

不完全情報ゲーム「ガイスター」における 相手駒色推定の有効性評価

竹内 聖悟^{1,a)} 栃川 純平¹ 松崎 公紀^{1,b)}

受付日 2021年5月31日, 採録日 2021年12月3日

概要: 不完全情報ゲームにおいては、一般に未知の情報を推定することで有利にプレイできると考えられている。ガイスターは、相手の駒の色が分からない二人不完全情報ゲームである。その重要な特徴は、駒の色のみが未知であり、その不完全情報によってとりうる着手が変化しないことである。これまでガイスターのコンピュータプレイヤーが研究・開発されており、相手駒色推定（未知の情報の推定）を行うものもある。しかし、その推定の有効性については十分に調査されているとはいえない。本研究では、本来知りえない情報を知っているプレイヤー（Cheating）を対戦させたときの勝率から、駒色推定の有効性を評価する。具体的には、初期局面におけるすべての未知の情報を得るものに加えて、途中局面や部分的にしか情報が得られない場合についても調査する。さらにゲーム特有の知識や推定のための知見を得るため、推定する駒の数や色、配置、推定精度に着目して調査する。実験の結果、推定する駒数が多いほど、また、推定精度が高いほど勝率が良くなり、ガイスターにおいて相手駒色推定が有効であることが確認された。また、重視すべき特徴として青駒よりも赤駒を推定する方が有効であることなどの知見が得られた。

キーワード: 不完全情報ゲーム, ゲーム木探索, 推定

Impact of Imperfect Information Estimation in the Game of Geister

SHOGO TAKEUCHI^{1,a)} JUNPEI TOCHIKAWA¹ KIMINORI MATSUZAKI^{1,b)}

Received: May 31, 2021, Accepted: December 3, 2021

Abstract: It is generally believed that estimating the unknown information gives an advantage in imperfect information games. Geister is a two-player imperfect information game in which the colors of the opponent's pieces are unknown. An important feature of Geister is that only the colors of the pieces are unknown, and the legal moves are not changed by the imperfect information. Computer players have been developed for Geister, and some of them estimate the color of the opponent's pieces (i.e., estimate the unknown information). However, the effectiveness of estimation has not been sufficiently investigated. In this study, we evaluate the effectiveness of piece color estimation based on the winning rate of a (cheating) player who knows information that is not originally known. Specifically, in addition to obtaining all unknown information in the starting position, we also investigate cases in which information can be obtained only in the middle phase or partially. In addition, we focus on the number of pieces to be estimated, their colors, placement, and estimation accuracy in order to obtain game-specific knowledge and insights for estimation. As a result of the experiment, it was confirmed that the more the number of pieces to be estimated and the higher the estimation accuracy, the better the winning rate, and the more effective the opponent piece color estimation is in Geister. In addition, we found that the estimation of red pieces is more effective than that of blue pieces as a feature to be emphasized.

Keywords: imperfect information game, game tree search, estimation

¹ 高知工科大学大学院工学研究科
Graduate School of Engineering, Kochi University of Technology, Kami, Kochi 782-8502, Japan

a) takeuchi.shogo@kochi-tech.ac.jp

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

1. はじめに

二人零和有限確定完全情報ゲームである将棋や囲碁、チェスでは、強いコンピュータプレイヤーを作る研究開発が進め

られ、人間よりも強いコンピュータプレイヤーが作成されるようになった [1]。また、不完全情報ゲームにおいてもバックギャモン、ポーカーでトッププレイヤーに相当する実力を持つ AI が研究されている [2]。

ガイスターは、相手の駒の色が分からない二人零和有限確定不完全情報ゲームである。このゲームは、6×6の盤面で各プレイヤーが青と赤の駒を4個ずつ使用して勝利を目指すゲームである。ガイスターは、国内でコンピュータプレイヤーの大会*1が開催されプレイヤーの研究や開発がされているほか、詰めガイスター問題 [3] の研究が行われるなど、新たな研究対象として注目を集めている。

二人零和有限確定不完全情報ゲームには、衝立将棋 [4] や Phantom Go, Kriegspiel [5] など、将棋や囲碁、チェスを元にした不完全情報ゲームが存在する。Kriegspiel, 衝立将棋では、初期配置は分かっているが、各プレイヤーの手(どの駒をどこへ動かしたか)が不明となる。駒を取ったとき、取られたとき、王手をかけられたときなどに情報を得ることができるもの、相手番において可能な手が不明であるため探索を行うことが難しい。Phantom Go は囲碁を元にしたゲームであり、何も置かれていない初期配置から互いに石を配置していくが、相手がどこに配置したかは不明となっている。石を取った際やアタリ(石を取れる状態になった)ことなどに情報が得られる。ガイスターは、駒を動かすゲームという点で衝立将棋や Kriegspiel と類似している。しかし、相手駒の色だけが不明であるものの、互いの駒の場所は分かっており、未知の情報が合法手に影響しないことは重要な特徴である。

不完全情報ゲームでも探索により指手選択を行うが、不完全情報を適切に扱う必要がある。単純ながら手法としては、可能な状態集合からランダムに状態を仮定し、モンテカルロ木探索(以下、MCTS)を行う [6] ものがあげられる。

Long らは、不完全情報ゲームのうちゲームが進行するに従って情報が増えるゲームにおいて、意思決定に未知の状態の推定を用いることが有効とした [7]。一方で、同様にゲームが進行するに従って情報が増える大貧民においては、正確に推定することが必ずしも有効ではないという結果が得られている [8]。未知の情報を推定することの有効性を評価する手法の1つに、Cowling ら [9] による、本来知りえない情報を知ったプレイヤー(Cheating と呼ぶ)を用いるものがある。その手法では、通常のプレイヤーと Cheating プレイヤーとで対戦を行い、勝率に差があれば正確な推定が有効であるとす。

ガイスターにおいても推定は重要と考えられるが、推定が本当に有効であるかや有効となる条件などについての研究は著者らの知る限りない。有効となる条件や重視すべき特徴などについて知識を得ることができれば、ガイスター

プレイヤー開発に役立てることができる。

まずは、初期局面を対象として、Cheating との対戦から推定の有効性を評価する。さらに、部分的な情報であっても有効であるかを調べるため、すべての情報を得るのではなく推定する駒の数や色を変えて実験するほか、推定精度を変えての実験や途中局面からの実験なども行い、より現実的な局面での推定の有効性の評価を行う。

以下本稿では、この研究で扱うガイスターについて説明し、不完全情報ゲームについての関連研究について述べる。続いて、提案手法について説明し、実験とその結果について述べる。その後、結果に基づいて議論を行い、内容のまとめを行う。

2. ガイスター

ガイスターは、互いのプレイヤーが青駒と赤駒を各4個ずつ計8個の駒を用いて行う二人不完全情報ゲームである。このゲームは、各プレイヤーは自身の駒の色は確認することができるが相手の駒の色は確認できないという特性を持っている。ゲームは、6×6の盤面で行い、以下の勝利条件を目指す。

- 自分の青駒を脱出させる。
- 相手の青駒をすべて取る。
- 自分の赤駒をすべて取らせる。

脱出は自陣の反対側にある脱出口(図1の黒いマス)である。ゲーム開始時に各プレイヤーは決められた場所に8個の駒を自由に配置し、交互に駒を動かしていく。自身のターンでは、自身の駒1つを上下左右に1マス動かすことができる。移動先に相手の駒がある場合はその駒を取ることができ、取った駒の色を確認することができる。ゲームを進めていきどちらかのプレイヤーが勝利条件のいずれかを満たしたらゲーム終了である。

駒移動型の不完全情報ゲーム Kriegspiel や衝立将棋などとガイスターとの違いとして、ガイスターで不明な情報は駒の色だけであり、可能な指手が色(情報)によって変わらないという点がある。これは、色が違っていても駒の移動可能なマスは変わらないためである。行動の違いとして

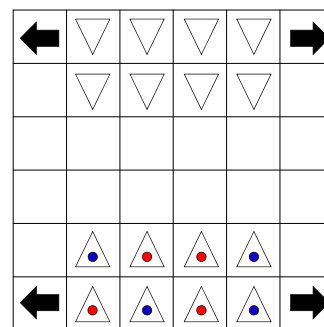


図1 ガイスターの盤面

Fig. 1 Starting board of Geister.

*1 <http://www.matsue-ct.jp/home/hashimoto/geister/GAT/>, 最終アクセス日 2021年5月28日

青駒は脱出でき赤駒は脱出できないという違いがあるが、青駒の脱出は勝利条件でありゲームが終わるため、実質的には違いがない。

3. 関連研究

不完全情報ゲームにおける探索手法や不完全情報の推定、ガイスターにおける関連研究などを紹介する。

3.1 モンテカルロ木探索

モンテカルロ木探索は囲碁において提案された探索手法で、その後ゲームに限らない広い分野で成功をおさめている [10]。モンテカルロ木探索は、木探索とプレイアウト部分に分かれており、木探索部分で Upper Confidence Bound (UCB) によってこれから評価すべき局面を得て、プレイアウト部分ではシミュレーションによる評価を得る。アルゴリズムをアルゴリズム 1 に記載する。なお、ノード n に対応する局面を $s(n)$ 、訪問回数を $N(n)$ 、報酬の総和を

Algorithm 1 モンテカルロ木探索 (Monte-Carlo Tree Search: MCTS)

```

function MCTS( $s_0$ )
  局面  $s_0$  に対応するルートノード  $n_0$  の生成
  while 訪問回数が閾値未満 do
     $n := \text{TREEPOLICY}(n_0)$ 
     $v := \text{PLAYOUTPOLICY}(s(n))$ 
     $\text{BACKUP}(n, v)$ ;
  return  $n_0$  の子ノードで訪問数最大のノードに相当する指し手

function TREEPOLICY( $n$ )
  while  $n$  が非終端ノード do
    if  $n$  が未展開 then
      return  $\text{EXPAND}(n)$ 
    else
       $n$  を、子ノードの中で UCB1 値最大のノードに更新
  return  $n$ 

function EXPAND( $n$ )
  局面  $s(n)$  の合法手の中で未試行の手  $m$  を選ぶ
   $n$  に、局面  $s(n)$  から指し手  $m$  で遷移した局面に対応するノード  $n'$  を追加する
  return  $n'$ 

function PLAYOUTPOLICY( $n$ )
  局面  $p := s(n)$ 
  while 局面  $p$  が非終端局面 do
    局面  $p$  での合法手から指し手  $m$  をランダムに選ぶ
    局面  $p$  を、 $p$  から  $m$  で遷移した局面として更新
  return 局面  $p$  の報酬

function BACKUP( $n, v$ )
  while  $n$  が null でない do
     $N(n) := N(n) + 1$ 
     $W(n) := W(n) + v$ 
     $v := -v$ 
     $n := n$  の親ノード

```

$W(n)$ とする。また、式 (1) によって、ノード n の子ノード n_i の UCB 値を計算する。アルゴリズムの TreePolicy における UCB 値はこの式に従って計算される。

$$UCB \text{ 値} = \frac{W(n_i)}{N(n_i)} + c \sqrt{\frac{\ln N(n)}{N(n_i)}} \quad (1)$$

3.2 Information Set UCT

不完全情報ゲームにも不確実性や不完全情報によって違いがあるため、それぞれに対応するモンテカルロ木探索 (Information Set MCTS) を提案した研究がある [9]。比較対象として、本来知ることができない不完全情報を得て完全情報として探索を行う Cheating UCT や不完全情報について可能なパターンをいくつか選び、それぞれのパターンについて完全情報の MCTS を独立に行う Determinized UCT を用いている。Determinized UCT は複数の木ができるため、戦略の融合などの問題がある。Cowling らは、それを解決する手法として、作る木は 1 つだが情報集合によって分岐を行う Single Observation Information Set Monte-Carlo Tree Search (SO-ISMCTS) などの手法を提案した。また、相手の指手が部分的にしか観測できない場合、同時に着手する場合なども対象とした探索手法も提案した [9]。

また、不完全情報の推定が有効でない局面の存在について、Determinized UCT と Cheating UCT の勝率の差を調べることで評価を行っている。勝率に差がない場合は、不完全情報の推定があっても勝率が変わらないため、推定の効果がないと考えられる。

3.3 不完全情報ゲームにおける推定

不完全情報ゲームは、ゲームの状態に不明な部分が存在しているためモンテカルロ木探索などの探索手法の適応が難しい。そのため、文献 [6] では可能な状態集合からランダムに状態を仮定し、モンテカルロ木探索を行っている。

Long らは、不完全情報ゲームのうちゲームが進行するに従って情報が増えるようなゲームにおいては、着手決定において未知の状態の推定を用いることが有効としている [7]。一方で、西野らは同じ特徴を持つ大貧民においては、推定が必ずしも有効でない例があることを実験的に確認した [8], [11]。

3.4 ガイスタープレイヤにおける推定

ガイスターにおいては推定が有効と考えられており、いくつか推定手法が提案されている。末續らは、取られそうな駒を逃がすか逃さないかで青らしいか赤らしいかを増すなど、ルールベースの推定手法を提案した [12]。また、相手の駒が脱出口にいて脱出をしない場合はその駒を赤と見なすなど、必勝手順との関係から駒を推定する手法も提案した [13]。ほかにも三塩らはモンテカルロ法のプレイヤに

において、2手前の着手決定に使用したプレイアウト結果から相手駒の色を推定する方法を提案し、対戦からその有効性を示した [14]。また、寺村らはニューラルネットワークによる駒色推定を提案し、モンテカルロ法プレイヤを相手として学習し、対戦実験から有効性を示した [15]。

4. 提案手法

4.1 手法の概要

ガイスターにおいて、相手の駒色推定が有効であるか、有効だとしてどの程度の精度が必要とされるか、1つの駒でも有効となるかなど、推定の有効性や有効となる特徴などを調査する。そのために、通常のゲームプレイから得られる情報だけを使う MCTS プレイヤと、本来知りえない相手駒色の情報を得る Cheating プレイヤとの対戦を行い、Cheating プレイヤの勝率から評価を行う。

4.2 手法

駒色推定の有効性を評価するため、通常のゲームプレイから得られる情報だけを使う MCTS プレイヤと、本来知りえない相手駒色の情報を得る Cheating プレイヤとで対戦を行う。推定するプレイヤに相当する Cheating プレイヤが有意に勝ち越せば、推定が有効であると評価できる。次のような対戦実験を行う。

- (1) 推定する駒数・駒色を変えた、推定精度 100%での、初期局面からの対戦
- (2) 推定する駒数・駒色を固定し、推定精度を変えた、初期局面からの対戦
- (3) 推定する駒数・駒色を固定し、推定精度を変えた、途中局面からの対戦

(1) は、初期局面での推定の有効性の調査が目的となる。さらに、推定する駒の数や色が有効性にどれだけ影響するかを調査も行うことができる。このとき、「推定する駒の位置」についても調査し、推定する数や色、位置の違いが有効性に影響するかを調査する。次の (2) は、推定の有効性に対して、推定精度がどれだけ影響するかを調査となる。最後の (3) は、より現実的な評価を行うため、途中局面からの対戦を行い、推定の有効性を評価する。情報を得る駒については、動きの多い駒ほど推定しやすいと考え、今回は単純のため相手プレイヤの駒で自陣に近い駒を対象とする。また、この実験では「推定精度」を変えながらの対戦を行い、推定精度の影響を調査する。(1) から (3) のすべての実験で、Cheating プレイヤの勝率とそのときの勝敗条件について調査を行う。表 1 に実験条件と調査項目をまとめる。

4.3 期待される効果

まず、すべての実験で調査する勝率と「勝敗条件の変化」の結果から期待される効果について述べる。Cheating

表 1 実験条件と調査項目について

Table 1 Experimental conditions and investigations.

	実験条件			調査項目			
	数・色	精度	開始局面	勝率	勝敗条件	色	位置
(1)	変動	100%	初期局面	✓	✓	✓	✓
(2)	固定	変動	初期局面	✓	✓		
(3)	固定	変動	途中局面	✓	✓		

UCT が有意に勝ち越すのであれば、推定によって有意に強くなったと考えられるため、推定の有効性の評価が可能となる。勝敗条件の変化の調査では、実験条件の違いによって勝利の仕方や敗北の仕方が変わることを見出されれば、メカニズム解析や新しい知見の発見につながると期待できる。

次に (1) の「推定する駒の数」「推定する駒の色」を変えた実験から期待される効果について述べる。初期局面における推定の有効性を調べられるほか、「推定する駒の数」を変える実験から、部分的な情報の取得であっても有効であるかを調べることができる。これにより、1つの駒でも情報を得ることが重要であるか、それとも複数の駒の情報がなくとも効果がないかを調べることができる。また、「推定する駒の色」を変えた実験結果から、どちらの情報を得た方が勝ちやすいなどの知見が得られることが期待される。推定する駒が赤駒のときは赤駒を 4 個取って敗北する割合が減少する、青駒が 4 個分かっていると青駒を 4 個取って勝利する割合が増加するという結果が期待される。さらに、実験結果で「推定する駒の位置」について調査することで期待される効果について述べる。推定する「駒の位置」によって有効性が異なる場合、初期配置決定の良し悪しに対する知見が得られると期待される。なお、初期配置については対称性を考えると 4 カ所に分類され、その場所によって違いがあるかを調べる。

続いて (2) の「推定精度」を変えた実験から期待される効果について述べる。推定には精度とコストのトレードオフがあり、「推定精度」を変えた実験から推定精度ごとの有効性などが分かれば、どこまでコストをかけて精度を高めるかなどの指針になりうる。

最後に、(3) の対戦開始局面をゲームの途中局面に変更することで期待される効果を述べる。これまでは初期局面から Cheating によって本来得られない情報を得ており、実際の対戦へ適応できない可能性がある。より現実的な設定としてゲームの途中局面から開始すること、情報を得る駒について、より多く動いたために推定しやすいと考えられる自陣に近い駒を対象とするなどした。この対戦結果においても Cheating プレイヤが有意に勝ち越すのであれば、推定の有効性が評価できる。さらに、これまでの実験結果と比較して、大きく異ならないとすれば、それまでに得られた知見もまた実戦に有効と考えられる。

5. 実験

本実験では、推定の有効性を調査するために相手の駒色を推定するプレイヤー (Cheating プレイヤ) と通常のプレイヤーとの対戦を行う。以下の、3種類の対戦実験を行った。

- (1) 推定する駒数・駒色を変えた、推定精度 100%での、初期盤面からの対戦 (推定する駒数と駒色の影響)
- (2) 推定する駒数・駒色を固定し、推定精度を変えた、初期盤面からの対戦 (推定精度の影響)
- (3) 推定する駒数・駒色を固定し、推定精度を変えた、途中局面からの対戦 (対戦開始局面の影響)

まず、(1)の対戦では、推定駒数は1から7で、駒数ごとに可能な青・赤のパターンをすべて試す。たとえば、推定駒数4であれば、青1赤3、青2赤2、青3赤1の全パターンで対戦を行う。これにより、推定の有効性や有効となる駒数、推定した駒色の違いが勝率に影響するかが調査できる。さらに、推定する場所の影響や勝敗条件からのに着目し、初期配置の良し悪しへの影響を調査する。

次に、(2)の対戦では、「推定精度の影響」に着目し、情報を取得する駒数を青2赤2個と固定し、推定の精度を60%から90%を10%ごとに変化させて対戦実験を行う。対戦結果から、精度と有効性の関係を調査するほか、どのような勝利・敗北条件を満たしたかも調べ、精度の影響が生じているかを調査する。

最後に、(3)の対戦では途中局面からの対局を行う。過去の対戦などから20手目の局面で、互いに駒が同数残っている局面を対象とした。その中から、青4赤4、青3赤3、青2赤2となっている局面を対象として対戦を行う。ここでは、残っている駒の半分の情報を得る。その際、現実的にも情報を得やすい駒として、自陣に近い駒から情報を得る。なお、途中局面からの対戦でも、これまでの初期盤面からの対戦と同数対戦を行う。

対戦は、開始局面にかかわらず先手後手500試合の計1,000試合行い、勝率で強さを評価する。今回は引き分けを0.5勝とする。引き分けは、大会*2の合計200手というルールに従う。実験は、CPU AMD Ryzen Threadripper 3970X 32-Core Processor, メモリ 256 GBのマシンで行った。実験には、対戦サーバとして geister_server*3, プレイヤ作成に Naotti-2020*4を使用した。プレイヤーの1手あたりのプレイアウト回数は50,000回とした。

5.1 実験で用いるプレイヤーについて

相手の駒色を推定するプレイヤーとして Cheating プレイ

ヤを用い、推定を行わないプレイヤーと対戦を行う。いずれも、不完全情報を扱うために3.2節のSO-ISMCTSを探索手法として用いる。ガイスターにおいては、脱出などの勝敗条件に関わるものを除いて、情報集合によって可能な行動が変わることがないため、情報集合に適応する指し手だけを選択するなどの処理が必要ない。そのため、アルゴリズム1に記載したMCTSからの違いとしては、探索のルートノードを訪問するたびに完全情報化を行うことと、完全情報化で得られた局面 s' をそれ以降現れる $s()$ で利用することである。すると、実験で用いるプレイヤーの動作は、アルゴリズム2のようになる。以降、このアルゴリズムではUCTを用いており、推定を用いる Cheating プレイヤを「Cheating UCT」と呼ぶことにする。

Algorithm 2 IS-MCTS

```

function ISMCTS( $s_0$ )
  不完全情報である局面  $s_0$  に対応するルートノード  $n_0$  の生成
  while 訪問回数が閾値未満 do
    不完全情報である局面  $s_0$  から生じ得る、完全情報なゲーム局面  $s'_0$  を生成し、以降の  $s()$  を得られた完全情報  $s'()$  で置き換える。
     $n := \text{TREEPOLICY}(n_0)$ 
     $v := \text{PLAYOUTPOLICY}(s'(n))$ 
    BACKUP( $n, v$ );
  return  $n_0$  の子ノードで訪問数最大のノードに相当する指し手

```

なお、UCB値の計算における c には $\sqrt{2}$ を用いた。ルートノードを訪問するたびに駒色を決定しているため、プレイアウトごとに駒色の配置は変わっている。また、取得してきた駒色情報は、完全情報化と勝敗判定にだけ使用しており、「青なら取る」などの使い方はしていない。アルゴリズム中の完全情報化では、駒数や推定精度に基づいて決定する。

Cheating プレイヤの完全情報化では、推定精度を用いる。推定精度は、Cheatingによって得た相手の駒色情報の信頼度のように扱う。

- (1) 推定精度が100%なら、確定した情報として扱う。
- (2) 残りの青と赤の駒数が同数なら、推定精度をその駒色である確率として扱う。
- (3) 残りの青と赤の駒数に偏りがある場合、情報のない駒の精度を50%として、残りの駒枚数などを加味した確率を計算し、その駒色である確率として用いる。

(3)については、残り駒が青1赤4の場合を考えると、駒数から赤である確率は80%となる。このとき、赤である推定精度が70%だったとして単純に精度を確率としてもちいることを考えると、他の駒よりも赤らしいと考えているにもかかわらず、単純割合から計算される80%よりも低い確率の赤として扱われることとなる。これを回避するため、推定精度を信頼度と考え、駒数から計算される確率と組み合わせて補正を行う (以下、推定精度を信頼度とも呼

*2 <http://www2.matsue-ct.ac.jp/home/hashimoto/geister/GAT/>, 最終アクセス日 2021年5月28日

*3 https://github.com/miyo/geister_server.java, 最終アクセス日 2021年5月28日

*4 <https://github.com/j1211/Naotti-2020>, 最終アクセス日 2021年5月28日

表 2 情報ごとの Cheating UCT の勝率

Table 2 Winning percentage of Cheating UCT for each estimation.

青\赤	0	1	2	3	4
0	(0.500)	0.828	0.892	0.906	
1	0.574	0.821	0.897	0.891	
2	0.639	0.829	0.874	0.905	
3	0.709	0.823	0.879	0.887	
4					0.914

ぶ). 推定精度を er とし, 青, 赤の総駒数を B, R , 推定した青, 赤の駒数を B_e, R_e とする (推定していない駒の信頼度は 50% と考える). 駒を青と推定する信頼度の総和 $S_B = er \times B_e + (1 - er) \times R_e + 0.5(B + R - B_e - R_e)$, 赤と推定する信頼度の総和 $S_R = (1 - er) \times B_e + er \times R_e + 0.5(B + R - B_e - R_e)$ を計算し, 補正にもちいる. $S_B = 0.5(B + R)$ のときは, 信頼度が 50% で青と赤の駒数が同数のような, 信頼度や駒数のバランスが取れているときであり, 各駒の推定精度をその駒色である確率として扱う. $S_B > 0.5(B + R)$ であれば, 青である信頼度が高いため割合に応じて青である確率を増やすことを考え, 青と推定した駒は $er \times B/S_B$, 赤と推定した駒は $(1 - er) \times B/S_B$ をその駒色である確率として扱う. もし $S_B < 0.5(B + R)$ であれば, 赤である信頼度が高いため赤である確率を増やす. 青と推定した駒は $1 - (1 - er) \times R/S_R$, 赤と推定した駒は $1 - er \times R/S_R$ をその駒色である確率として扱う. これを使うと, 青 1 赤 4 のケースでは, 推定した駒が赤である確率は 93.33% と計算され, 推定していない駒よりも高い確率になり, 問題の改善が確認できる.

5.2 推定する駒数と駒色の影響

情報を得る駒の色と駒数ごとに対戦を行った結果を表 2 にまとめた. 青 0 赤 0 のときの勝率は理想値 0.5 を記載した.

情報を得る駒数が増えるほどに勝率が高くなる傾向が得られたが, 青よりも赤の情報を得た方が顕著に有効である. 敗北条件の 1 つは相手の赤をすべて取ることであり, 1 つでも赤が分かっていたらその条件を回避できることから説明できる. また, 赤と分かった駒以外をすべて取れば, 必ずその中に青 4 個が含まれるため勝ちとなる. 一方で, 青駒は 4 個取らないと勝利とはならないので 1, 2 個分かっているでもそれが勝利に結びつきにくいと考えられる.

続いて, 勝利条件と敗北条件の結果を表 3 にまとめる. 勝利条件は, 青駒を脱出させる, 相手の青駒をすべて取る, 自分の赤駒をすべて取らせる, の 3 つである.

赤が 1 つでも分かっていると, 赤を取った敗北 (表の右端) が大幅に減少している. また, 青の脱出による敗北も大幅に減っている. 一方で青について赤の情報がない場合に着目すると, 青の駒数が増えるほど赤を取った敗北が減

表 3 Cheating UCT の勝利条件と敗北条件

Table 3 Winning and losing condition of Cheating UCT for each estimation.

	勝利			敗北			引分
	脱出	青駒	赤駒	脱出	青駒	赤駒	
青 0 赤 1	354	205	254	106	18	33	30
青 0 赤 2	367	187	309	52	20	8	57
青 0 赤 3	282	252	333	46	7	2	78
青 1 赤 0	267	60	211	191	34	166	71
青 1 赤 1	345	190	263	95	27	35	45
青 1 赤 2	345	224	298	49	18	7	59
青 1 赤 3	266	264	310	46	11	1	102
青 2 赤 0	279	78	251	173	23	134	62
青 2 赤 1	332	202	275	97	17	37	40
青 2 赤 2	334	217	284	61	19	8	77
青 2 赤 3	270	256	336	41	11	1	85
青 3 赤 0	286	135	258	189	19	53	60
青 3 赤 1	283	212	302	116	13	22	52
青 3 赤 2	263	248	330	63	11	10	75
青 3 赤 3	270	252	321	61	6	3	87
青 4 赤 4	210	310	348	27	11	2	92

表 4 駒位置と色ごとの Cheating UCT の勝率

Table 4 Winning percentage of Cheating UCT for each color and position.

色\場所	1	2	3	4
青	0.550	0.574	0.573	0.595
赤	0.779	0.833	0.825	0.875



図 2 初期位置の分類

Fig. 2 Classification of starting position.

少し, 青の脱出による敗北は減少しない. 相手の青の情報が増えているため直感に反する結果といえる.

最後に, 情報を得る駒について, 青を 1 枚だけのケース, 赤を 1 枚だけのケースそれぞれについて, どの位置の情報を得るかのデータを得た結果を表 4 にまとめる. 場所については, 初期位置を対称性から図 2 の 4 パターンに分類しており, 表における数字 1-4 は図 2 における番号が示す場所を表している. いずれも青よりも赤を推定する方が勝率が高く, 今までの傾向と一致している. 傾向として位置 4 の勝率が高く, 位置 1 の勝率が低い. 相手のゴールに最も近い位置 1 にいる駒は前進して取られやすく, 後列でもゴールからもっと遠い位置 4 の駒はなかなか動けず取られにくい. そのため, 情報を得た駒が終盤まで残りやすい位置 1 の勝率が低く, 終盤まで残りやすい位置 4 の勝率が高くなったと考えられる. また, 位置ごとの勝敗条件を表 5

表 5 駒位置と色ごとの勝利・敗北条件

Table 5 Winning and losing condition for each color and position.

	勝利			敗北			引分
	脱出	青駒	赤駒	脱出	青駒	赤駒	
青 場所 1	64	16	49	58	3	43	19
青 場所 2	70	17	57	46	13	45	14
青 場所 3	61	9	59	47	10	37	16
青 場所 4	72	18	46	40	8	41	22
赤 場所 1	85	41	70	34	6	12	10
赤 場所 2	86	46	58	25	3	7	8
赤 場所 3	86	52	63	25	7	7	9
赤 場所 4	97	66	63	22	2	7	3

表 6 推定精度ごとの Cheating UCT の勝率

Table 6 Winning percentage of Cheating UCT with various accuracy of estimation.

精度	60	70	80	90	100
勝率	0.5205	0.54	0.574	0.6615	0.809

表 7 推定精度と勝利・敗北条件

Table 7 Winning and losing condition with various accuracy of estimation.

精度	勝利			敗北			引分
	脱出	青駒	赤駒	脱出	青駒	赤駒	
60	187	30	240	321	34	61	127
70	182	45	262	307	39	63	102
80	213	58	247	267	46	57	112
90	225	73	321	220	30	46	85
100	260	173	352	112	26	29	48

にまとめたが、それぞれ約 250 ゲームであり、有意な差はない傾向にも特に差は見られなかった。

5.3 推定精度の影響

初期局面で駒の情報を間違いなく得ることはほぼ不可能である。そこで精度を導入する。与えられる情報に誤りが含まれる場合とプレイヤーが与えられた情報を疑う場合が考えられ、今回は後者を扱う。

それぞれ青 2、赤 2 の情報が与えられるとして、精度を 60% から 100% までの 10% 刻みとしたときの対戦結果を表 6 に、勝利条件と敗北条件について表 7 にまとめる。

精度が高いほど勝率は高く、精度 0.7 以上で有意な差が得られた。また、精度が高くなるにつれ、相手青の脱出と相手赤を取ることに伴う敗北が減少し、自分の赤が取られる勝利が増加している。ほかにも精度の増加にともない、引き分けの減少が見られる。

精度を 100% へ変化させると勝率が大幅に上昇しており、この理由としては精度 100% で推定した駒を確定させられることと実験に用いたプレイヤーが保守的であることが原因として考えられる。精度が 100% でない限り青駒と推定

表 8 途中局面から開始したときの Cheating UCT の勝率

Table 8 Winning percentage of Cheating UCT (starting positions: after 20 moves played).

開始局面 (推測駒数)	精度				
	60	70	80	90	100
青 4 赤 4 (4)	0.4995	0.54	0.592	0.6705	0.788
青 3 赤 3 (3)	0.5155	0.548	0.5815	0.6675	0.784
青 2 赤 2 (2)	0.449	0.5155	0.542	0.6145	0.773

表 9 途中局面からの対戦における推定精度と勝利・敗北条件 (青 2 赤 2)

Table 9 Winning and losing condition of Cheating UCT (starting positions: after 20 moves played. estimated colors: (#blue, #red) = (2,2)).

精度	勝利			敗北			引分
	脱出	青駒	赤駒	脱出	青駒	赤駒	
60	187	18	210	311	36	169	68
70	223	20	243	291	43	121	59
80	239	29	235	273	42	102	78
90	198	63	306	230	39	66	95
100	303	140	291	118	33	37	78

される駒は赤駒の可能性があり、青を取って勝てる局面が赤を取って負けの可能性を含むために選ばれなくなってしまふ。表 7 の勝敗条件についての結果を見ると、相手の青駒の脱出による敗北が大きく減り、相手の青駒を取っての勝利が大幅に上昇しており、青駒を取ることに伴う勝敗の増減は上記の理由で説明できる。

5.4 対戦開始局面の影響：途中局面からの対戦

現実的な推定を考えると、初期局面での推定は実際には不可能である。また推定する対象の色を指定することもできない。より現実的な設定に近づけるため、途中局面を対象とし、情報を得やすい駒から情報を得て、推定の有効性について評価を行う。

途中局面については、過去の対戦棋譜から取り出す。20 手目で互いに駒が同数残っている局面を対象とした。その中から、青 4 赤 4、青 3 赤 3、青 2 赤 2 となっている局面を対象として対戦を行う。情報を得る対象は、動いた駒ほど情報を得やすいとして、自陣に近い駒が対象となる。いずれも残り駒の半分が推定対象となり、相手の駒位置によっては推定する駒色に偏りが生じることとなる。

結果を表 8、表 9 にまとめる。青 2 赤 2 以外は精度と勝率の関係も勝敗条件も、初期局面から始めた場合と同様の傾向だった。一方で、青 2 赤 2 についてはゲームの終局が近いのか形勢の偏りが多く、傾向の変化が大きい傾向があり、勝率が全体的に低く、勝敗条件も 90% で大きく変動している。また、精度の増加にともなう引き分けの減少が見られないことも特徴である。

6. 議論

実験結果から、推定の有効性や得られた知見について議論する。

初期局面において相手のすべての情報を得る Cheating プレイヤの勝率は 0.9 を超えており、情報を得ることによる強化は明確であり、ガイスターにおいて相手駒色の推定は有効であると考えられる。さらに、すべての情報ではなく、相手の一部の駒情報を得る実験でも勝率は有意に高くなったため、すべての情報ではなく一部の情報を推定するだけでも効果があると考えられる。また、初期局面だけでなく途中局面からの対戦実験も行い、初期局面のときと同様に一部の情報でも有意に強化され、対戦相手の情報を得ることが有効であると確かめられた。最後に、初期局面と途中局面とで一部の情報を得たものとして、その推定精度を変えながら実験を行った。推定精度が 70% 程度から有意に勝ち越すようになっており、ある程度の精度があれば推定が有効であるという結果が得られた。

ほかに得られた知見として、推定する駒色による有効性の違いがある。赤 1 つの情報を得ることで勝率が 0.8 を超える一方で、青 1 つだけではそこまで有効ではなかった。推定した赤駒はそれ以外の駒をすべて取れば勝ちになる一方で、青駒はそこまでの有効性がないことが原因であり、ガイスターの特徴についての知見が得られた。また、推定する駒の初期位置については差は得られたものの、情報を得た駒が終盤まで残りやすいかどうかの影響していると考えられ、有用な知見とは言い難い。

推定精度を変えて行った実験では、精度 90% であっても精度 100% と比べると勝率がかなり低く、これは推定精度が高くても駒を確定できないことに起因していると考えられる。今回用いたプレイヤが負ける可能性のある手を避ける、保守的なプレイヤであることも原因と考えられる。今回の結果から、プレイヤにリスクをとることで改善が考えられ、今回の実験結果が推定精度と探索時の勝率がどのようときにリスクをとるかの判断基準を設けることに有用と考えられる。

7. おわりに

不完全情報ゲームであるガイスターは、囲碁将棋チェスなどの二人零和有限確定完全情報ゲームとは不完全情報以外は変わらないこと、その情報によって可能な手が変わらないことなどの特徴があるゲームである。

不完全情報ゲームにおいては、その情報の推定が重要と考えられる。一方で、大貧民などのゲームでは正確に推定することが有効でないことも知られている。

本研究では、ガイスターにおいて推定が有効であるかを調査した。本来知りえない未知の情報を知るプレイヤと対戦し、その勝率から有効性を評価する方法を利用した。結

果から、推定する駒数が多いほど、また、推定精度が高いほど勝率が良くなり、ガイスターにおいて相手駒色推定が有効であることが確認された。また、重視すべき特徴として青駒よりも赤駒を推定する方が有効であることなどの知見が得られた。

推定精度を変えての実験からは、精度が 70% から効果があることが確認できたほか、推定精度が 100% のときと比べて保守的な振舞いとなっているため、リスクをとることで性能が改善する可能性があることを確認した。さらに、より現実的な推定として、途中局面からの推定を行い、その設定でも推定が有効であることを示した。

今後の研究としては、実際のプレイヤの推定精度や推定方法を調査し、より現実 に即した条件での有効性の調査が考えられる。また、ランダムプレイヤ相手では推定はほぼできないなど、対戦相手によって推定のしやすさが変わるため、対戦相手の影響についての調査なども考えられる。

謝辞 本研究は JSPS 科研費 17K12807, 20K12124 の助成を受けたものです。

参考文献

- [1] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K. and Hassabis, D.: A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play, *Science*, Vol.362, No.6419, pp.1140–1144 (2018).
- [2] 伊藤毅志, 保木邦仁, 三宅陽一郎: ゲーム情報学概論—ゲームを切り開く人工知能, コロナ社 (2018).
- [3] 川上直人, 池田 心, 石井岳史, 橋本 剛: 後退解析による詰めガイスター問題の列挙, 情報処理学会研究報告, Vol.2020-GI-43, No.13, pp.1–8 (2020).
- [4] Sakuta, M. and Iida, H.: Solving Kriegspiel-like Problems: Exploiting A Transposition Table, *ICGA Journal*, Vol.23, No.4, pp.218–229 (2000).
- [5] Ciancarini, P. and Favini, G.P.: Monte Carlo tree search in Kriegspiel, *Artificial Intelligence*, Vol.174, No.11, pp.670–684 (2010).
- [6] Schäfer, J., Buro, M. and Hartmann, K.: The UCT algorithm applied to games with imperfect information, Diploma, Otto-Von-Guericke Univ. Magdeburg, Magdeburg, Germany, Vol.11 (2008).
- [7] Long, J.R., Sturtevant, N.R., Buro, M. and Furtak, T.: Understanding the success of perfect information monte carlo sampling in game tree search, *24 AAAI Conference on Artificial Intelligence* (2010).
- [8] 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, Vol.2011-MPS-85, No.9, pp.1–6 (2011).
- [9] Cowling, P.I., Powley, E.J. and Whitehouse, D.: Information Set Monte Carlo Tree Search, *IEEE Trans. Computational Intelligence and AI in Games*, Vol.4, No.2, pp.120–143 (online), DOI: 10.1109/TCIAIG.2012.2200894 (2012).
- [10] Browne, C., Powley, E., Whitehouse, D., Lucas, S., Cowling, P., Rohlfshagen, P., Tavener, S., Perez, D., Samothrakis, S. and Colton, S.: A Survey of Monte Carlo Tree Search Methods, *IEEE Trans. Computational Intelligence and AI in Games*, Vol.4, No.1, pp.1–43 (2012).
- [11] 西野順二, 西野哲朗: 多人数不完全情報ゲームのモンテ

- カルロ木探索における推定の効果, 情報処理学会研究報告, Vol.2011-BIO-27, No.31, pp.1–4 (2011).
- [12] 末續鴻輝, 織田祐輔: 機械学習を用いないガイスターの行動アルゴリズム開発, GAT2018 論文集, Vol.2018, pp.13–16 (2018).
 - [13] 末續鴻輝, 織田祐輔: ガイスターにおける必勝局面を利用した敵駒推定, 情報処理学会研究報告, Vol.2020-GI-43, No.19, pp.1–4 (2020).
 - [14] 三塩武徳, 小谷善行: ゲームの不完全情報推定アルゴリズム UPP とそのガイスターへの応用, 情報処理学会研究報告, Vol.2014-GI-31, No.4, pp.1–6 (2014).
 - [15] 寺村舞童華, 松崎公紀: ニューラルネットワークを用いたガイスターの相手駒色推定とその拡張, 情報処理学会研究報告, Vol.2021-GI-45, No.6, pp.1–8 (2021).

付 録

A.1 不完全情報における MCTS について

今回の実験では, プレイヤの探索手法に SO-ISMCTS を用いた. 他の手法として, カードゲームなどではモンテカルロ法 (MC) が使われているほか, 不完全情報の扱い方が異なる Determinized UCT などの手法もあり, SO-ISMCTS を用いる正当性を説明する.

不完全情報の扱い方として, Determinized UCT, SO-ISMCTS の 2 種類がある (またガイスターでは紫駒と呼ばれる手法もある). 前者については複数の木があるため, 指手決定の手法がいくつか考えられる. たとえば, 訪問回数, 各木の最善手の多数決, 各木の勝率, 各木の Lower Confidence Bound (LCB) などである. 特に, ある手があるパターンでは負けるという場合, その手を選ぶのはリスクが高い. そのような手を選ぶことを避けるために, 全部の木の最低値を利用することが考えられる.

200 ゲームの結果を表 A.1 にまとめる. 表の勝率は, 最左部で A-B として表記される A の勝率である. いずれも MCTS は MC に対して勝ち越したほか, 今回採用した SO-ISMCTS は他の手法に対して勝ち越しており, 実験に SO-ISMCTS を用いたことは妥当であったと考えられる.

表 A.1 不完全情報下のモンテカルロ木探索どうしの対戦結果
Table A.1 Tournament results among MCTS players in imperfect information setting.

手法	Det				SO
	訪問数	多数決	勝率	LCB	訪問数
MCTS-MC	0.630	0.580	0.738	0.615	0.835
Det-訪問数		0.630	0.115	0.140	0.138
Det-多数決			0.095	0.100	0.095
Det-勝率				0.613	0.37
Det-LCB					0.333



竹内 聖悟 (正会員)

2005 年東京大学教養学部広域科学科卒業. 2007 年同大学大学院総合文化研究科修士課程修了. 2010 年同博士課程修了. 博士 (学術). JST ERATO 研究員, 東京大学研究員等を経て, 2018 年より高知工科大学講師. ゲーム情報

学分野に興味.



栃川 純平

2020 年高知工科大学情報学群卒業. 同年より同大学大学院修士課程に進学.



松崎 公紀 (正会員)

2001 年東京大学工学部計数工学科卒業. 2003 年同大学大学院情報理工学系研究科修士課程修了. 2005 年同博士課程中退. 博士 (情報理工学). 東京大学助手, 助教, 高知工科大学准教授を経て, 2018 年より同教授となり現在に至る. 並列プログラミング, アルゴリズム導出, ゲームプログラミング等に興味を持つ. 日本ソフトウェア科学会, ACM, IEEE 各会員.