Home * Engines * **Gibbon**

Gibbons at Play [2]

Gibbon,
an UCI compliant open source chess engine by Eric Marathée written in C++. Gibbon originated from Small-C in 2005, playing Massy 2006 and improving since then. While featuring some bitboards, it has no annoying magic bitboard stuff, and is a counter approach of a Fruit like programming style, full of gotos, and difficult to follow control flow due to indent style and preprocessor switches for conditional compilation. Gibbon computes a few nodes, but is an attempt to compute the right ones [1].

Table of Contents

Features

[3]

Search

- Principal Variation Search (PVS) with Iterative Deepening and Aspiration Windows
- Two-tier Transposition Table, Pawn Hash Table
- Move Ordering with Hash Move and Killer Moves
- Internal Iterative Deepening
- Enhanced Transposition Cutoff (conditional compiled)

Selectivity

- Null Move Pruning with R = 4
- Non-PV nodes pruning (Late Move Reductions Fail-Low/Fail-High History)
- Futility Pruning
- Extended Futility Pruning
- Extensions (Check Extensions, Recapture Extensions, Passed Pawn Extensions)

Quiescence Search

- One check allowed at the horizon
- Quiescence heuristic for checks winning hanging pieces
- Static Exchange Evaluation (SEE)
- One Reply Extensions in quiescence

Mate at a Glance

- Mate in one recognition for queen moves
- Heuristic to identify moves threatening mate with queen and piece

Evaluation

- Lazy Evaluation
- Mobility
- Pawn Structure considering king placement
- Passed Pawns
- Trade down bonus that encourages the winning side to trade pieces but no pawns
- Incentive to keep the queen in case of material down
- Incentive to keep at least one pawn
- Bishop Pair

Opening

- Development

Middlegame

- King Safety considering castling rights
- Outposts

Endgame

- Bishops of Opposite Colors
- Rule of the Square
- King Pawn Tropism
- King Centralization
- Draw evaluation
- Pawn Endgame

Interior Node Recognizer

- KPK
- KRPKR
- KQKQ
- Wrong color Bishop and Rook Pawn
- KNKP
- KBKP
- KRKP

Misc

- Incremental Move Generation (few bitboards)
  => Gibbon knows all legal moves, material, mobility, and check-giving squares
- Opening Book

BitScan with Reset

In serializing bitboards, Gibbon applies a bitscan with reset based on the De Bruijn multiplication approach published by Charles Leiserson, Harald Prokop and Keith H. Randall in 1998 [4]. The De Bruijn sequence chosen is even, which implies five leading zeros [5]:

```
/**
 * bitScanForward
 * @author Charles E. Leiserson
 *         Harald Prokop
 *         Keith H. Randall
 * "Using de Bruijn Sequences to Index a 1 in a Computer Word"
 * @param bb bitboard to scan
 * @precondition bb != 0
 * @return index (0..63) of least significant one bit
 */
const char xindex64[64] = {
   63,  0, 58,  1, 59, 47, 53,  2,
   60, 39, 48, 27, 54, 33, 42,  3,
   61, 51, 37, 40, 49, 18, 28, 20,
   55, 30, 34, 11, 43, 14, 22,  4,
   62, 57, 46, 52, 38, 26, 32, 41,
   50, 36, 17, 19, 29, 10, 13, 21,
   56, 45, 25, 31, 35, 16,  9, 12,
   44, 24, 15,  8, 23,  7,  6,  5
};

#define set_bit_to_0(x64,xr)     (x64) &= ~((unsigned __int64)1<<(xr))


char SEARCH::bitScanForward_Clear(unsigned __int64 &bb)
{
   char index;
   unsigned __int64  debruijn64 =(unsigned __int64)(
0x07EDD5E59A4E28C2);
   ASSERT (bb != 0);
   // Ignorer le warning de compilation
   index=xindex64[((bb & -bb) * debruijn64) >> 58];
```

```
   set_bit_to_0(bb, index);
   return index;
}
```

See also

- Mammal
- Small-C

Forum Posts

- Gibbon 2.32a : 2219 by Patrick Buchmann, CCC, April 16, 2007
- Gibbon 2.41a : 2258 by Patrick Buchmann, CCC, June 19, 2007
- Gibbon 2.42c : 2200 by Patrick Buchmann, CCC, October 01, 2007
- Gibbon 2.51a : 2249 by Patrick Buchmann, CCC, February 12, 2009
- Gibbon 2.52a : 2208 by Patrick Buchmann, CCC, March 03, 2009

External Links

Chess Engine

- gibbon home page
- Gibbon 2.57a 64-bit in CCRL 40/4
- Gibbon 2.52a in CCRL 40/40

Misc

- Gibbon from Wikipedia
- Gibbon, Wikipedia.fr (French)
- Gibbon (disambiguation) from Wikipedia

References

1. ^ gibbon home page
2. ^ Gibbons at Play - Chinese painting 15th century, Hanging scroll, ink and colors on paper, Current location: National Palace Museum, Taipei, Taiwan, Gibbons Wikipedia.de (German)
3. ^ Features based on the gibbon home page
4. ^ Charles E. Leiserson, Harald Prokop and Keith H. Randall (**1998**). *Using de Bruijn Sequences to Index a 1 in a Computer Word*, pdf
5. ^ gibbon home page, Gibbon 2.57.a - uci - Cchess_2.h, Cchess_4.h, maj.cpp

What links here?

Page	Date Edited
BitScan	Sep 10, 2017
Engines	Mar 10, 2018
Eric Marathée	Dec 27, 2014
Gibbon	Dec 23, 2014
Massy 2006	Jan 29, 2013
Small-C	Jan 31, 2013

Up one level