

# Amazons へのカテゴリの適用

副田 俊介<sup>†</sup> 田中 哲朗<sup>††</sup>

Amazons での手のカテゴリを提案し、このカテゴリを用いた実現確率探索の有効性を示した。Amazons は二人零和有限完全情報ゲームで、チェスなどと比べ平均分岐数が多いことが特徴の一つである。そこで、同じように平均分岐数の多い将棋で成功した実現確率探索を Amazons に適用することを考える。そのために Amazons の手のカテゴリをいくつか提案し、それぞれについて Generic Game Server より得た棋譜を元に実現確率を計算した。この実現確率を用いたプログラムと単純な反復深化を行うプログラムを 100 ゲーム対戦させた結果、一手あたり 5,000 個の末端ノードを展開したら探索を打ち切るといった条件の元で、実現確率を用いたプログラムが 80 勝 20 敗 と大幅に勝ち越し、Amazons における実現確率探索の有効性と、提案したカテゴリの有効性を示された。

## Categories for Amazons Moves

SHUNSUKE SOEDA<sup>†</sup> and TETSURO TANAKA<sup>††</sup>

We have adopted Realization Probability Search on Amazons. Amazons is a two-person, zero-sum, finite, perfect information game. Compared to Chess, which is also a two-person, zero-sum, perfect information game, there are many choice of moves in each state of the game. This makes it difficult for full-width search - that went well in Chess - to succeed in Amazons. Realization Probability Search is a technique to acquire the parameters used in forward pruning and search extension. Realization Probability Search was first proposed for use in Shogi, automatically learning values from games by professional Shogi players. We have proposed move categories for Amazons, and calculated the realization probability of each category from records from Generic Game Server. We had a program using Realization Probability Search (RP) play 100 games against a simple Iterative-Deepening program (ID), each searching up to 5,000 node per move, resulting in 80 wins by the RP, showing the effectiveness of Realization Probability for Amazons as well as the usefulness of the categories we have proposed.

### 1. 導 入

Amazons は二人零和有限完全情報ゲームである。Amazons は序盤では合法手の数が多く、初期局面では 2,176 通りの合法手が存在する。ゲームが進むにつれて合法手は減っていくものの、チェスと比べて 1 局面あたりの平均の合法手の数 (平均分岐数) が多い。

全幅探索によって局面を評価する場合には深く読めなければ良い評価を行うことができない。平均分岐数の大きいゲームでは全幅探索では深く読むことができず、そのようなゲームに関しては全幅探索はチェス程の成功は収めていない。

同じように平均分岐数の多い将棋では、実現確率探索が大きな成功を収めた<sup>4)</sup>。また、将棋以外では Lines Of Action での実現確率でも同様に実現確率探索が有効であることが示されている<sup>3)</sup>。

本論文では Amazons において実現確率探索を行うための手のカテゴリを提案し、そのカテゴリを用いたプログラムを単純な反復深化を行うプログラムと対戦させることで、Amazons での実現確率探索の有効性と提案したカテゴリの有効性を示す。

#### 1.1 Amazons のルール

Amazons は二人のプレーヤが順に一般的には 10 × 10 の盤面の上で駒を動かして行き、先に駒を動かさなくなった (合法手がなくなった) 方のプレーヤが負けとなるゲームである。

各プレーヤはそれぞれ駒を 4 つ持っている。また、これ以降先手の駒の色を黒、後手の駒の色は白であるとする。駒を動かす際のルールを説明する。駒は下記のように動かすことができる。

<sup>†</sup> 東京大学大学院総合文化研究科広域科学専攻  
Department of General Systems Studies, Graduate  
School of Arts and Science, The University of Tokyo  
<sup>††</sup> 東京大学情報基盤センター  
Information Technology Center, The University of  
Tokyo

- (1) Amazons の駒はチェスのクイーンと同じ動きで動くことができる。
- (2) 駒を動かす際には駒や矢を飛び越えることはできない。
- (3) 駒が移動した後にその移動した駒の次に動くことのできる場所に矢を放たなければならない。
- (4) おかれた駒や矢が局面上から除かれることはない。

駒を移動して矢を放つことを(一手)打つと言う。自分の手番のときに4つの駒のすべてが動けなくなった方のプレイヤーの負けである。

## 2. 実現確率探索

一般に、探索によって局面を評価する場合には深く読めなければ良い評価を行うことができない。平均分岐数の大きいゲームでは全幅探索では深く読むことができず、そのようなゲームに関しては全幅探索はチェス程の成功は収めていない。

平均分岐数が多いゲームに関して有効な探索の方法として、見込みのなさそうな局面に関しては探索は行わない(前向き枝刈り)、見込みのありそうな局面に関しては余分に深く探索(探索延長)すると言った手法を用いることが多い。

しかし、見込みのある局面・ない局面に関する絶対的な判断の基準はなく、良い局面を枝刈りしてしまうことや悪い局面を多く探索してしまうことがある。従来は前向き枝刈りや探索延長の判断のパラメータは人の手で調整することが多かった。

鶴岡らによる実現確率探索<sup>4)</sup>は前向き枝刈りや探索延長と同様のことをシステムチックに実現している。この方法ではプロの棋譜から、ある特徴を持った手などの程度の確率で実際に指されるかを学習し、これに基づいて手を生成する。多くの探索では深さ打ち切りの反復深化を用いるが、実現確率探索では深さで打ち切るかわりに、実現確率による最良優先探索を行う。

鶴岡らは将棋での実現確率探索の有効性を示した。また、将棋以外では橋本らによる LOA への適用例<sup>3)</sup>がある。

## 3. Amazons への実現確率の適用

Amazons は、序盤では1局面あたりの合法手の数(平均分岐数)が多く、通常の初期局面では2,176通りの合法手が存在する。また、序盤では駒の自由度が重要となる。

一方で、ゲームが進むにつれ、平均分岐数は減る傾向にあり、50手進んだ時点では平均分岐数も50手程

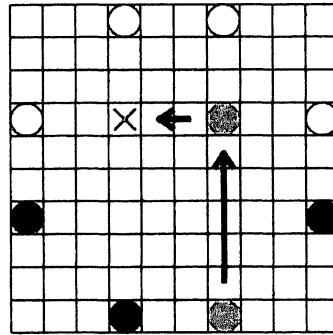


図1 BLOCK44 の例

度となる。また、終盤に行くにつれて駒の自由度よりも囲碁のように地に関する考察が重要となってくる。

### 3.1 カテゴリ

実現確率探索による前向き枝刈りと探索延長が特に有効に機能するのは平均分岐数の多い、ゲームの序盤から中盤にかけてとなる。そこでゲームの序盤で重要な駒の自由度への影響が大きい手を含むカテゴリを考える。

本研究では次のカテゴリを提案する：

- 相手の駒の利きがあるマスに自分の駒を動かす・矢を放つ手 (BLOCK)
- 相手の駒を閉じ込める手 (IMMOBILIZE)
- 相手の駒の隣のマスに矢を放つ手 (APPROACH)

BLOCK に関しては次のように更に細かく分類した：

- 自分の駒を動かしたマスへの利きがある相手の駒の数 ( $m$  個以上)
- 自分が矢を放ったマスへの利きがある相手の駒の数 ( $n$  個以上)

これを  $BLOCK_{mn}$  と表記する。

図1に BLOCK44 の例を示す。なお、マスへの利きがある相手の駒を数える際には自分の駒を動かす前の状態の盤面で数えている。

次に、図2に IMMOBILIZE に分類される手の例を示す。なお、図2は BLOCK11 と APPROACH の例にもなっている。

### 3.2 実現確率の計算

次に、それぞれカテゴリの実現確率を、Generic Game Server(GGS)<sup>1)</sup>の棋譜1,102局52,259手を分析した。

Amazons は序盤と中盤、終盤では大きく性質の違うゲームであるため、初期局面からの手数によって結果を分類してある。具体的には1手目から8手目ま

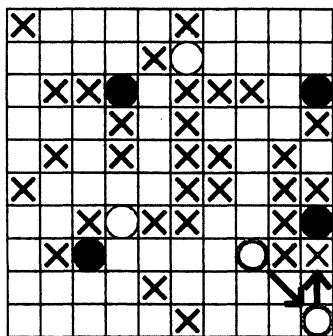


図2 IMMOBILIZEの例

での手の実現確率, 9手目から17手目までの実現確率... というように8手ごとにわけて考えている。

各カテゴリについて、「カテゴリに含まれる手が存在した数」と「カテゴリに含まれる手が実際に指された数」とそれぞれ数え、実現確率を計算した。結果を表1に示す。この表でAllとあるのは全ての手にマッチするカテゴリの実現確率である。これは分岐数の逆数となっている。

表でn/aとなっているのは、棋譜に存在する局面から生成される手で、そのカテゴリに分類されるものがなかったことを示している。

なお、Allよりも実現確率の低いカテゴリも存在する。これは「このカテゴリに含まれる手が減りに指されなかった」ことを示している。

この表より、ゲームの序盤ではBLOCKの実現確率が非常に高いが、終盤になるにつれ、Allに近づいていくことがわかる。これは駒の自由度よりも地の要素が重要になってくるためであると思われる。

#### 4. 評価

単純な深さ打ち切り反復深化を行うプログラム(ID)と実現確率探索を行うプログラム(RP)を対戦させ、実現確率探索を評価した。

これらのプログラムにはECC Amazonの評価関数を用いた。この評価関数は、Min-distance Heuristic<sup>2)</sup>をベースとして、相手のプレイヤーによる邪魔も考慮した上で各プレイヤーの利きのあるマス調べ、評価値を求める方法を用いている。

どちらのプログラムも $\alpha\beta$ 法によって探索を行う。また、前回の反復の結果の評価値によるmove reorderingも行っている。

複数のカテゴリに含まれる手が存在した場合には、実現確率が最大となるカテゴリに含まれている手とし

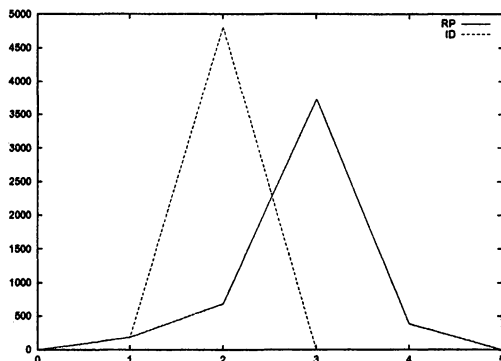


図3 探索ノード数の比較

て扱う。

RPについては水平線効果を回避するため、激指で用いている、評価値を更新した手の再探索を行っている<sup>4)</sup>。ここでは再探索する際の、実現確率は元の実現確率の10倍を用いている。

#### 4.1 実験

RPとIDの二つを実装し、毎回ハッシュに新しく5000個の末端ノード登録されたら読みを打ち切るという条件で、先手と後手を入れ替えて、ランダムに生成した初期局面50個について対戦を行った。

結果は、RP先手で36勝14敗、RP後手で44勝6敗といずれもRPが大きく勝ち越し、Amazonsにおいて実現確率を用いた探索が単純な深さ打ち切りの反復深化に比べて優秀であることを示すとともに、本論文で提案したカテゴリの有効性を示すことができた。

ゲームの中盤のある局面における、RPとIDそれぞれの探索ノード数を図3に示す。激指<sup>4)</sup>や(T-T)<sup>3)</sup>と同様に実現確率探索の特徴的な分布になっており、IDより効率的に探索を行っているのがわかる。

一方で、ゲームの終盤においてはIDの方がRPよりも深く探索する場面が見られた。これは本研究で用いたカテゴリが駒の自由度の変化に注目したものであるため、終盤では有効に働かないことが原因として考えられる。

#### 5. 結論

Amazonsで実現確率探索を行うための、カテゴリを提案した。それぞれにカテゴリについて、GGSより得た棋譜を実現確率を計算した。この実現確率を用いたプログラムと単純な反復深化をするプログラム対戦させた。その結果実現確率を用いたプログラムが大幅に勝ち越し、Amazonsでの実現確率の有効性と、提

表 1 各カテゴリの実現確率 (単位は %)

Categories	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	64-71	72-79	80-87	88-95
All	0.082	0.121	0.201	0.349	0.641	1.139	1.770	2.208	2.778	3.773	7.134	28.571
BLOCK01	0.123	0.171	0.301	0.581	1.304	3.078	6.512	4.000	2.786	3.659	2.941	10.000
BLOCK02	0.176	0.205	0.327	0.626	1.257	3.008	5.398	0.000	0.000	0.000	n/a	n/a
BLOCK03	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK04	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK10	0.123	0.171	0.301	0.581	1.304	3.078	6.512	4.000	2.786	3.659	2.941	10.000
BLOCK11	0.123	0.171	0.301	0.581	1.304	3.078	6.512	4.000	2.786	3.659	2.941	10.000
BLOCK12	0.176	0.205	0.327	0.626	1.257	3.008	5.398	0.000	0.000	0.000	n/a	n/a
BLOCK13	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK14	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK20	0.176	0.205	0.327	0.626	1.257	3.008	5.398	0.000	0.000	0.000	n/a	n/a
BLOCK21	0.176	0.205	0.327	0.626	1.257	3.008	5.398	0.000	0.000	0.000	n/a	n/a
BLOCK22	0.176	0.205	0.327	0.626	1.257	3.008	5.398	0.000	0.000	0.000	n/a	n/a
BLOCK23	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK24	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK30	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK31	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK32	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK33	0.447	0.234	0.368	0.439	1.266	3.604	0.000	n/a	n/a	n/a	n/a	n/a
BLOCK34	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK40	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK41	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK42	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK43	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
BLOCK44	0.837	0.181	0.000	0.000	0.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
APPROACH	0.246	0.336	0.541	0.978	1.972	4.080	6.979	3.015	3.286	4.106	1.923	16.667
IMOBILIZE	6.250	5.422	5.611	5.008	5.254	6.973	9.156	3.315	15.000	0.000	n/a	100.000

案したカテゴリの有効性を示した。

の実現確率に基づくゲーム木探索アルゴリズム。  
*In Game Programming Workshop*, pp. 17-24,  
 2001.

## 6. 今後の課題

終盤では ID の方が RP よりも深く読むことがあった。これが望ましいことなのかを調べるとともに、新しいカテゴリの提案も含め、Amazons の終盤で実現確率探索が有効であるか調べたい。

また、Amazons の一手は駒を動かす部分と矢を放つ部分とに分かれるが、それぞれについての実現確率を求め、探索する方法も試す価値があると考えられる。

## 参 考 文 献

- 1) Michael Buro and Igor Durdanovic. An overview of neci's generic game server. In *Game Programming Workshop*, pp. 72-77, 2001.
- 2) Theodore Tegos. Shooting the last arrow. Ms thesis, University of Alberta, Edmonton, Canada, 2002.
- 3) 橋本剛, 長嶋淳, 作田誠, Jos Uiterwijk, 飯田弘之. 実現確率探索のゲーム全般への応用 - lines of action を題材にして -. In *Game Programming Workshop*, pp. 81-86, 2002.
- 4) 鶴岡慶雅, 横山大作, 丸山孝志, 近山隆. 局面