasa Tsuruoka[2]

Abstract: Learning the AI of a multiplayer imperfect information game without prior knowledge is one of the important challenges toward the application of AI to real-world problems. In this study, we attempted to learn action-value functions for trick-taking games, which is a kind of multiplayer imperfect information games. We built linear action-value functions using Q-learning. Experimental results show that the player built by Q-learning is superior to a simple rule-based player and that learning with self-play is better than using a random player as the opponent.

1. はじめに

人工知能の研究の中で、ゲーム AI という分野の最大の特徴は、状態や行動の定義、目標を厳密かつ自由に定義することが可能であり、その評価がしやすいということである。このような特徴から、ゲーム AI についての研究は、より不確実で評価の曖昧な現実世界の問題を解決する人工知能の開発への応用するために非常に重要な研究分野である。ゲーム AI の中でも、不完全情報ゲームの AI は、状態を正確に把握することのできない現実の問題に近い問題設定であるため、不完全情報ゲームの AI の研究は他の人工知能をもちいる分野への応用に大いに役立つと考えられる。

[1] 東京大学工学部電子情報工学科
Department of Information and Communication Engineering, The University of Tokyo
[2] 東京大学大学院工学系研究科電気系工学専攻
Depertment of Electrical Engineering and Information Systems, Graduate School of Engineering, The University of Tokyo
[a)] ysaito@logos.t.u-tokyo.ac.jp

完全情報ゲームの AI については、ゲーム画面の画像とスコアのみから強化学習を行い、人間以上のスコアを出すエージェントを生み出す [1] など、抽象化やヒューリスティクス、教師データを用いない研究が進んでいるが、不完全情報ゲームの AI はヒューリスティクスやモデルに頼ったものが多い [2]。ヒューリスティクスやモデル、教師データを用意することにはコストがかかり、ルールがわずかに異なるだけで対応できなくなるという問題がある。したがって、不完全情報ゲームにおいても、事前知識や教師データのない強化学習によって最適な行動をする AI を生成できるようにすることが重要だと考えられる。

Bowling らによって不完全情報ゲームの一つである 2 人リミットテキサス・ホールデム (Heads-Up Limit Texas Hold'em) のナッシュ均衡解が Counterfactual Reguret Minimization+ (CFR+) という手法によってよって求められた [3]。しかし、CFR+は不完全情報ゲームのゲーム木を完全探索する手法であるため、その時間、空間計算量の大きさを考えると解くことのできる不完全情報ゲームは

一部であり、トリックテイキングゲームを含む多くの不完全情報ゲームにこの手法を用いることはできない。

Heinrich らは、仮想自己対戦 (Fictitious Self-Play, FSP) に Deep Q-Network (DQN) を用いた Neural Fictitious Self-Play (NFSP) という手法を用いて、リミットテキサス・ホールデムについて CFR+の求めたナッシュ均衡解に匹敵する成果をあげた [2]。

本研究では、CFR+を用いて解くことができない多人数不完全情報ゲームの 1 つであるトリックテイキングゲームに対して、その状態評価関数を NFSP によって事前知識なしに学習することを目標に、線形関数近似による Q 学習を行った。これは NFSP に用いられている手法の 1 つである Q 学習をトリックテイキングゲームに用いたときの学習の効果を確認するためである。

トリックテイキングゲームのような CFR+で解くことのできない不完全情報ゲームを NFSP を用いて事前知識なしで学習できることを示すことで、より複雑なトリックテイキングゲームや他の不完全情報ゲーム、ひいては現実的な問題などの、従来までは人の手による状態の抽象化やルールベース化、教師データが必要であったゲームについて、NFSP によって事前知識を用いないような学習を行うことができる可能性を示すことができると考えられる。

2. 関連研究

2.1 Q 学習

Q 学習 [4] とは、マルコフ決定過程で表される環境において、与えられる報酬のみを元により多くの報酬を得られるような行動を学習する強化学習の手法である。まず、マルコフ過程においてエージェントが時不変な方策 π をとるとする。ここで π は各状態 s について行動 a を与える関数とする。ここで、あるステップ t における利得 V_t を

$$V_t = \sum_{i=0}^{\infty} \gamma^i r_{t+i} \tag{1}$$

と定義する。ただし、r_t はステップ t における報酬であり、γ は割引率と呼ばれる $0 < \gamma < 1$ を満たす定数である。時不変な政策 π に従って行動した時の V_t の期待値を $Q^{\pi}(s,a)$ とすると、

$$Q^{\pi}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s,a,s')\, Q^{\pi}(s', \pi(s')) \tag{2}$$

となる。ここで、$R(s,a)$ は報酬、$P(s,a,s')$ は遷移確率である。このとき、

$$Q^{\pi^*}(s,a) \geq Q^{\pi}(s,a) \quad for\ all\ s \in S\ ,\ a \in A\ and\ \pi \tag{3}$$

を満たす π^* が少なくとも 1 つ存在する。これを最適方策と呼ぶ。Q 学習では、最適方策をとったときの行動価値関数 $Q^*(s,a) = Q^{\pi^*}(s,a)$ を関数 $Q(s,a)$ で近似することによって学習を行う。

関数 $Q(s,a)$ は以下の式

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha_t \left(r_t + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t) \right) \tag{4}$$

によって更新される。ここで、α_t は学習率と呼ばれる。学習率 α_t が

$$\sum_{t=1}^{\infty} \alpha_t = \infty \quad , \quad \sum_{t=1}^{\infty} {\alpha_t}^2 < \infty \tag{5}$$

を満たすとき、関数 $Q(s,a)$ は更新によって確率 1 で $Q^*(s,a)$ に収束する。これを Q 学習の収束定理と呼ぶ。

Q 学習の収束定理より、全ての行動を十分な回数選択すれば関数 $Q(s,a)$ は収束するが、より収束を早めるために以下のような行動選択方法が提案されている [5]。

- ε-greedy 選択：確率 ε でランダムに行動を選択し、それ以外では最大の Q 値をもつ行動を選択する。
- ボルツマン選択：パラメータ T を用いて $e^{Q(s,a)/T}$ に比例した確率で行動を選択する。ただし、T は時間経過によって 0 に収束する。

2.2 関数近似

原理的には 2.1 節で述べた手法を用いることで最適な $Q(s,a)$ を求めることが可能だが、例えば将棋の状態数がおよそ 10^{70} であるように、ゲームの状態数は膨大であるため、そのような問題で厳密な最適行動価値関数を求めることは計算量的に難しい。このような問題に対して Q 学習を適用する場合、関数 $Q(s,a)$ に何らかの関数近似を用いる必要がある。

Q 関数をパラメータ $\boldsymbol{w}$ を使って $Q_{\boldsymbol{w}}(s,a)$ と関数近似することを考える。この場合、最大利得

$$V_{max} = r + \gamma \max_{a'} \{ Q_{\boldsymbol{w}}(s',a') \} \tag{6}$$

との 2 乗誤差を小さくするように以下の式でパラメータを更新する。

$$\begin{aligned} \boldsymbol{w} &\leftarrow \boldsymbol{w} - \frac{1}{2}\alpha \boldsymbol{\nabla}_{\boldsymbol{w}} \left(V_{max} - Q_{\boldsymbol{w}}(s,a) \right)^2 \\ &= \boldsymbol{w} + \alpha \left(V_{max} - Q_{\boldsymbol{w}}(s,a) \right) \boldsymbol{\nabla}_{\boldsymbol{w}} Q_{\boldsymbol{w}}(s,a) \end{aligned} \tag{7}$$

行動価値関数を状態ベクトル $\boldsymbol{x}(s,a)$ を用いて線形関数

$$Q_{\boldsymbol{w}}(s,a) = \boldsymbol{w} \cdot \boldsymbol{x}(s,a) \tag{8}$$

で近似する場合、$\boldsymbol{\nabla}_{\boldsymbol{w}} Q_{\boldsymbol{w}}(s,a) = \boldsymbol{x}(s,a)$ であるため、パラメータベクトル $\boldsymbol{w}$ の更新式は

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \alpha \left(r + \gamma \max_{a'} \{ Q_{\boldsymbol{w}}(s',a') \} - Q_{\boldsymbol{w}}(s,a) \right) \boldsymbol{x}(s,a) \tag{9}$$

となる。

2.3 Neural Fictitious Self-Play (NFSP)

仮想プレイ (Fictitious Play, FP) [6] は他のプレイヤの平均戦略を用いて多人数不完全情報ゲームのナッシュ均衡解を求める手法であり、これを一般化して Generalized Weakened Fictitiou Play (GWFP) [7] という手法が提案された。GWFP によって得られる平均戦略は 2 人ゼロ和ゲームなどの一部のゲームにおいてはナッシュ均衡解に収束することが知られている。

展開型ゲームではゲーム木の深さに対して指数的に戦略空間が増加するため、FP をそのままの形で適用するのは難しい。そこで、展開型ゲームに FP を適用するために Extensive-form Fictitious Play (XFP) [8] という手法が提案された。XFP では、ゲーム木の深さに対し線形な計算量で FP を実行することができる。

この XFP の平均戦略を教師あり学習で行い、最適応答戦略を強化学習で近似したものが Fictitious Self-Play (FSP) であり、FSP の強化学習に Deep Q-Network (DQN) [1] を、教師あり学習にニューラルネットワークをそれぞれ用いる手法が Neural Fictitious Self-Play (NFSP) である [2]。NFSP によって、不完全情報ゲームである 2 人リミットテキサス・ホールデムのナッシュ均衡解を前提知識を用いずに近似的に求めることに成功している。

3. 提案手法

本研究では、トリックテイキングゲームの状態価値関数を線形関数近似し、Q 学習を行うことを提案する。

トリックテイキングゲームとは、トランプゲームの分類の一つであり、欧米で人気のあるコントラクトブリッジ、ハーツ、スペードなどのゲームもこれに分類される [9]。ゲームのルールは単純であり、全プレイヤに同じ枚数の手札が配られた状態から、トリックと呼ばれるミニゲームを繰り返していく。各トリックでは、最初にカードを出すプレイヤ (リードと呼ばれる) から決められた順番にカードを一枚ずつ場に出していく。最も強いカードを出したプレイヤがそのトリックの勝者となり、次のトリックをリードする。全員のカードがなくなるまでトリックを続けた後、勝者を決める。勝者の決め方やカードの強さ、出すことのできるカードはルールによって異なる。

以上のようにトリックテイキングゲームには様々な種類があるが、本研究ではトリックテイキングゲームの基本的なルールを単純化して独自に定義したルールと、トリックテイキングゲームの代表的なゲームの 1 つであるハーツに対して線形行動価値関数を Q 学習する。

4. 実験

4.1 単純なルールにおける行動価値関数の Q 学習

4.1.1 実験方法

トリックテイキングゲームを単純化したルールとして、次のようにルールを定める。

- プレイヤの人数は 4 人。
- ジョーカーを除いたトランプ 52 枚を 1 人 13 枚ずつ配る。
- 最初のトリックをリードするプレイヤはランダムに決定する。
- リードされたカードと同じスートのカードを持つ場合、他のスートのカードを出すことはできない。
- リードされたカードと同じスートのカードを持たない場合、全てのカードを出すことができる。
- カードをリードするとき、出すカードに制限はない。
- カードのランクは A,K,...,2 の順に高いものとする。
- リードされたカードと同じスートのカードの中で最もランクの高いカードを出したプレイヤがトリックの勝者となる。
- 勝利したトリックの数を各プレイヤの得点とする。

表 1 状態ベクトル x の要素

特徴	次元数
相手 3 人が合計 n 枚持っているスートで、m 番目に強いカードをリードする	78
相手 3 人が合計 n 枚持っているスートで m 番目に強い、場の中で一番強いカードを l 番目に出す	234
既に場に出されたカードを除き、各スートで自分が持っているカードの強さの組み合わせ	8,191

状態 s と行動 a を抽象化して表 1 のような特徴量を持つ 8,503 次元のベクトル $\boldsymbol{x}(s,a)$ で表現し、Q 学習の関数を式 (8) のように線形近似してパラメータベクトル $\boldsymbol{w}$ の値を更新式 (9) に従って更新した。更新式のパラメータは $\alpha = 0.1$、$\gamma = 0.8$ とし、$\boldsymbol{w}$ は $\mathbf{0}$ で初期化し、報酬 r はそのトリックで勝利した際には 1、その他の場合には 0 とした。ランダムなプレイヤ、学習過程で生成した行動価値関数を用いたプレイヤを対戦相手に用いた場合についてそれぞれ 100,000 ゲームの対戦によって学習を行った。学習中の手の決定方法には $\varepsilon = 0.1$ とする ε-greedy 法を用いた。後者の自己対戦を行う場合については、100 ゲームの学習を行うごとに、学習中の Q 関数が最大となる手を選択するようなプレイヤを保存し、それまでに保存された全プレイヤとランダムプレイヤ 1 つの中から等確率でプレイヤを選ぶ作業を 3 回繰り返し、それらを対戦相手とした。

また、以下のような単純なルールで手を決定するルールベースのプレイヤを作成し、学習結果の評価に用いた。

- 最初にカードをリードする場合は手札からランダムに選択したカードを出す。
- リードされたカードと同じスートのカードを持っている場合
  - 場に出ている中で最も強いカードより強いカードがあれば、それらの中で最も弱いカードを出す。
  - 場に出ている中で最も強いカードより強いカードが

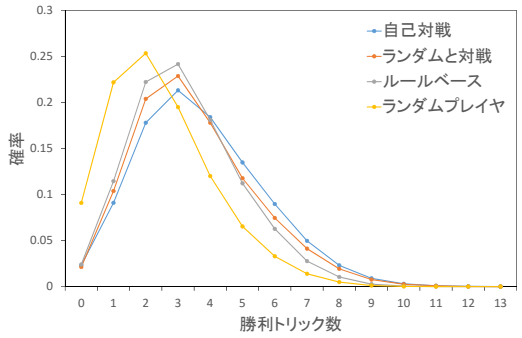

図 1　勝利トリック数の分布

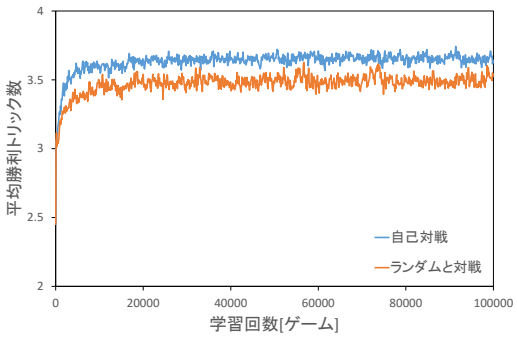

図 2　学習による平均勝利トリック数の変化

なければ、そのスートの中で最も弱いカードを出す。

- リードされたカードと同じスートのカードがない場合、手札からランダムにカードを選び、そのカードと同じスートの中で最も弱いカードを出す。

4.1.2　実験結果

結果を図 1、図 2 に示す。図 1 は、各プレイヤをそれぞれルールベースのプレイヤ 3 つと 100,000 ゲーム戦わせた時の勝利トリック数の分布を表す。各プレイヤの平均獲得トリック数はそれぞれ 3.69、3.49、3.25、2.47 であった。この結果から、学習によってルールベースのプレイヤより優れた結果が出ていることが分かる。

図 2 は、ルールベースのプレイヤ 3 つと 10,000 ゲーム対戦させた時の平均勝利トリック数を、それぞれの学習方法について学習を進めながらプロットしたものである。10,000 対戦の手札のパターンは各点において同じものを用いた。この結果より、自己対戦によって学習を行うことで学習の進行が早く、高い値で収束していることが分かる。しかし、結果にばらつきはあるものの、どちらの学習方法においても 20,000 ゲーム以降は勝利トリック数の平均に大きな変化はない。

表 2　状態ベクトル x の要素

特徴	次元数
相手 3 人が合計 n 枚持っているスートで、m 番目に強いカード k をリードする	4,056
相手 3 人が合計 n 枚持っているスートで m 番目に強い、場の中で一番強いカード k を l 番目に出す	12,168
既に場に出されたカードを除き、各スートで自分が持っているカードの強さの組み合わせ	8,191

4.2　ハーツの行動価値関数の Q 学習

4.2.1　実験方法

トリックテイキングゲームの一種であるハーツに対し、4.1 と同様の手法で線形行動価値関数の Q 学習による強化学習の実験を行った。ハーツのルールは以下のようなものとした。

- プレイヤの人数は 4 人。
- ジョーカーを除いたトランプ 52 枚を 1 人 13 枚ずつ配る。
- カードの点数はスペードの Q が 13 点、ハートのカードがそれぞれ 1 点、その他のカードは 0 点である。
- 最初のトリックはクラブの 2 を持っているプレイヤがクラブの 2 をリードする。
- リードされたカードと同じスートのカードを持つ場合、他のスートのカードを出すことはできない。
- 最初のトリックでは、点数が 0 のカードを持っていない場合を除き、点数が 0 でないカードを出すことはできない。
- 一度もハートが場に出ていない場合、手札が全てハートのカードである場合を除き、ハートのカードをリードすることはできない。
- カードのランクは A,K,...,2 の順に高いものとする。
- リードされたカードと同じスートのカードの中で最もランクの高いカードを出したプレイヤがトリックの勝者となる。
- 自分の勝利したトリックで出されたカードの点数の和を得点とし、より低い得点を目指す。

プレイヤ a の報酬は各トリックで獲得した点数 p_i として $r = \frac{1}{4}\sum p_i - p_a$ とし、特徴ベクトル $\boldsymbol{x}(s,a)$ の要素には表 2 のような 24,415 次元の要素を用いた。

また、以下のような単純なルールで手を決定するルールベースのプレイヤを作成し、学習結果の評価に用いた。

- 最初にカードをリードする場合は手札からランダムに選択したカードを出す。
- スペードの Q,K,A を出すことができ、出しても自分がそのトリックに勝たないことが確定している場合、スペードの Q,K,A を出す。スペードの Q,K,A を複数枚持っている場合、この優先順位で出す。
- リードされたカードと同じスートのカードを持っている場合

表 3 ルールベースプレイヤとの対戦結果

プレイヤ	得点が 0 となった割合	平均得点
自己対戦	0.39	4.37
ランダムと対戦	0.37	4.55
ルールベース	0.31	6.51
ランダムプレイヤ	0.13	9.24

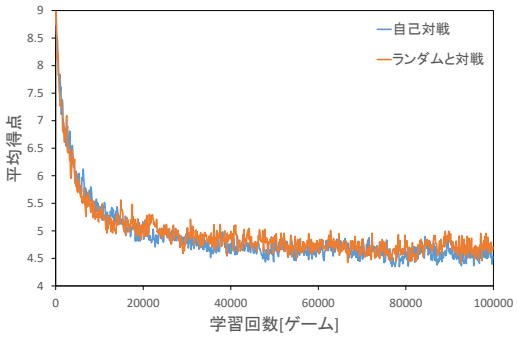

図 3 学習による平均勝利トリック数の変化

- 場に点数が 0 でないカードが出ている場合、場に出ている中で最も強いカードより弱いカードの中で最も強いカードを出す。場に出ている中で最も強いカードより弱いカードがない場合、そのスートで一番弱いカードを出す。
- 場に点数が 0 でないカードが出ていない場合、そのスートの中で最も強いカードを出す。

- リードされたカードと同じスートのカードがない場合、手札からランダムにカードを選び、そのカードと同じスートの中で最も強いカードを出す。

4.2.2 実験結果

結果を表 3、図 3 に示す。表 3 は、各プレイヤをそれぞれルールベースのプレイヤ 3 つと 100,000 ゲーム戦わせた時の得点が 0 だったゲームの割合と平均得点を表す。この結果から、ハーツのようなより複雑なルールでも、学習によってルールベースのプレイヤより優れた結果が出ていることが分かる。

図 2 は、ルールベースのプレイヤ 3 つと 10,000 ゲーム対戦させた時の平均勝利トリック数を、それぞれの学習方法について学習を進めながらプロットしたものである。この図より、単純なトリックテイキングゲームに比べて差異は少ないものの、自己対戦による学習の方が平均得点は低い値を示していることが分かる。

5. おわりに

以上の結果から、状態と行動を抽象化することで、単純なトリックテイキングゲームの行動価値関数を線形関数の形で学習することが可能であること、自己対戦を学習に用いることによってより良い結果が得られることが示された。線形関数近似によって簡単なルールベースのプレイヤを上回る結果が得られたことから、行動価値関数にニューラルネットワークを用いることで、さらなるプレイヤの強化が期待される。

また、同じトリックテイキングゲームの中でも単純化したトリックを取ることを目的とするルール、ハーツのようなトリックを取らないようにするルールの両方に Q 学習を適用した結果、どちらのルールにおいても似たような状態ベクトルを用いることで学習の成果が得られた。このことから、NFSP を用いて単純なトリックテイキングゲームの強化学習を行い、その有用性を示すことができれば、NFSP は他の多人数不完全情報ゲームにも有向な手段である可能性を示すことができると考えられる。

一方で、本実験では線形関数近似を行うために状態や行動の抽象化を行っており、これは予備知識を必要とする作業である。また、今回の実験で学習を行ったプレイヤは決定的に出すカードを選択するものであるため、相手の手から手札を推測するようなプレイヤに搾取されることが考えられる。よって確率的に行動を選択するようなプレイヤを作る必要があると考えられる。

以上を踏まえ、NFSP を用いて、予備知識を用いずにトリックテイキングゲームの行動価値関数の強化学習を行うことを今後の課題とする。

参考文献

[1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, et al. Human-level control through deep reinforcement learning. *Nature*, Vol. 518, No. 7540, pp. 529–533, 2015.

[2] Johannes Heinrich and David Silver. Deep reinforcement learning from self-play in imperfect-information games. *arXiv preprint arXiv:1603.01121*, 2016.

[3] Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit hold'em poker is solved. *Science*, Vol. 347, No. 6218, pp. 145–149, 2015.

[4] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, Vol. 8, No. 3-4, pp. 279–292, 1992.

[5] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, Vol. 1. MIT press Cambridge, 1998.

[6] George W Brown. Iterative solution of games by fictitious play. *Activity analysis of production and allocation*, Vol. 13, No. 1, pp. 374–376, 1951.

[7] David S Leslie and Edmund J Collins. Generalised weakened fictitious play. *Games and Economic Behavior*, Vol. 56, No. 2, pp. 285–298, 2006.

[8] Johannes Heinrich, Marc Lanctot, and David Silver. Fictitious self-play in extensive-form games. In *Proceedings of the 32nd International Conference on Machine Learning, ICML*, pp. 805–813, 2015.

[9] Édouard Bonnet, Florian Jamain, and Abdallah Saffidine. On the complexity of trick-taking card games. In *Proceeding of the 23rd International Joint Conference on Artificial Intelligence*, pp. 482–488, 2013.