

相手手札推定を動的に切り替える大貧民プレイヤーの研究

1275108 中山 友里歌 【高度プログラミング研究室】

A Study on DAIHINMIN Players That Dynamically Switch Opponent-Hand-Estimation Methods

1275108 NAKAYAMA, Yurika 【High-Level Programming Lab.】

1 はじめに

ゲーム情報学では、囲碁や将棋、チェスなどの完全情報ゲームにおいて優れたコンピュータプレイヤーが実現されており、近年は不完全情報ゲームに関しても研究が進んでいる。本研究の対象とする「大貧民」では UEC コンピュータ大貧民大会 (UECda) [1] を通して多数のコンピュータプレイヤーが作成されてきた。

UECda の優秀なクライアントの多くは、相手手札情報を何らかの形で補完したうえでモンテカルロ法により手を選択する。著者の先行研究 [2] では、相手手札情報の推定精度や推定するカードの種類がプレイヤーの強さに寄与することを確認している。相手手札の推定手法については複数の手法が提案されているが、一般に推定精度と計算コストはトレードオフの関係がある。

富岡 [3] が開発した UECda の 2020 年無差別級優勝クライアント Ninja では、モンテカルロ法におけるプレイアウト方法を相手に応じて選択バンディット法で動的に切り替えた。本研究は、この相手に応じた選択バンディット法を、より大きな効果が期待される相手手札推定方法に適用するものである [4]。

2 提案手法

本研究で提案するアルゴリズムは、事前に用意した複数の相手手札推定手法を用いて、モンテカルロ法の相手手札推定時にそれらを切り替えるというものである。相手手札推定方法の切り替えは試合ごとに行い、各試合で選択する相手手札推定方法は UCB_V により決定する (Ninja の選択バンディット法と同じ)。本研究で提案する相手手札推定手法の切り替え方法の概念図を、図 1 に示す。

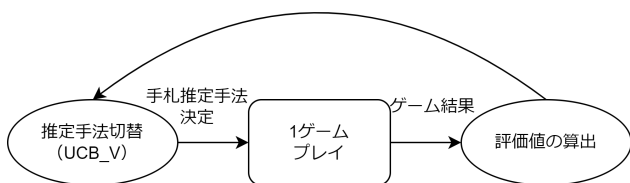


図 1 相手手札推定アルゴリズムの切り替え方法

提案アルゴリズムは以下の手順で動作する。

1. バンディットアルゴリズム UCB_V に基づき、その試合で用いるバンディット手法を決定する。
2. 選択したバンディット手法で 1 試合対戦する。
3. 試合の結果から、バンディットアルゴリズムの評価値を更新する。
4. 指定の試合数 (1)~(3) を繰り返す。

本研究では、大渡 [5] が開発した Blauweregen をベースプレイヤーとして利用し、相手手札推定手法として、Blauweregen の実装に含まれている 4 種類の方法と、全ての相手手札を正しく埋める CHEAT を利用する。

RANDOM 自分の手札以外を完全にランダムに分配する。

SBJINFO 交換等で判明している確実な情報のみ埋め、残りはランダムに分配する。

BIAS 逆関数法を用いて、重みづけに基づいたカード分配を行う。

REJECTION 採択棄却法を用いて複数のカード配置候補を生成し、方策関数で算出した尤度が最大の配置を選択する。

CHEAT サーバから送られた相手手札情報を用いて正確に埋める。

推定手法の強さはおよそ、RANDOM $\approx$ SBJINFO $\approx$ BIAS < REJECTION < CHEAT である。

3 評価実験

Blauweregen をベースに、提案手法を実装したプログラム Kunoichi を用いて評価実験を行う。評価実験では相手手札推定手法が収束するか、推定手法の変更により強くなるかを確認する。対戦相手は既存の UECda ライト級プログラムである switch2 を 4 体、PoliX Lite4 体、それぞれ 2 体ずつの 3 通りとした。

推定手法の強さより、CHEAT あり (cheat_Kunoichi) では CHEAT、CHEAT なし (Kunoichi) では REJECTION に収束することが望ましい。

表 1 評価プレイヤーの獲得得点平均

CHEAT	switch2	PoliXLite	2 体ずつ
cheat_Kunoichi	40792.3	42989.7	41498.7
Kunoichi	36586.5	36260.9	36124.8
Blauweregen	37079.1	36832.2	36480.3

表 2 CHEAT ありにおいて、各相手手札推定手法が選択された回数と、その試合での獲得得点の平均（括弧内）

対戦相手	RANDOM	SBJINFO	BIAS	REJECTION	CHEAT
switch2	201.4 (3.5487)	227.7 (3.6004)	269.8 (3.6820)	253.8 (3.6690)	9047.3 (4.1264)
PoliXLite	167.6 (3.6611)	167.3 (3.6623)	150.7 (3.5839)	152.2 (3.5926)	9362.2 (4.3448)
2 体ずつ	161.1 (3.4786)	160.5 (3.4798)	188.2 (3.5765)	207.5 (3.6410)	9282.7 (4.1961)

表 3 CHEAT なしにおいて、各相手手札推定手法が選択された回数と、その試合での獲得得点の平均（括弧内）

対戦相手	RANDOM	SBJINFO	BIAS	REJECTION	CHEAT
switch2	851.1 (3.5560)	907.7 (3.5699)	874.4 (3.5581)	7366.8 (3.6934)	—
PoliXLite	817.2 (3.4969)	779.2 (3.5069)	1100.9 (3.5278)	7302.7 (3.6681)	—
2 体ずつ	617.7 (3.4586)	1225.8 (3.5152)	1375.6 (3.5398)	6780.9 (3.6588)	—

表 1 に、各実験における評価プレイヤー（cheat_Kunoichi, Kunoichi）の獲得得点の合計を示す。また、最終的な相手手札推定手法の選択回数、および、選択された試合での獲得得点の平均を、表 2 と表 3 に示す。各実験における相手手札推定手法の累積選択割合の推移のうち、2 種類のプレイヤーとの対戦をそれぞれ図 2、図 3 に示す。

Kunoichi の得点は Blauweregen よりは下がっているが、その差は最大で対 PoliXLite の 571.3 点と十分に小さい。

CHEAT ありでは 1000 から 1500 試合、CHEAT なしでも 6000～8000 試合程度で直近の 100 試合において最適な相手手札推定手法を 9 割選択できるようになった。対戦全体では CHEAT ありでは全て 9 割以上、CHEAT なしにおいては 7 割程度の確率で最適な相手手札推定手法が選択できた。

4 まとめ

実験の結果、CHEAT ような極端に有効な相手手札推定手法でなくともある程度最も適したアルゴリズムに収束することが示された。

このことから、相手手札推定手法を切り替える手法は特別有効な推定手法が存在する場合には一定の有効性があると考えられる。

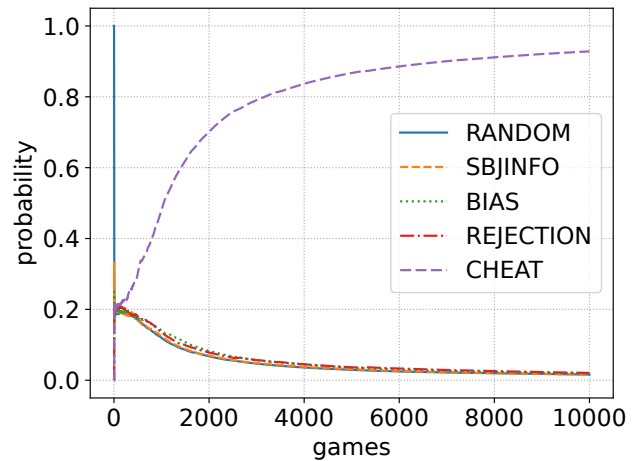


図 2 2 種類のプレイヤー、CHEAT ありにおける推定方法の累積選択割合の推移

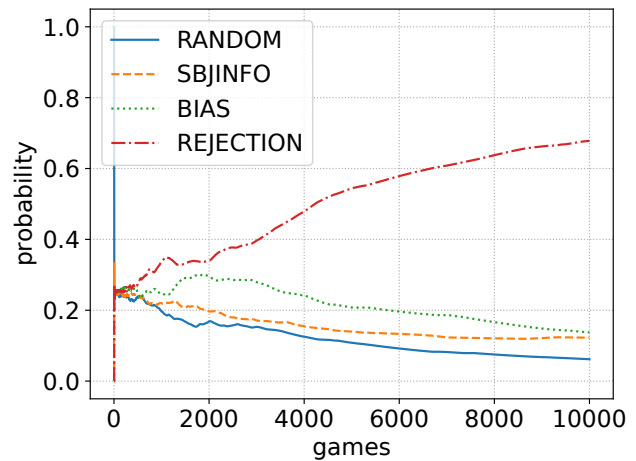


図 3 2 種類のプレイヤー、CHEAT なしにおける推定方法の累積選択割合の推移

参考文献

- [1] UECda-2023: [https://flute.u-shizuoka-ken.ac.jp/daihinmin/2023/\(2023\)](https://flute.u-shizuoka-ken.ac.jp/daihinmin/2023/(2023)).
- [2] 中山友里歌, 植田桂広, 鴨川翔太, 松崎公紀: コンピュータ大貧民における手札推定の有効性についての再評価と考察, 情報処理学会研究報告, 2022-GI-48, No. 6, pp. 1-7 (2022).
- [3] 富岡聖, 大久保誠也, 湯瀬裕昭, 武藤伸明: コンピュータ大貧民における各種モンテカルロ法の検討, 情報処理学会研究報告, 2021-GI-45, No.12, pp. 1-8 (2021).
- [4] 中山友里歌, 松崎公紀: コンピュータ大貧民における相手手札推定を切り替えるプレイヤーの検討, ゲームプログラミングワークショップ 2024 論文集, pp. 173-178 (2024).
- [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).