

Amazons と Knight-Amazons の対戦プログラムの検討

加藤 光† 高谷 真弓† 山村 明弘†

†秋田大学工学資源学研究科情報工学専攻

1 はじめに

Amazons[1] は二人対戦ゲームの一つであり、また Knight-Amazons は Amazons のルールの一部を変更したオリジナルの対戦ゲームである [2]. Amazons ではいくつかの評価関数が考案されており [3], それら評価関数とモンテカルロ木探索を組み合わせた対戦プログラムの研究が行われている [4]. また, Amazons において評価関数を適用していない単純なモンテカルロ木探索プログラムと Alpha-Beta 法プログラムを対戦させる実験がされており, 10×10 の盤を用いる Amazons では単純なモンテカルロ木探索プログラムは Alpha-Beta 法に比べて勝率が低いということが分かっている [5].

本論文では 5×5, 7×7, 9×9 の盤を用いた Amazons および Knight-Amazons において, 単純なモンテカルロ木探索プログラムと Alpha-Beta 法を対戦させる実験を行う. これら実験結果より, Knight-Amazons におけるモンテカルロ木探索は Amazons よりも勝率が高くなること, および Knight-Amazons において有効な評価関数は盤の大きさにより異なることを示す.

2 Amazons と Knight-Amazons

Amazons および Knight-Amazons は 5×5, 7×7, 9×9 のいずれかの大きさのチェスボード上の指定の位置に白・黒の駒を3個ずつ配置したものを使用してゲームを行う. 先手プレイヤーは任意の白の駒を一つ選択し, Amazons ならばチェスの Queen, Knight-Amazons の場合はチェスの Knight の移動範囲に従って他の駒が配置されていない任意のセルに移動させる. その後, 移動した地点からチェスの Queen(Knight) の移動範囲となる他の駒が配置されていないセルを一つ選び, そのセルを不可侵セルとする. 不可侵となったセルは以降駒を動かしたり, 選択することが出来ないセルとなる. 後手プレイヤーは同様に黒のナイトを一つ選択し, 空いているセルに移動させ, 同様にセルを一つ選び, そのセルを不可侵のセルとする. このような着手を二人のプレイヤーは交互に行い, 自分の手番中に駒を動かさなくなったプレイヤーが敗者となる. 図1に5×5の

ゲームにおける, Knight-Amazons の先手・後手プレイヤーの着手の例を示す.

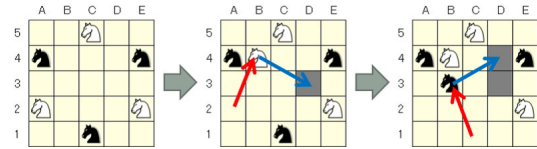


図1: Knight-Amazons のゲーム例

3 Alpha-Beta 法

Alpha-Beta 法 [6] は, 与えられたゲーム局面を Min-Max 原理に従って全探索し, 各局面に評価関数を適用することで評価値が高くなるような着手を選択する古典的な探索アルゴリズムである. 評価関数とは与えられたゲームの局面において手番プレイヤーがどれだけ有利な状態であるかを定量的な数値として返すヒューリスティックな関数であり, Alpha-Beta 法のプログラムの強さは評価関数によって決定される. Amazons 及び Knight-Amazons においては以下の3種類のものが考案されており, 本実験ではこれら3種類の評価関数をそれぞれ実装する.

- **Mobility(M)** 与えられたゲーム局面におけるターンプレイヤーの合法手数から相手プレイヤーの合法手数を引いた値を評価値とする.
- **Territory(T)** 自分の任意の駒が相手の駒よりも少ない手数で到達できる空白セルをプレイヤーの「領土」と定義したとき, 与えられたゲーム局面におけるターンプレイヤーの領土の数から相手プレイヤーの領土の数を引いた値を評価値とする.
- **Relative Territory(RT)** 各空白セルごとに, 自分の任意の駒が到達するまでの手数から相手の駒が到達するまでの手数の差分を求め, 全ての空白セルにおける手数の差分の合計を評価値とする.

4 モンテカルロ木探索

モンテカルロ木探索プログラムとは, Playout と呼ばれるランダムなシミュレーションを繰り返し, 勝率が最も高くなるような着手を選択する確率的アルゴリズム

Game playing program of the Amazons and the Knight-Amazons
†Hikari Kato †Mayumi TAKAYATaro †Akihiro YAMAMURA
†Department of Computer Science and Engineering, Akita University

ムである。モンテカルロ木探索プログラムは評価関数を用いる必要が無く、また Playout の回数を増やすことで探索の精度が高まり、より強力なプログラムとなるのが特徴である。本実験では代表的なモンテカルロ木探索プログラムである UCT[7] を実装する。UCT プログラムに設定する定数 C は全て 1 とする。

5 対戦実験

Amazons と Knight-Amazons それぞれのルールにおいて、5×5,7×7,9×9の大きさの盤ごとに UCT プログラムと Alpha-Beta 法プログラムを対戦させる実験を行う。プレイアウト回数が 10000, 30000, 50000 の UCT プログラムと、探索深さが 2、上述の 3 種類の Alpha-Beta 法プログラムを、各組合せ (全 9 通り) で UCT を先攻・後攻として 100 回ずつの対戦を行い、各対戦条件における UCT プログラムの Alpha-Beta 法に対する勝率を記録する。

対戦組合せごとの UCT プログラムの勝率を図 2 と 3 に示す。まず、Amazons における UCT プログラムの勝率に注目すると、いずれの大きさのゲームにおいても Mobility 評価関数に対する勝率が最も高く、Territory 評価関数に対する勝率が最も低い結果となっている。特に、9×9 の Amazons においては Mobility に対して 90% の勝率を持つのに対し、Territory に対しては最大でも 20% 程度しか勝利しておらず、対戦を行う相手の評価関数によって UCT の勝率に顕著な差が出る結果となった。

一方、Knight-Amazons における UCT プログラムの勝率に注目すると、5×5 および 7×7 のゲームでは Mobility に対する勝率が最も低く、9×9 のゲームでは Territory に対する勝率が低いという結果となり、対戦を行う盤の大きさによって有効な評価関数が異なるということが判明した。また、Amazons と比べると評価関数ごとの勝率の差異が小さく、Knight-Amazons における UCT プログラムの勝率は全体的に Amazons と比べて高くなることが分かった。

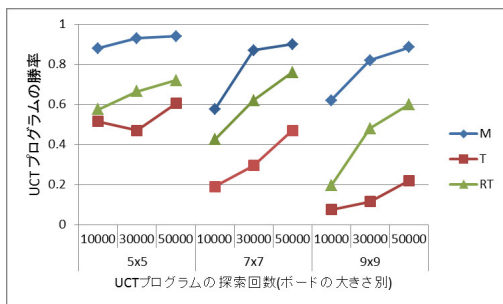


図 2: Amazons における UCT プログラムの勝率

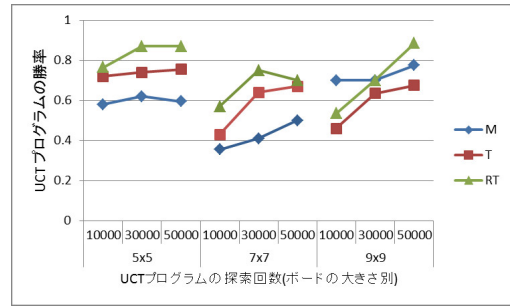


図 3: Knight-Amazons における UCT プログラムの勝率

6 まとめ

本論文では、Knight-Amazons における UCT プログラムは Amazons よりも勝率が高くなること、Knight-Amazons では盤の大きさにより有効な評価関数が異なることを示した。今後の課題として、Knight-Amazons における評価関数の適用度に関してさらに調べる必要がある。

参考文献

- [1] Zamkauskas, W.: Amazons, <http://www.chessvariants.org/other.dir/amazons.html>
- [2] Kato, H, Takaya, M., Yamamura, A: Analysis of a Monte Carlo Tree Search in Knight-Amazons, *Procedia Computer Science*, 62 pp. 31–38 (2015)
- [3] Hensgens, P.: A Knowledge-based Approach of the Game of Amazons. Master’s thesis, Maastricht University (2001)
- [4] Kloetzer, J.: Monte-Carlo Techniques: Application to Monte Carlo tree search and Amazon. <http://hdl.handle.net/10119/8867> (2011)
- [5] Kato, H, Fazekas, S., Takaya, M., Yamamura, A: Comparative Study of Monte-Carlo Tree Search and Alpha-Beta Pruning in Amazons. *Information and Communication Technology Vol 9357*, pp. 139–148 (2015)
- [6] Knuth, D., Moore, R.: An Analysis of alpha-beta pruning. *Artificial Intelligence Vol 6*, pp. 293–326 (1975)
- [7] Kocsis, L., Szepesvári, C.: Bandit based Monte Carlo planning, *17th European Conf. on Machine Learning (ECML 2006)*, pp. 282–293 (2006)