

CGP アピール文章

2018/3/30
大熊 三晴

主な特徴

- 無駄に一から作成
- 非ビットボード型
- 無駄に高 NPS を目指してるけど最近この部分はさぼり気味
- 強くするらならライブラリ不使用前提でも評価関数から手を入れていくべき状態だけど無駄に探索をいじっていました。
- 局面構造体に各マスへの利きの状態を保持
- 局面構造体に評価関数の演算途中結果のうち変化の頻度が少ないものを中心に保持
- 評価関数も自力で学習(結果は駄目駄目です)
- AVX-512 命令をはじめとした拡張命令を活用
- コンピュータ将棋では一般的にはあまり使われていない機能を使用
- 今までの大会は WCSC、電王トーナメント計 5 回すべてで毎回勝率 5 割
- 今のところ結果的に昨秋からはあまり変更できていません。
- 大会までにはアピール文書を書き直すくらいに開発が進んで欲しいです。
- 一般に流布している定跡データや、一般に流布している「局面と評価値のセット」、読み筋等は使用しておりません。無駄なこだわりだとは思いますが。

• 1から作成

強さをあまり考えずに高 NPS を目指して自作したプログラムをベースとしております。
並列化手法は現在は LazySMP です。

• 非ビットボード型, 利き等を保持

非ビットボードだとビットボードに比べ遅くなる処理もありますが、複雑な情報を持てることにより速く処理できる可能性もあります。ビットボードに比べ遅い処理をうまく避けるために利きを保持したり、局面構造体の配置をビット位置を含めて工夫しております。AVX-512 でかなりの並列化が出来そうですがまだ AVX2 を使っていた部分からの置き換えくらいしか出来ておりません。

また利きの保持以外にも、演算途中のデータを保持することによりメモリアクセス待ち時に演算を回す事により高速化を狙っております。

• 評価関数

現在は手番付き KPP です。評価関数テーブルは駒割+入玉時の位置評価のみの初期値から自力で機械学習したものですが、学習の仕方が古いので精度はかなり悪いです。評価関数の拡張は考えてますが無駄に探索部を頑張ったせいで手が回っておりません。

•SIMD 等の活用

高速化のため SIMD を活用しております。SIMD は現在評価値の算出、オーダリング、構造体のコピーが主な使用箇所です。

オーダリングの一部は VPMAX 命令で次に試す手を抜き出す方式を取っております。この方式は条件分岐なく複数の手を比較できるため、挿入ソートを通常の x86 命令で行うより高速化できております。SIMD を用いたソートの使用は試していないのでこちらのほうがより高速になる可能性もあります。

また置換表や評価値テーブルのページテーブルにラージページ(Windows での言い方 Linux 用語だと Huge Page)を使用し、高速化を図っております。(1GB サイズのページテーブルも確かめてみたいけど扱える OS があるのかすら未調査)