

『海底』アピール文書

迫田真太郎

平成 30 年 3 月 30 日

1 概要

『海底』は 2017 年 1 月から開発が始められた将棋ソフトです。ライブラリを使わずフルスクラッチで作られており、将棋ソフト開発経験を通じて作者の技術力を向上させ、採取的により強い将棋ソフトを作ること为目标としています。

去年は第 27 回世界コンピュータ将棋選手権、第 5 回電王トーナメントに参加し、結果はそれぞれ 3 勝 4 敗の 24 位で一次予選敗退、4 勝 4 敗の 26 位で予選敗退でした。今年は勝ち越しを目指して日々開発を行っています。

将棋ソフトのレートを測定しているサイトで は WCSC27 版が R:707, SDT5 版が R:970 となっており、現時点で最新のバージョンは SDT5 版に対して 9 割程度勝つことのできる棋力となっています。

2 探索部

$\alpha\beta$ 法を基本とした探索を行っています。StockFish や強豪ソフトを、開発者自身の技術力が許す範囲で部分的に模倣した形となっており、採用している枝刈りとしては

- Razoring
- Futility pruning
- Null Move Pruning
- 多重反復深化
- PVS(Principal Variation Search)

- LMR(Late Move Reduction)

が挙げられます。

また残り探索深さが 1 未満になった段階で静止探索を行っています。静止探索では最初の一度だけその局面で駒を取る手を全て生成し、以降は直前に動いた駒を取り返す手だけを生成しています。

3 評価関数

2 駒関係 (KP,PP) を特徴量とする線形評価関数を用いています。SDT5 版は floodgate の棋譜から Bonanza メソッドを用いて学習したものでしたが、そこから elmo 式の RootStrap を行うことで棋力が向上しました。

棋譜を書き出すのではなく数百局の自己対局を行い、棋力変化の検証と学習用データ生成をまとめて行ったのちその場で学習することを繰り返しているのですが、かなり早い段階で頭打ちとなってしまっています。

4 指し手生成など

基本的には Bitboard を用いて指し手生成を行っています。さらにピンや自殺手などの細かい合法性の取り扱いのため、各マスについて利きの数と、長い利きがある方向を管理しています。

指し手のオーダリングに関しては駒を取る手を先に生成することを基本とし、駒を取らない手については history を用いて浅い探索での結果が考慮されるようにしています。

5 その他

現在は Deep Learning を用いた将棋ソフト開発も並行して進めており、何らかの形で組み合わせることができないかと方法を模索しているところです。

実戦ではフレームワークとして tensorflow を用いて C++ で動かすことを想定していますが、学習においては python を用いて実装をしているため python-shogi をライブラリ登録しています。また学習部の細かい実装やおおまかなアルゴリズムは dlshogi を参考にしたためこちらもライブラリ登録しています。