

CGP アピール文章

2017/3/12
大熊 三晴

主な特徴

- 無駄に一から作成
- 非ビットボード型
- 高 NPS (6~10Mnps@core i7 6700K 4コア 8スレッド)
- 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
- 評価関数は現在も学習中(まだ次元下げのコードを使っており大規模学習は未実施)
- SIMD 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

•1から作成, 高 NPS

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 ですが YBWC を使用していたときのなごりで一部無駄な処理が残っています。他にも高速化手法のアイデアはありますが(おそらく2割くらいは NPS を上げられる)今年の大会にはおそらく間に合わないでしょう。

•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、複雑な情報を持つ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。

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 Bonanza + α 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中です。

•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。