2N-01

8x8 リバーシの解法 II
- ゲームにおける探索空間の定義法の提案 -

石川 大介
神奈川大学 理学研究科情報科学専攻

1 はじめに

近年のコンピュータリバーシ(オセロ)の研究において[1]、強さという意味での研究はほぼlogistelloに集約されている。しかし、必勝法(解)を求めるという視点でリバーシ問題を扱う場合は異なったアプローチが必要であろう。なぜなら、簡単に言えば、強さの研究は必要な筋だけを読めば良いのに対し、必勝法を見つける場合は全ての局面を勝ちか負けかを証明せねばならないからである。

リバーシ問題を必勝法を求めるために扱おうとすると、必要な基礎的な課題が山積みであることに気づく。そこで本稿では探索空間に着目して、どのように扱うかの定義方法の提案をした。

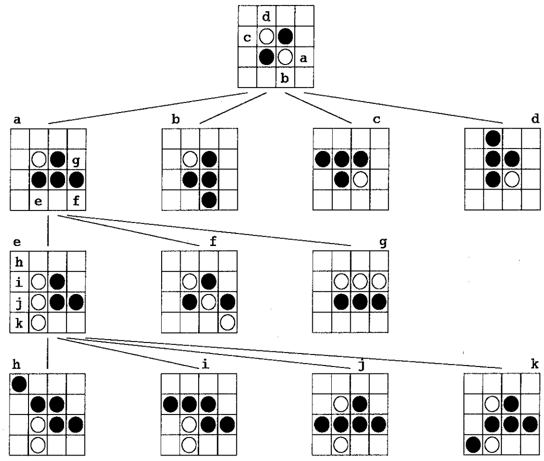

図 1: 4x4 リバーシのゲーム木の一部

2 背景

まず、ゲーム木とは、あるゲームの開始局面から終了までを、木構造で表したものである。ここで、木構造として扱うのは探索時のことを考慮して簡略化するためであり、厳密にゲームの状態推移を構造化するとグラフ構造(リバーシではlattice)になる。ゲーム木の例として4x4リバーシのゲーム木の一部を、図1に示した。

あるゲームの大きさ(複雑さ)を表す指標が2種類ある[2]。ゲーム木の大きさ(game tree complexity)と、状態数(state space complexity)という考えである。また、バックトラック問題の探索空間の大きさの予測法[3]といった研究もなされてきた。

しかし、これらから算出される値はあくまで大きさの指標であり、あるゲームの探索空間を大きさだけでなく、形として捉えるには別の指標が必要となる。

3 ゲームにおける探索空間の定義

一般的に、ゲーム情報学においてゲームの空間は前述のゲーム木として扱われる。しかし、置換表を用いて擬似的にグラフとして実装している。そこで、ゲームにおける探索空間をゲーム空間と呼び、以下の2種類に分類して扱うことにする。

Consideration on how to solve Reversi(Othello) II, -Definition of searching space on games-
Daisuke ISHIKAWA
e-mail: dais@comp.reversi.net
Department of Information Science, Kanagawa University

- ゲーム木空間：ゲーム木を3次元空間に見立てた空間。
- ゲームグラフ空間：ゲーム木空間から、重複節点を取り除いた空間。

ここで、ゲーム木空間を以下のように定義する。

ある深さ (d) における局面の総数を N_d とする。この N_d と比例するような面積を持つ円の平面 $S(d)$ を

$$\text{平面} \quad S(d) \quad = \quad \pi R_d^2$$

$$\text{半径} \quad R_d \quad = \quad \sqrt{\frac{N_d}{\pi}}$$

と定義する。D を末端局面までの深さとするとゲームの全節点の総数は

$$\text{体積} \quad V \quad = \quad \int_0^D S(d)\Delta d$$

$$(V \quad = \quad \sum_{d=0}^{D} N_d)$$

と表せる。

4 4x4 リバーシのゲーム空間

この定義に従って、ゲーム空間を作成する。ここで、全ての節点を考慮するために、規模の小さい4x4リバーシを例に取ることにする。

まず、4x4リバーシの初期局面を図1 のaにセットし、そこから末端局面まで全ての局面を探索した。探索節点の数と深さの関係を図2に示す。

前節の定義に基づいてゲーム空間を3次元グラフで図3 に示した。ただし、ゲーム木空間は破線で、ゲームグラフ空間は実線である。

5 まとめ

この図3 から、4x4リバーシ空間は玉ねぎのような形をしていることが分かり、NxNリバーシはこの相似形と見なせよう。しかし、この計算量は、実にミニマックス法による探索と同等の探索量が必要だったため、効率的な形の算出方法というのが今後の課題である。

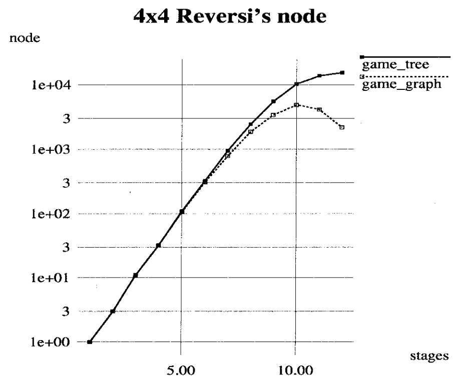

図 2: 探索節点数と深さ

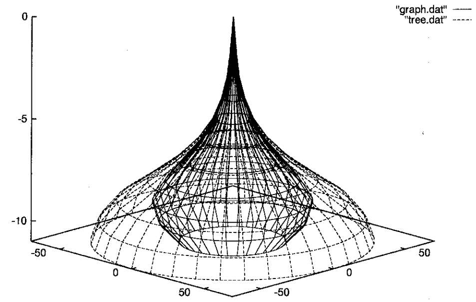

図 3: 3次元グラフによるゲーム空間

参考文献

[1] 石川大介：「コンピュータリバーシの現状」第一回ゲームプログラミング研究とリバーシ、bit, Vol32, No.5, pp.59-64, 共立出版, 2000

[2] Allis, V.: Searching for Solutions in Games and Artificial Intelligence, Ph.D thesis, Univ. of Limburg, The Netherlands, 1994

[3] Knuth, D.E.: Estimating the Efficiency of Backtrack Program, Mathematics of Computation, Vol.29, No.129, pp.121-136, 1975