オセロの解析アルゴリズムを考える

東京都立小石川中等教育学校　5年　渡辺　朝飛

1.動機

最近、オセロ AI についての話を耳に挟む。8×8 の盤面しかないのなら必勝法、あるいはどっちが勝つことができるか、を解析できるのではないかと思った。オセロは二人零和有限確定完全情報ゲームであり解析が可能である。しかし、6×6 のオセロは既に解析されているが、8×8 のオセロは未だに解析されていないので、盤面の大きさを減らしたら解析できるのではないかと思い、この研究をすることにした。今回は C++ を使って解析していく。

2.仕様

オセロのルールは一般的なものであり、仕様を以下のように設定した。
オセロには黒と白があり、黒が先手で白が後手である。オセロの縦と横の長さをそれぞれ $H,W\ (\geq 2)$ とする。オセロは $N\times N$ のグリッド状であり、左上を $(1,1)$、右下を (H,W) とする。初期位置は $(\left\lfloor\frac{2}{\mathrm{H}}\right\rfloor,\ \left\lfloor\frac{2}{\mathrm{W}}\right\rfloor)$ と $(\left\lfloor\frac{2}{\mathrm{H}}\right\rfloor+1,\ \left\lfloor\frac{2}{\mathrm{W}}\right\rfloor+1)$ を白、$(\left\lfloor\frac{2}{\mathrm{H}}\right\rfloor+1,\ \left\lfloor\frac{2}{\mathrm{W}}\right\rfloor)$ と $(\left\lfloor\frac{2}{\mathrm{H}}\right\rfloor,\ \left\lfloor\frac{2}{\mathrm{W}}\right\rfloor+1)$ を黒、それ以外は何も置かれていない状態とする。また、引き分けは黒（先手）の勝利とする。

3.研究方法

深さ優先探索（DFS）を使って解析をした。使用した言語は C++ で、工夫した点は以下の通りである。
1. プログラムを読みやすくした
2. 実装を簡潔にした

4.研究結果

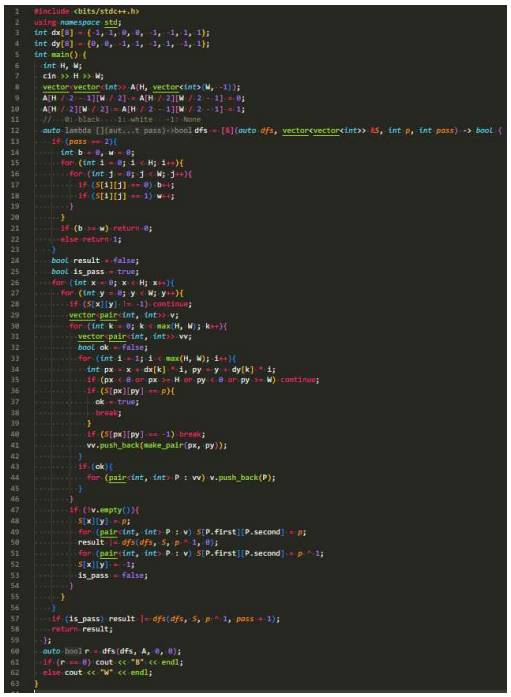
実装した結果右のようなコードになった。愚直だが、オセロの探索の実装は簡潔にできたと思う。例えばグリッドの 8 方向の探索をするとき、事前に 8 方向のベクトルを持ってループを回した。

```
int dx[8] = {-1, 1, 0, 0, -1, -1, 1, 1};
int dy[8] = {0, 0, -1, 1, -1, 1, -1, 1};
```

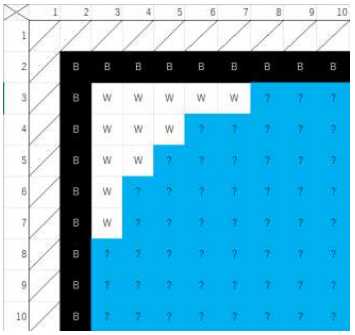
また、探索の時オセロの盤面を前の状態に戻すとき、キューを使って分かりやすく実装した。

```
if (!v.empty()){
  S[x][y] = p;
  for (pair<int, int> P : v) S[P.first][P.second] = p;
  result |= dfs(dfs, S, p ^ 1, 0);
  for (pair<int, int> P : v) S[P.first][P.second] = p ^ 1;
  S[x][y] = -1;
  is_pass = false;
}
```

他には、盤面の状態を 0 が黒、1 が白、-1 が何も置かれてないとし、bit 演算を可能にした。



実行した結果、以下の表のようになった。縦と横の数字がそれぞれオセロの大きさに対応している。B が 黒（先手の勝ち）、W が 白（後手の勝ち）、？が計算時間が 1 日を超えた部分である。



結果として、H×W がせいぜい 20 ぐらいまでは 1 分以内に結果が出た。5×5 ですら 1 日かけても結果が出せなかった。

5.考察

H, W のどちらかが 2 のときは必ず黒（先手）が勝っているように見える。これは盤面を実際に考えてみると打てる手は毎回 1 通りしかなく、必ず引き分けになることが証明できる。次にそれ以外の時。計算できた範囲では H,W が どちらも 2 ではないときは必ず白（後手）が勝っている。つまり引き分けではなく最善手を尽くせば必ず後手が勝つということだ。計算できなかった範囲も図から予測できるようにもしかしたら全て後手が勝つのではないかよと予測できる。

6.結論

8×8 の解析を目指していたが、現時点では 5×5 すら解析できなかった。
だが、解析の地盤となるコードが書けた。

7.今後の展望

先のコードを改良して、
Minimax法やαβ法などを導入して更なる高速化を目指していきたい。