

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

中山 友里歌^{1,a)} 松崎 公紀^{2,b)}

概要：不完全情報ゲームにおいて、相手プレイヤーの手札など見えない情報の推定は重要であると考えられている。本研究の対象である不完全情報ゲームのひとつである大貧民では、相手手札推定の推定精度がプレイヤーの強さに正の影響があることが示されている。しかしながら、これまでに相手手札推定を行う手法が複数提案されているものの、他のゲームと比較して高い推定精度が達成できていない。本研究では、相手手札推定をより効果的に行うためのアプローチとして、事前に用意した複数の相手手札推定手法を対戦中に切り替えることを提案する。相手手札を完全に知るチートを含めた5つの相手手札推定手法を切り替えるプレイヤーを作成し、2種類のプレイヤーを相手に評価実験を行った。その結果、用意した相手手札推定手法のうち最も適切なものに収束するという結果が得られた。

キーワード：大貧民, 大富豪, 相手手札推定, バンディットアルゴリズム

A Study on Players Switching Opponent-hand-estimation Methods in Computer DAIHINMIN

YURIKA NAKAYAMA^{1,a)} KIMINORI MATSUZAKI^{2,b)}

Abstract: In incomplete information games, estimating hidden information such as the opponents' hand is important in general. In the target game of this study — DAIHINMIN, an incomplete information game —, it has been shown that the accuracy of opponent-hand estimation positively affects the player's strength. Despite several methods being proposed for estimating the opponents' hand, the estimation accuracy remains relatively low compared to other games. In this study, to achieve more effective estimation, we propose an approach to switch between multiple pre-prepared hand-estimation methods during gameplay. We created a player that switches between five hand-estimation methods, including a cheating method that fully knows the opponent's hand, and conducted evaluation experiments against two types of players. As a result, the player converged to the most appropriate estimation method among those available.

Keywords: DAIHINMIN, DAIHUGO, Opponent-hand Estimation, Bandit Algorithms

1. はじめに

ゲーム情報学において、囲碁や将棋、チェスなどの完全情報ゲームに関するAIの研究は盛んに行われ、これらのゲームでは人間よりも明確に優れたコンピュータプレイヤーが実現されている [1], [2]。近年は、ポーカー、麻雀、大貧民などの不完全情報ゲームへの関心も高まっており、数多

くのプレイヤーが開発されている。本研究の対象とする「大貧民」は多人数不完全情報ゲームのひとつであり、これまでに UEC コンピュータ大貧民大会 (UECda) [3] 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モンテカルロ法で実装されている。モンテカルロ法を行うにあたっては、各プレイヤーから見えない相手手札を何らかの方法で埋めてシミュレーションを行うのが一般的である。そこで補完する相手手札情報をより正確に推定することはコンピュータ大貧民プレイヤーの強さに正の影響があると考えられている。実際著者らの先行研究 [8] において、相手手札情報を正し

¹ 高知工科大学大学院工学研究科
Graduate School of Engineering, Kochi University of Technology

² 高知工科大学情報学群
School of Informatics, Kochi University of Technology

a) 275108a@gs.kochi-tech.ac.jp

b) matsuzaki.kiminori@kochi-tech.ac.jp

く推定する割合（推定精度）がプレイヤーの強さに正の影響があることだけでなく、弱い手札を推定できることがより大きな効果があることを確認した。

大貧民における相手手札の推定手法については、機械学習や深層学習を利用する方法を中心に、複数の手法が提案されている [5], [6], [9], [10], [11]。これらの手法は一般に、推定精度を高くするためにはより多くの計算量を必要とするため、特にモンテカルロ法のようにシミュレーション回数も重要なアルゴリズムにおいては、推定精度と計算コストはトレードオフの関係がある。したがって、著者らが知る限りでは、モンテカルロ法と組み合わせた場合に最も適切な相手手札推定手法と言い切れる手法は現段階では存在しない。

本研究では、優れた相手手札推定手法を開発するのではなく、既存の複数の相手手札推定手法をもとに対戦中にそれらを切り替えることを考える。この提案手法は、富岡 [7] が開発した UECda の 2020 年無差別級優勝クライアント Ninja に着想を得た。Ninja は、モンテカルロシミュレーションにおけるプレイアウト方法を選択バンディット法で切り替えることで、相手のプレイスタイルを模倣しようとした。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。具体的には、近年の UECda 優勝プログラムのベースである Blauwereggen をもとに、チートを含む 5 つの相手手札推定手法を切り替えるプレイヤーを作成した。作成したプレイヤーを、既存の UECda ライト級プログラムと対戦させることで、相手に応じてどのように最適な手札推定を発見するか、それによりプレイヤーがどのように強化されるかを評価・検証する。

実験の結果、以下の知見を得た。まず、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。他方、評価値の差が生じにくい対戦条件では、本手法によって最適な相手手札推定手法が有意に多く選択されるまでに時間がかかることもある。

本論文の構成は以下のとおりである。第 2 節では、本研究で用いるコンピュータ大貧民のルールを説明し、また本研究において使用する関連クライアントを紹介する。第 3 節では、相手手札推定の切り替えアルゴリズムについて説明し、本研究で用いる相手手札推定手法についての予備実験の結果も示す。第 4 節では、第一の評価実験として、評価対象プレイヤーが 1 つの場合について、チートの有無によりどの相手手札推定手法がどれだけの割合で選ばれるかについて評価する。第 5 節では、第二の評価実験として、ゲーム開始時の階級の影響を評価値に含めた場合の選択バンディット法の動作について評価する。これらの結果を受けて、第 6 節で本研究をまとめる。

2. コンピュータ大貧民

2.1 コンピュータ大貧民のルール

大貧民はトランプを用いた日本発祥のカードゲームの一つであり、手札を使い切る早さで順位と階級が決定する。数多くのローカルルールが存在するが、本研究では 2006 年から電気通信大学で開催されている UEC コンピュータ大貧民大会 (UECda) [3] の UEC 標準ルールを用いる。

UEC 標準ルールのうち重要なものを以下に示す。

ゲームの流れ ゲームは 5 プレイヤで行われる。手札の配布・交換の後、各プレイヤーが順に手札を提出する。スペード、ハート、ダイヤ、クラブのそれぞれ A~K とジョーカーの計 53 枚のカードを用いる。

階級 手札がなくなった順に順位が決定し、順位に応じて次ゲームでの階級が割り振られる。階級は 1 位のプレイヤーから順に大富豪、富豪、平民、貧民、大貧民であり、順に 5 点、4 点、3 点、2 点、1 点の得点が与えられる。

手札配布 各プレイヤーにランダムに 10 枚か 11 枚の手札が配布される。

手札交換 手札配布後、大富豪と大貧民は 2 枚、富豪と貧民は 1 枚カードを交換する。大富豪と富豪は任意のカードを、大貧民と貧民は最も強いカードを渡す。

カード役 カード 1 枚のみを使用する単体役、同ランクのカードを複数使用する複数役、同マークで連続したランクのカードを 3 枚以上使用する階段役の 3 種類がある。

手札提出 ダイヤの 3 を持つプレイヤーからゲームを開始する。手番が回ってきたプレイヤーは、カードの提出かパスを行う。場に何もない状態であれば任意のカードを提出できる。場にカードが出ている場合は、場のカードと同じ役、同じ枚数で、より強い手のみ提出できる。

8 切り 8 を含んだ役を出した場合、場が流れる（場にカードがない状態となる）。

スペードの 3 ジョーカーの単体役が提出されている場合、スペードの 3 を提出できる。その場合は場が流れる。

ジョーカー 単体役もしくは複数役において全てのカードの代用にすることができる。単体役として出した場合は、2 より強い単体役となる。

革命 4 枚以上の複数役もしくは 5 枚以上の階段役が出された場合、ジョーカー以外のカードの強さが逆転する。革命中に革命が起きた場合はカードの強さが元に戻る。

しぼり 場と同じマークが出された場合、場が流れるまで同じマークのカードしか出すことができない。

席順変更 3 試合に 1 回、各ゲーム開始時にプレイヤーの席順がランダムに決定される。

2.2 コンピュータ大貧民の既存クライアント

2.2.1 Blauwereggen

大渡が開発した Blauwereggen [4] は 2017 年無差別級優勝クライアントである。Blauwereggen では、方策勾配法により調整した方策関数を使用してモンテカルロシミュレーションを行う。その際、採択棄却法 (REJECTION) により相手手札推定を行っている。本研究では、Blauwereggen の実装に含まれている 4 つの相手手札推定に、チートを加えた 5 つを用いる。

2.2.2 Ninja

富岡が開発した Ninja [7] は 2020 年無差別級優勝クライアントである。Ninja は、Blauwereggen をベースとし、そのモンテカルロ法部分を改良している。具体的には、モンテカルロ法で用いるバンディットアルゴリズムとして UCB-root, UCB1-Tuned, UCB2, KLUCB, UCB-V の 5 つを事前に用意し、それらのうち採用するアルゴリズムを選択バンディット法により決める。

2.2.3 switch2

switch2 は 2022 年 UECda ライト級優勝クライアントである。switch2 では複数の戦略を実装しており、最初の一定数の試合ではそれらを均等に使用するが、その後は勝率の最も高い戦略のみを使用する。

2.2.4 PoliXLite

PoliXLite は 2020 年 UECda ライト級優勝クライアントである。PoliXLite も Blauwereggen をベースとしているが、モンテカルロシミュレーションを行わずに方策関数の結果をそのまま用いて手を選択するという違いがある。

2.3 相手手札推定手法

相手手札推定手法として、今回は Blauwereggen の実装に含まれている以下の 4 種類の手法と、すべての相手手札を正しく埋めるチート (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、手札親和度が最大の配置を選択する。Blauwereggen で実際に利用された方法である。

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。

実験に用いる各種手札推定手法の強さを確認する予備実験を行った。実験には推定手法を固定した改造 Blauwereggen を用い、以下の組み合わせで 10000 ゲームからなる対戦を 10 回行った。

予備実験 1 switch2 × 4, 改造 Blauwereggen × 1

予備実験 2 PoliXLite × 4, 改造 Blauwereggen × 1

表 1 各推定手法を用いた改造 Blauwereggen の獲得得点

対戦相手	switch2	PoliXLite
	平均 ±標準偏差	平均 ±標準偏差
RANDOM	35670.8± 1689.8	34987.5± 2974.6
SBJINFO	35654.9± 1739.9	35083.2± 2873.9
BIAS	35904.4± 1760.5	35633.3± 2788.4
REJECTION	37098.3± 1540.2	36741.6± 3042.7
CHEAT	39444.4± 1487.4	41881.7± 2942.2

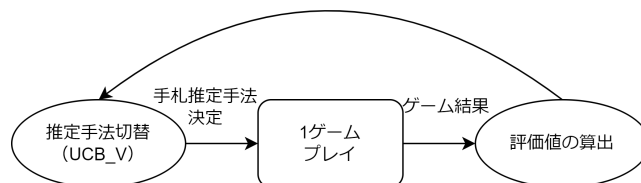


図 1 相手手札推定アルゴリズムの切り替え方法

各推定手法を用いた改造 Blauwereggen の獲得得点を表 1 に示す。相手手札推定手法を固定したプレイヤーはおよそ、RANDOM ≈ SBJINFO ≈ BIAS < REJECTION < CHEAT の順に並んでいる。REJECTION は、Blauwereggen に実装された他 3 つの推定方法と 1000 点以上離れている。また、CHEAT は対 switch2 において 2400 点、対 PoliXLite において 5000 点と、大きく REJECTION を上回っている。

3. 提案手法：選択バンディット法による相手手札推定の切り替え

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。

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 ゲーム対戦する。
- (3) ゲームの結果から、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。

4. 実験 1: 評価対象プレイヤーが 1 つの場合の評価

Blauwereggen をベースに、相手手札推定手法を動的に切り替える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。具体的には、CHEAT を含む場合には CHEAT が、CHEAT を含まない場合には REJECTION が最も高頻度選ばれる

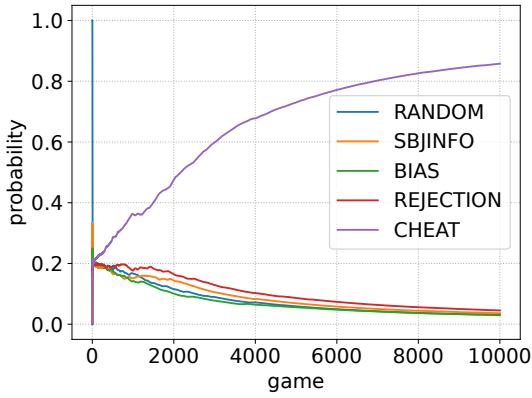


図2 実験 1-S-C (switch2, CHEAT あり) における推定方法選択割合の推移

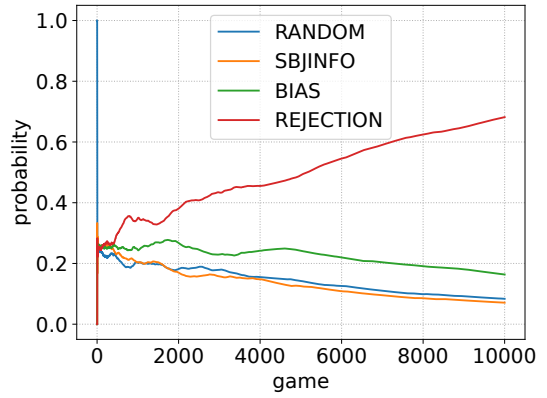


図3 実験 1-S-K (switch2, CHEAT なし) における推定方法選択割合の推移

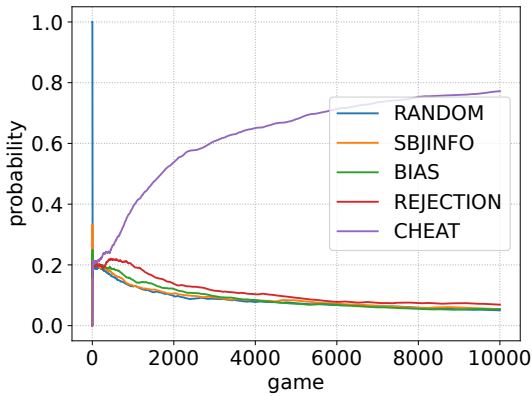


図4 実験 1-P-C (PoliXLite, CHEAT あり) における推定方法選択割合の推移

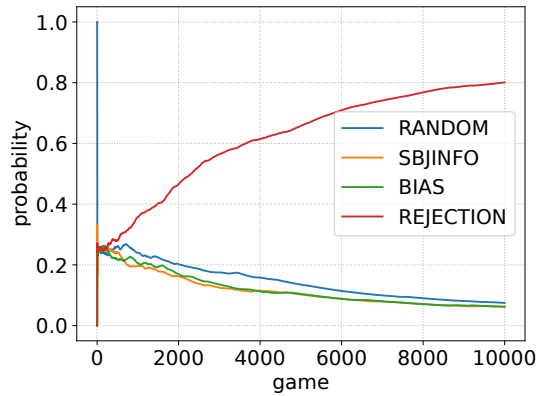


図5 実験 1-P-K (PoliXLite, CHEAT なし) における推定方法選択割合の推移

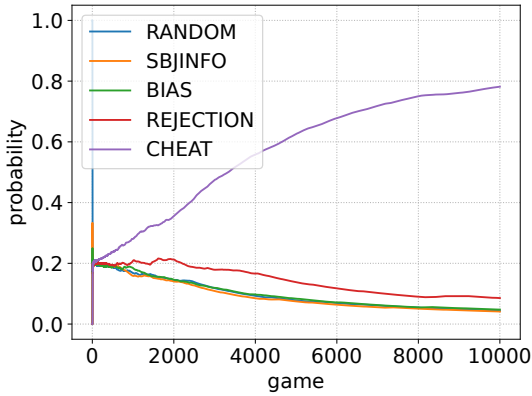


図6 実験 1-M-C (2種類のプレイヤー, CHEAT あり) における推定方法選択割合の推移

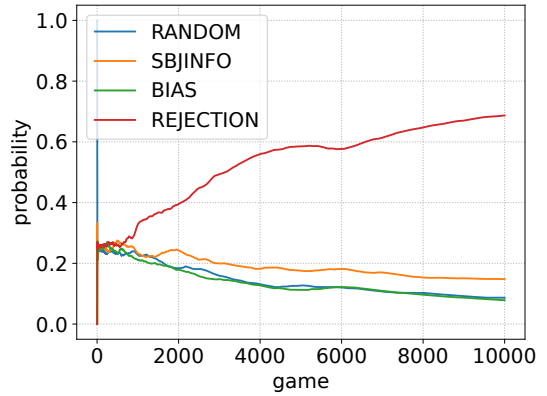


図7 実験 1-M-K (2種類のプレイヤー, CHEAT なし) における推定方法選択割合の推移

ことが望む結果である。

実験には、提案手法によるクライアントとして、相手手札推定手法に CHEAT を含まない Kunoichi, 相手手札推定手法 CHEAT を含む cheat_Kunoichi の 2 つを用いる。それらの対戦相手には、switch2 と PoliXLite を用いる。以下の組み合わせで 10000 ゲームからなる対戦を 10 回行った。

実験 1-S-C switch2 × 4, cheat_Kunoichi

実験 1-P-C PoliXLite × 4, cheat_Kunoichi

実験 1-M-C switch2 × 2, PoliXLite × 2, cheat_Kunoichi

実験 1-S-K switch2 × 4, Kunoichi

実験 1-P-K PoliXLite × 4, Kunoichi

実験 1-M-K switch2 × 2, PoliXLite × 2, Kunoichi

各実験における相手手札推定方法の選択割合の推移を、図 2~7 にそれぞれ示す。表 2 に、各実験における評価プ

表2 実験1における評価プレイヤーの獲得得点

CHEAT	switch2	PoliXLite	2体ずつ
cheat_Kunoichi	36509.3	42513.3	41108.0
Kunoichi	36622.8	38974.7	39192.4

表3 実験1-S-C, 実験1-P-C, 実験1-M-C (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	312.2 (3.571)	360.4 (3.621)	296.3 (3.560)	453.0 (3.696)	8578.1 (3.943)
PoliXLite	503.2 (4.254)	542.3 (4.254)	544.4 (4.262)	689.3 (4.291)	7720.8 (4.247)
2体ずつ	437.1 (3.910)	412.7 (3.895)	478.3 (3.935)	855.6 (4.031)	7816.3 (4.153)

表4 実験1-S-K, 実験1-P-K, 実験1-M-K (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	836.4 (3.5508)	708.7 (3.5277)	1636.5 (3.5942)	6818.4 (3.7063)	—
PoliXLite	1276.0 (3.5346)	1518.2 (3.5047)	1637.9 (3.5085)	5567.9 (3.6842)	—
2体ずつ	865.3 (3.8707)	1479.0 (3.9199)	785.9 (3.8054)	6869.8 (3.9382)	—

レイヤ (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方法の選択回数、および、選択された試合での獲得得点の平均を、表3と表4に示す。

CHEATありの対戦では、全試合を通じて80%前後の確率でCHEATが選ばれ、チートなしの対戦では60%前後の確率でREJECTIONが選ばれることとなった。また、全体として、PoliXLiteとの対戦は相手手札推定の精度が低い。これはPoliXLiteとの対戦での勝率が平均的に高いため、最適な相手手札推定手法以外でも評価値が高くなり、特定の手法の有用性が見えないためだと考える。

5. 実験2: 試合開始時の階級を考慮した評価値を用いる効果

この実験では、手札推定手法の切り替えに用いる評価値について調査する。Ninjaの手札推定手法の切り替えに用いる評価値は、得点のみから算出されるものであった。大貧民は前の試合の勝者(大富豪, 富豪)が勝ちやすく、敗者(大貧民, 貧民)が負けやすいゲームである。このことを考慮して、以下のように評価値の与え方を変更することを考える(階級考慮あり, WR)。

- 前の試合の結果が大貧民または貧民であるとき、評価値を獲得点数+1とする。
- 前の試合の結果が平民であるとき、評価値を獲得点数とする。

表5 実験2における評価プレイヤーの獲得得点合計

	Kunoichi × 4	Kunoichi × 3
評価値: 得点のみ	31681.7	33015.5
評価値: 階級考慮	31760.5	32764.0

- 前の試合の結果が大富豪または富豪であるとき、評価値を獲得点数-1とする。

実験にはKunoichiとUECdaの2022年ライト級優勝クライアントであるswitch2を用い、以下の組み合わせで10000ゲームからなる対戦を10回行った。

実験2-PT-K4 評価値は得点のみ, switch2, Kunoichi×4

実験2-PT-K3 評価値は得点のみ, switch2×2, Kunoichi×3

実験2-WR-K4 評価値に階級考慮, switch2, Kunoichi×4

実験2-WR-K3 評価値に階級考慮, switch2×2, Kunoichi×3

各相手手札推定方法の選択割合の推移を図8~11に示す。また、評価プレイヤー(プレイヤー4とする)の獲得得点の合計を表5に示す。

本実験の結果では、評価値の設定方法で結果に大きな差が開いている。特に、階級を考慮した評価値算出方法を用いる実験2-WR-3において、得点のみで評価値を算出した場合の0.63倍しかREJECTIONが採用されていない。この結果から、対戦相手に同レベルのプレイヤーが多い場合には得点だけを考慮する方がよい可能性が示唆される。

6. まとめ

本研究の目的は、相手手札推定方法を動的に切り替える手法の提案ならびにその有効性の検証である。実験の結果、チートのような極端に有効な相手手札推定手法があれば4000試合程度で推定方法が定まること、チートなしの場合でも6000試合程度で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。

PoliXLiteとの対戦や同じクライアント同士の対戦では最適な相手手札推定手法の選択割合が低い傾向にあったが、これはどの相手手札推定手法でも勝率が変わりにくい条件であったためだと考えられる。

チートありの対戦では必ずチートが頻繁に選ばれていたため、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。

また、前試合の階級を考慮した評価値を用いた実験では、得意な相手には僅かにしか強くならず、そうでない相手に対しては最適でない相手手札推定手法が多く選ばれる結果となった。大貧民は前の試合の結果が大きく影響するゲームではあるが、最初に配られる手札の強弱も大きく影響するため、ゲーム開始時の手札の強弱を反映した評価値算出等により精度を上げられる可能性が考えられる。

今回の研究ではすべてのプレイヤーに対して同じ推定手法を利用した。実際の対戦では、対戦相手それぞれに別の推

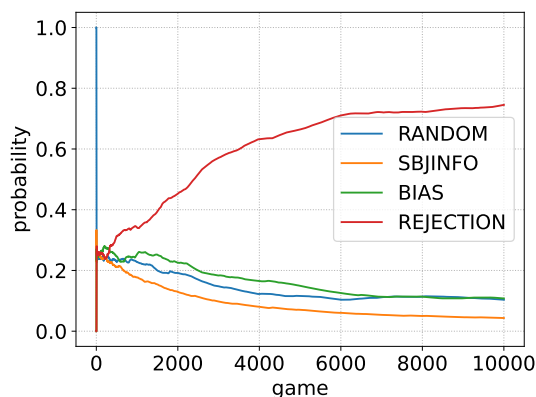


図8 実験 2-PT-K4 (得点のみ, Kunoichi × 4) における各推定方法の選択割合の推移

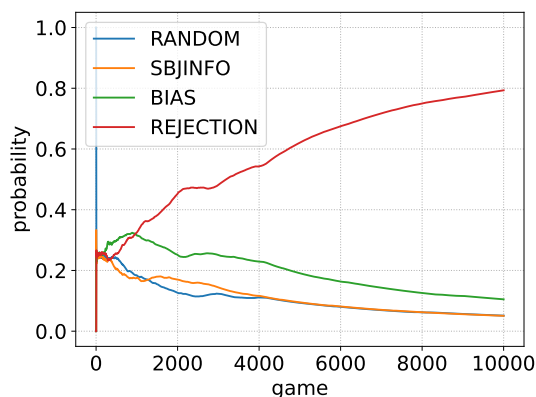


図9 実験 2-WR-K4 (階級考慮, Kunoichi × 4) における各推定方法の選択割合の推移

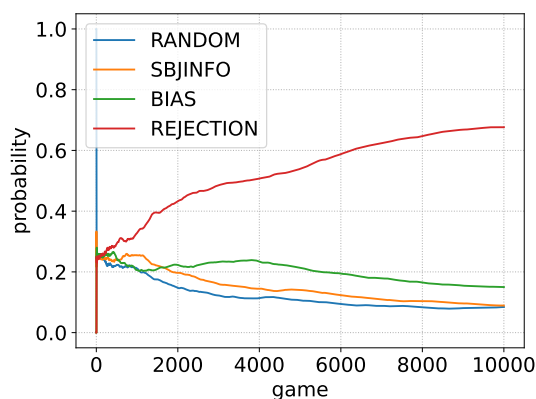


図10 実験 2-PT-K3 (得点のみ, Kunoichi × 3) における各推定方法の選択割合の推移

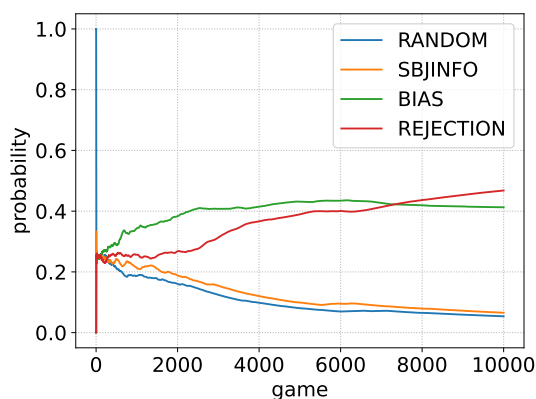


図11 実験 2-WR-K3 (階級考慮, Kunoichi × 3) における各推定方法の選択割合の推移

定方法を選べるようにする必要がある。また、通常は有効ではないものの、特定の戦法に特化した相手手札推定手法を導入することが、より強力なプレイヤーの作成に有効である可能性がある。

本研究では、CHEAT もしくは REJECTION がすべての対戦相手に対して有効な相手手札推定手法であった。相手によって最適な手札推定手法が異なる場合の実験を行い、本手法の有効性をより詳細に検証することは重要な今後の課題である。

参考文献

- [1] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel and D. Hassabis: Mastering the game of Go without human knowledge, *Nature*, **550**:354–359, 2017.
- [2] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan and D. Hassabis: A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play, *Science*, **362**(6419):1140–1144, 2018.

- [3] UECda-2023: <https://flute.u-shizuoka-ken.ac.jp/daihinmin/2023/> (2023).
- [4] 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, 2016-GI-35, No. 10, pp. 1–8 (2016).
- [5] 神田直樹, 伊藤毅志: コンピュータ大貧民における LSTM を用いた手札推定, 情報処理学会研究報告ゲーム情報学 (GI), 2018-GI-39(8), pp. 1–8, 2018.
- [6] 須藤郁弥, 成澤和志, 篠原歩: UEC コンピュータ大貧民大会向けクライアント「snow」の開発, 第 2 回 UEC コンピュータ大貧民シンポジウム, 2010.
- [7] 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, 2021-GI-45, No.12, pp.1-8 (2021)
- [8] 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, 2022-GI-48, No. 6, pp. 1–7 (2022).
- [9] 西野順二, 西野哲朗: 大貧民における相手手札推定, 情報処理学会研究報告数理モデル化と問題解決 (MPS), 2011-MPS-85(9), pp. 1–6, 2011.
- [10] 濱田恭輔, 鈴木海友, 松澤智史: 大貧民エージェントの強化に向けた手札推定, 2023-GI-49, No. 5, pp. 1–7 (2023).
- [11] 柳澤佑介, 松崎公紀: 大貧民における出現頻度と提出手役履歴を用いた相手手札推定, 情報処理学会研究報告ゲーム情報学 (GI), 2015-GI-33(9), pp. 1–6, 2015.