Vol.2014-GI-31 No.4
2014/3/17

# 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

三塩武徳[†1]　　小谷善行[†1]

ゲームの不完全情報の推定を行うアルゴリズム Using Past Playout(UPP)を提案する．UPP はモンテカルロ法において過去のシミュレーション結果のうち現在局面に至るものを取り出し，仮定した情報の間の勝率を比較する．相手側の勝率が高い部分は実際の局面と等しい可能性が高い.これを使って不完全情報の推定を行う．
アレックス・ランドルフ（Alex　Randolph）[1]によって発表された二人零和確定不完全情報ゲームである「ガイスター」において UPP を用いたプログラムと既存手法の猪突戦法，および通常のモンテカルロ法とで対局を行った．結果，猪突戦法に対しては思考時間 0. 25 秒で 94%の勝率，モンテカルロ法との対局ではお互いの思考時間 1 秒で 55%の勝率を挙げた．これらの結果より，ガイスターにおける UPP の有効性を示した．

# Estimation Algorithm UPP for Imperfect Information in Games and Application for Geister

TAKENORI MISHIO[†1]　YOSHIYUKI KOTANI[†1]

We propose an algorithm Using Past Playout (UPP) which estimates incomplete information of the game. The algorithm UPP extracts the playouts of current position from the simulation results of the past, and compares the winning percentages between the assumed information. The higher the part the other side's winning percentage is, the higher the possibility equal to actual aspects is. It estimates the incomplete information with it.
We performed experiment of playing using UPP, Foolhardiness (Chototsu) Tactics and normal Monte Carlo method in the game "geister", two person zero sum determined incomplete information game, which was invented by ALEX RANDOLPH[1]. As a result, UPP listed a winning percentage of 94% in 0.25 seconds thinking time against Foolhardiness (Chototsu) Tactics and 55% in one seconds thinking time for both against normal Monte Carlo method.
The results show the effectiveness of the UPP in it.

## 1. はじめに

本研究では，不完全情報ゲームの一つである「ガイスター」をテーマとして取り上げている．「ガイスター」は「ゴースト」「ファンタスミ」とも呼ばれる 1982 年にドイツのアレックス・ランドルフ（Alex　Randolph）によって発表された二人零和確定不完全情報ゲームである．

このゲームでは，6×6 マスの盤面とプレーヤーごとに 8 個の駒を用いてゲームを行う．駒は良い駒と悪い駒の二種類存在し．それぞれ 4 個ずつある．

**- 初期配置 -**

ゲームの開始時に自分の駒を 2×4 のマスで囲まれた陣地内に一つずつ好きな並びで配置する．ゲーム中に自分の駒の種類を確認することは出来るが，相手の駒の種類は相手の駒を取った時にのみ確認することが出来る．取った駒の種類から盤面に残っている相手の良い駒と悪い駒の個数だけは知ることが出来る．

**- ゲームの流れ -**

先攻のプレーヤーと後攻のプレーヤーで交互に手番を行っていく．自分の手番では，自分の駒を一つ選択し上下左右のいずれかに動かすことができる．自分の駒の移動先に相手の駒がある場合はその駒を取る．自分の駒を取ることはできない．盤面の四隅のマスのうち自分の陣地から遠い二つは自分の出口となっており1, 2自分の良い駒が出口のマスの上にいる時は四方向に動く代わりに「脱出」を選択することが出来る．移動を行うと相手の手番となる．

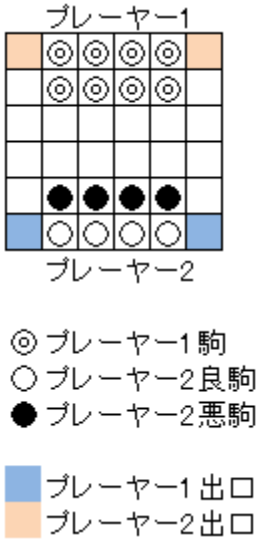


図 1　「ガイスター」初期配置例

†1 東京農工大学
Tokyo University of Agriculture and Technology

**- 勝利条件 -**

ゲーム中に次のいずれかを満たしたプレーヤーがいる場合，即座にそのプレーヤーの勝ちとなる．

- 自分の良い駒で「脱出」を行う（いずれか一つでよい）
- 相手の良い駒を全て取る
- 自分の悪い駒を全て相手に取らせる

ガイスターをテーマとした先行研究の中で，猪突戦法という着手決定方法が述べられている[2][3]．本実験ではこの手法を比較対象として利用しておりそのアルゴリズムを述べる．

**猪突戦法**

猪突戦法では，次のような初期配置と着手決定方法を用いてゲームを進めていく．

**- 猪突戦法の初期配置 -**

猪突戦法では下図のように外側の 4 つの駒を良い駒，内側の 4 つが悪い駒となるように初期配置を行う．

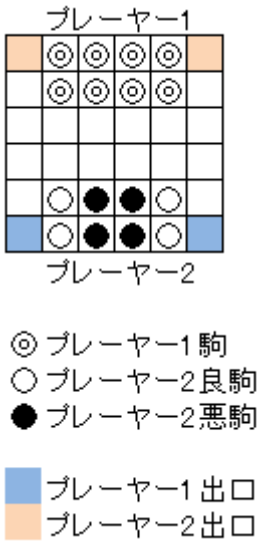


図 2　猪突戦法初期配置

**- 手番に指す手 -**

猪突戦法のプレーヤーは次のようにして着手の決定を行う．

- 自分の良い駒が脱出できる場合には脱出を行う．
- 脱出はできないが，自陣から最も遠い行（自分の出口を含む横並びの 6 マス）にたどり着いている自分の良い駒が有る場合，近い方の自分の出口に向かう手を指す．
- 最も遠い行にたどり着いている駒がない場合は，全ての自分の良い駒のうち最も遠い行に一番近いものを選択し上に進める．2 つ存在する場合はいずれか一つを選択する．

## 2. 不完全情報ゲームにおけるモンテカルロ法の課題

モンテカルロ法は思考ゲーム研究において良く用いられている手法の一つであり，多数回のランダムシミュレーションから最も成績の良かったものを選択する手法である．このランダムシミュレーションのことをプレイアウトという．代表的なものでは囲碁のプログラムにモンテカルロ法を利用したモンテカルロ囲碁などがあり，プログラムの一例として Sylvain Gelly らの作成した Mogo がある[4]．また，このモンテカルロ法の性能を上げようとする研究もいくつか存在し，Simulation Balancing はこのモンテカルロ法の中のプレイアウト性能を向上させようとした研究である[5]．

完全情報ゲームにおけるモンテカルロ法と不完全情報ゲームにおけるモンテカルロ法の最大の違いは，プレイアウト時に不完全情報を仮定する必要がある点である．不完全情報の仮定がモンテカルロのプレイアウト時にどのように影響するのか，図を用いて説明する．

次の二つの図は，完全情報ゲームにおいてのモンテカルロ法と不完全情報ゲームにおいてのモンテカルロ法を表した図である．

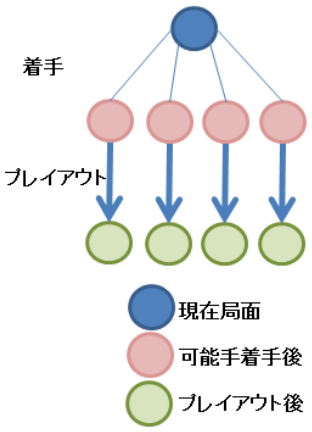


図 3　完全情報ゲームにおけるモンテカルロ法

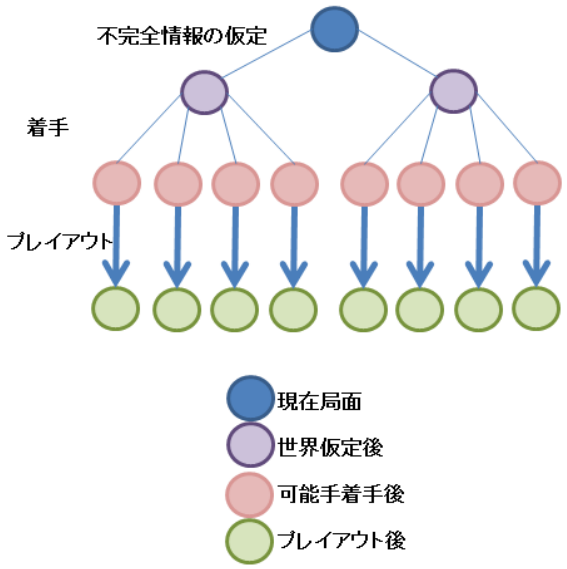


図 4　不完全情報ゲームにおけるモンテカルロ法

これらの図からわかるように，世界の仮定をしているために完全情報ゲームの場合と比べて不完全情報ゲームにおいては（1/世界の数）のプレイアウトしか割り振ることができない．また，世界の仮定によってプレイアウトの結果は大きく異なる場合がほとんどで，ある世界に対して非常に勝率の高い手が別の世界では非常に勝率の低い手となることがある．各世界からの異なる意見をどのようにまとめればよいかという問題も存在する．

これらの不完全情報ゲームに対してモンテカルロ法を適用した場合の課題を緩和しようとした先行研究が存在する[6][7]．研究内容からもわかるように，不完全情報ゲームの研究において仮定された不完全情報をどのように扱うのかというのは大きな課題となっている．提案手法では，過去のプレイアウト結果を利用することで，可能性の高い世界を推測する世界評価アルゴリズムを利用してこの問題を緩和することを目的としている．

## 3. アルゴリズム UPP

提案手法であるアルゴリズム UPP について述べる．アルゴリズム UPP の入力と出力は次のとおりである．

入力：現在の局面，一手前と二手前の着手，二手前の着手決定に用いたプレイアウト結果
出力：世界の評価値と着手

UPP は次の 5 つのステップにより構成されている．

**ステップ 1：着手確認ステップ**

直前の相手着手ともうひとつ前の自分の着手を取り出す．

**ステップ 2：プレイアウト結果の参照・比較ステップ**

二つ前の手番での着手決定に用いたプレイアウトの結果のうちステップ 1 で取り出した着手から始まるプレイアウトのみに注目する．これらのプレイアウト結果を，直前に相手が動かした駒を良い駒としている世界と悪い駒としている世界の二つに分け，それぞれの勝率を平均する．

**ステップ 3：不完全情報の推定ステップ**

次の二つの式を用いて各世界の評価値の更新を行う．この評価値が高いほど，可能性の高い世界であるということになる．

$$W_n = \begin{cases} 1 + t_n c & (F_n = 1) \\ 0 & (F_n = 0) \end{cases}$$

n:世界の番号

$W_n$: 世界 n の評価値

$t_n$: 世界 n の評価値

c:加算値

$F_n$: 世界 n が存在するか（1：存在する　0：存在しない）

この式は，世界の評価値を求める式である．

c は定数であり，これが大きいほど一回の評価で評価値が大きく変化する．

$F_n$については，ステップ 5 で詳しく説明する．

- if（「世界 n において直前の相手の駒は良い駒とされている」かつ「直前の相手の駒が悪い駒とされている場合の勝率が高い」）$t_n = t_n + +$
- else if（「世界 n において直前の相手の駒は悪い駒とされている」かつ「直前の相手の駒が良い駒とされている場合の勝率が高い」）$t_n = t_n + +$

この式はパラメータ$t_n$を求める式である．パラメータ$t_n$とパラメータ$W_n$は比例しており，この値が大きいほど評価値も大きい．

ステップ 1 からステップ 3 を図で説明する．

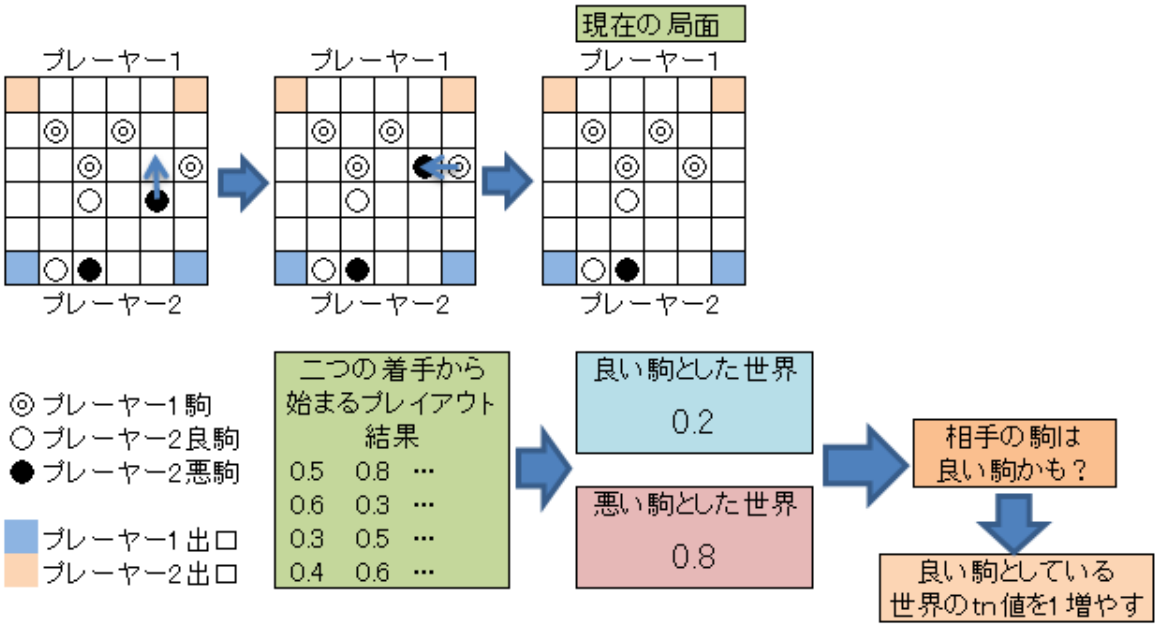


図 5　UPP ステップ 1-3

**ステップ４：重みつきモンテカルロ法**

世界の評価値と次の式を用いてモンテカルロ法を行い着手の決定をする．

$$P(W_n)=\frac{W_n}{\sum_k W_k}$$

$W_n$: 世界 n の評価値

$P(W_n)$: 世界 n にプレイアウトが割り振られる確率

①この式の確率で世界を決定

②決定した世界の各可能手に一回ずつプレイアウトを割り振る

この①と②を与えられた時間内で繰り返し行い，最後に各可能手の世界ごとの勝率を平均して最も勝率の高かった手を着手として選択する．

このステップで行ったプレイアウトの結果は次の着手決定時に利用するため，全て保存しておく．

**ステップ５：世界の刈り取りステップ**

最後に，ステップ 4 で選択された着手が相手の駒をとるものだった場合，その駒の正体と食い違った仮定をしている世界の Fn 値を 0 にする．例えば相手の悪い駒をとった場合には，その駒を良い駒としている全ての世界の Fn 値を 0 にする．これでありえない世界にプレイアウトが割り振られることがなくなる．

## 4. 実験概要と結果

本研究では，アルゴリズム UPP と世界評価を行わないモンテカルロ法，猪突戦法の３つの手法を対局させた．これらの対局において，UPP の加算値 C は 0.1 に設定した．これは予備実験で最も高い性能を上げた時の値を利用した．

**猪突戦法との対局実験**

世界評価無しのモンテカルロ法とアルゴリズム UPP の思考時間を様々に変更して猪突戦法を相手に対局実験を先攻後攻それぞれ 200 試合ずつ合計 400 試合行った．

世界評価無しのモンテカルロ法の対局結果は次のようになった．

表 1　モンテカルロ法の猪突戦法との対局結果

| 思考時間(秒) | 勝利数(回) | 試合数（回） | 勝率 |
|---|---|---|---|
| 0.25 | 318 | 400 | 0.80 |
| 0.5 | 371 | 400 | 0.93 |
| 1 | 381 | 400 | 0.95 |
| 2 | 387 | 400 | 0.97 |
| 4 | 385 | 400 | 0.96 |

アルゴリズム UPP との対局結果は次のようになった．

表 2　アルゴリズム UPP の猪突戦法との対局結果

| 思考時間(秒) | 勝利数(回) | 試合数（回） | 勝率 |
|---|---|---|---|
| 0.25 | 375 | 400 | 0.94 |
| 0.5 | 380 | 400 | 0.95 |
| 1 | 391 | 400 | 0.98 |
| 2 | 391 | 400 | 0.98 |
| 4 | 384 | 400 | 0.96 |

これらの結果をまとめたものが次の図である．

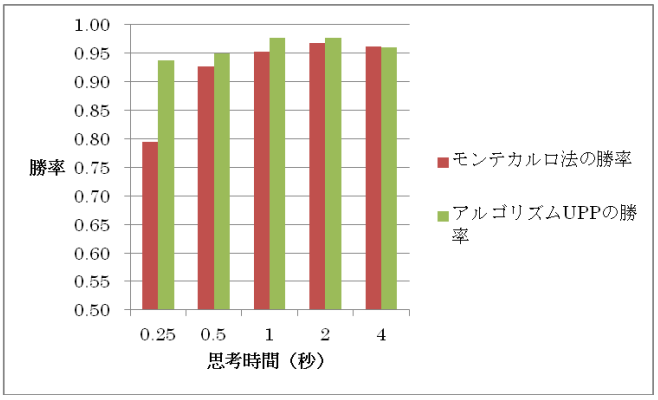


図 6　アルゴリズム UPP，モンテカルロ法の猪突戦法との対局結果

**モンテカルロ法とアルゴリズム UPP の対局実験**

モンテカルロ法とアルゴリズム UPP を用いたプログラムでの対局実験を行った．お互いの思考時間は 1 秒で，先攻後攻を入れ替えて 200 試合ずつの合計 400 試合行った．対局結果は次のようになった．

表 3　アルゴリズム UPP とモンテカルロ法の対局結果

| | 平均プレイアウト数（回/秒） | 勝利数（回） | 試合数（回） | 勝率 |
|---|---|---|---|---|
| UPP | 7208 | 221 | 400 | 0.55 |
| モンテカルロ | 7550 | 179 | 400 | 0.45 |

## 5. 考察

**猪突戦法との対局実験**

この実験は世界評価を行わないモンテカルロ法とアルゴリズム UPP を猪突戦法と比較してどちらの性能が高いかを確認することと，アルゴリズム UPP とモンテカルロ法ではどちらがより猪突戦法に対して有利かを確認することを目的として行った．

世界評価を行わないモンテカルロ法は，思考時間 0.25 秒で猪突戦法に対して 8 割の勝率を持っていることが分かる．一方アルゴリズム UPP は思考時間 0.25 秒で猪突戦法に対して 9 割以上の勝率を持っていることが分かる．いずれの手法も猪突戦法に対して高い性能を持っていることが分かるものの，この思考時間ではアルゴリズム UPP がより猪突戦法に対して有利であることが分かる．

思考時間を 0.5 秒に増やすとモンテカルロ法の勝率は 9 割に上昇するが，アルゴリズム UPP の勝率はほとんど変化しない．この結果とこれ以上思考時間を増やした実験結果から考えると，世界評価無しのモンテカルロ法は思考時間 0.5 秒の時点で性能の向上は見込めなくなり，アルゴリズム UPP は思考時間 0.25 秒の時点で性能の向上が見込めなくなっていることが分かる．また，どちらの手法も猪突戦法に対する勝率は 9.5 割ほどで収束するということも分かる．

**モンテカルロ法とアルゴリズム UPP の対局実験**

この実験ではモンテカルロ法とアルゴリズム UPP を対局させることにより比較し，どちらの性能が高いかということを確かめることを目的とした．

表 3 からはアルゴリズム UPP がモンテカルロ法に対して 5.5 割の勝率を持っていることが分かり，アルゴリズム UPP が「ガイスター」において世界評価無しのモンテカルロ法よりも高い性能を持っていることが分かる．

また，平均プレイアウト回数に注目するとアルゴリズム UPP の方が少ないことが分かる．これは世界評価を行うこ

とによる性能向上がプレイアウト回数の減少による性能低下を上回っているということであり,「ガイスター」において世界評価を取り入れることが非常に有効であるということが分かる.

さらに，有意水準５％のカイ二乗検定をこの結果に対して用いたところ，カイ二乗値は 4.41 となり，自由度 1，有意水準 5%の場合の値 3.841 に比べ大きくなっていることからこの結果が有意であることも証明された.

## 6. おわりに

本研究では，不完全情報ゲームのために世界評価アルゴリズムを有するアルゴリズム UPP を設計し,「ガイスター」ゲームにおいて既存の手法に対して高い性能を持っていることを証明した．世界評価アルゴリズムは不完全情報ゲームにおける不完全情報の仮定による問題を緩和するために,過去の着手から相手の不完全情報を推定するものである．猪突戦法との対局で挙げた思考時間 0.25 秒での勝率 94%という結果や, モンテカルロ法との対局で挙げたお互いの思考時間 1 秒での勝率 55%という結果から，アルゴリズム UPP の「ガイスター」においての従来手法に対する性能の高さを証明できた.

今回は「ガイスター」にアルゴリズム UPP を適用したが,この手法は他のゲームや現実世界における不完全情報を含んだ問題をコンピュータが解決する場合に応用できると考えている．この手法が「ガイスター」以外の問題解決にも有意な結果を挙げ，人工知能研究の進歩に寄与することを期待している.

### 参考文献

1) ガイスター
http://ja.wikipedia.org/wiki/%E3%82%AC%E3%82%A4%E3%82%B9%E3%82%BF%E3%83%BC
2) 南雲夏彦 : GPCC 報告「ゴースト」と「ループトラックス」と「ドット&ボックス」,３４回プログラミング・シンポジウム報告集,pp195-199(1993)
3) 南雲夏彦 : GPCC 報告「ゴースト」と「ポジット」,３５回プログラミング・シンポジウム報告集,pp173-176(1994)
4) Sylvain Gelly, Levente Kocsis, Marc Schoenauer, Michèle Sebag, David Silver, Csaba Szepesvári, Olivier Teytaud: The grand challenge of computer Go: Monte Carlo tree search and extensions, Communications of the ACM volume55 Issue3 March 2012,pp106-113(2012)
5) David Silver, Gerald Tesauro: Monte-Carlo simulation balancing, ICML '09 Proceedings of the 26th Annual International Conference on Machine Learning, pp945-952(2009)
6) Daniel Whitehouse, Edward J.Powley, Peter I.Cowling: Determinization and information set Monte Carlo Tree Search for the card game Dou Di Zhu, 2011 IEEE Conference on Computational Intelligence and Games, pp87-94(2011)
7) Jiajia Zhang, Xuan Wang, Jing Lin, Zhaoyang Xu: UCT Algorithm in Imperfect Information. Multi-Player Military Chess Game, 11th Joint Conference on Information Sciences(2008)