GNU Chess

for version 6.1.2, 29 July 2014

http://www.gnu.org/software/chess

This manual is for GNU Chess (version 6.1.2, 29 July 2014), which is a complete chess program, frequently used as a chess engine.

Table of Contents

1 Overview

GNU Chess (`http://www.gnu.org/software/chess/`) is a computer program for playing chess. It can be used to interactively play chess on a text terminal, but it is more often used in conjunction with a GUI program such as GNU XBoard.

Because it is protected by the GNU General Public License, users are free (in perpetuity) to share and change it.

The main author of GNU Chess version 6 is Fabien Letouzey, The original author is Stuart Cracraft.

2 Contact info

We are the GNU Chess developers and you may reach us at:

`bug-gnu-chess@gnu.org`

Our official web page is:

`http://www.gnu.org/software/chess`

We are indebted to our sponsor, the Free Software Foundation whose web page is:

`http://www.fsf.org`

and which also serves as our software depository for new versions of GNU and GNU Chess.

You can download the latest version from GNU's FTP site at:

`ftp://ftp.gnu.org/gnu/chess/`

The code is provided for the purpose of encouraging you to do the programming. If you lack the programming skills to do so, try dabbling in it. You might surprise yourself.

If you want to report a possible bug in GNU Chess, please send a message to the e-mail address indicated above, providing precise information about the conditions that led to the possible bug. As a general guideline, you can kindly include the follow information:

- Version of the program. The following command will print it:

  ```
  gnuchess --version
  ```

- How you started the program. Whether running standalone or in combination with other programs such as XBoard.
- The actions you performed and the output or behaviour you observed.
- Output files if any. In order to enable the writing to adapter.log, set the following option in config file gnuchess.ini:

  `Log = true`

- Whether the problem is systematic (it always happens) or occasional.
- Any other information you may deem relevant.

3 Running gnuchess

3.1 Invoking gnuchess

The format for running the `gnuchess` program is:

```
gnuchess option ...
```

With no options, `gnuchess` starts in interactive mode and it is ready to start a chess game.

`gnuchess` supports the following options:

`--help`
`-h` Print an informative help message on standard output and exit successfully.

`--version`
`-v` Print the version number and licensing information of Hello on standard output and then exit successfully.

`--quiet`

`--silent`
`-q` Make the program silent on startup.

`--xboard`
`-x` Start the program in xboard mode, i.e. as an xboard engine. This is typically used for using the program as backend of other chess GUI such as XBoard.

Option `xboard`is accepted without leading dashes for backward compatibility.

`--post`
`-p` Start up showing thinking.

Option `post`is accepted without leading dashes for backward compatibility.

`--easy`
`-e` Disable thinking. in opponent's time. By default, the program runs in hard mode, i.e. it thinks opponent's time to think too.

`--manual`
`-m` Enable manual mode.

`--uci`
`-u` Enable UCI protocol (externally behave as UCI engine).

`--memory size`
`-M size` Specify memory usage in MB for hashtable.

`--addbook filename`
`-a filename`
Compile book.bin from pgn book 'filename' and quits.

`--graphic`
`-g` Enable graphic mode based on Unicode chess symbols.

3.2 Interactive game

Moves are accepted either in standard algebraic notation (SAN) or in coordinate algebraic notation. Examples:

```
Nf3
g1f3
O-O
e1g1
```

3.3 Command list

`gnuchess` supports the following commands:

`^C` Quit the program.

`quit` Quit the program.

`exit` In analysis mode this stops analysis, otherwise it quits the program.

`help` Produces a help blurb corresponding to this list of commands.

`usage` Produce blurb on command line options.
(Same as `gnuchess --help`)

`book` Handle the book. Requires a subcommand:
add - compiles book.bin from book.pgn
on - enables use of book
off - disables use of book
best - play best move from book
worst - play worst move from book
random - play any move from book
prefer (default) - choose a good move from book (Method subject to variation)

`version` Prints out the version of this program
(Same as `gnuchess --version`)

`pgnsave FILENAME`
Saves the game so far to the file from memory

`pgnload FILENAME`
Loads the game in the file into memory (cf. pgnreplay)

`pgnreplay FILENAME`
Loads the game in the file into memory, and enables commands first, last, next, previous. This allows replaying a saved game step by step. (cf. pgnload)

`first` Go to start position of pgn loaded game with pgnreplay.

`last` Go to last position of pgn loaded game with pgnreplay.

`next` Advances one move in pgn loaded game with pgnreplay.

`n` Advances one move in pgn loaded game with pgnreplay.

`previous` Back one move in pgn loaded game with pgnreplay.

`p` Back one move in pgn loaded game with pgnreplay.

`force`
`manual` Makes the program stop moving. You may now enter moves to reach some position in the future.

(Same as `gnuchess --manual`)

`white` Program plays black, set white to move.

Note: not implemented in this version.

`black` Program plays white, set black to move.

(White and black commands are mainly for icsDrone and will cause the current en-passant capture square to be forgotten).

Note: not implemented in this version.

`go` Computer takes whichever side is on move and begins its thinking immediately

`easy` Disables thinking on opponent's time

(Same as `gnuchess --easy`)

`hard` Enables thinking on opponent's time

`post` Arranges for verbose thinking output showing variation, score, time, depth, etc.

If pondering (see hard) is on, the program will output it's thinking whilst the opponent is thinking.

(Same as `gnuchess --post`)

`nopost` Turns off verbose thinking output

`name NAME` Lets you input your name. Also writes the log.nnn and a corresponding game.nnn file. For details please see auxillary file format sections.

`result` Mostly used by Internet Chess server.

`activate`

This command reactivates a game that has been terminated automatically due to checkmate or no more time on the clock. However, it does not alter those conditions. You would have to undo a move or two or add time to the clock with level or time in that case.

Note: not implemented in this version.

`rating COMPUTERRATING OPPONENTRATING`

Inputs the estimated rating for computer and for its opponent

`new` Sets up new game (i.e. positions in original positions)

`time` Inputs time left in game for computer in hundredths of a second. Mostly used by Internet Chess server.

`otim` Inputs time left in game for opponent in hundredths of a second. Mostly used by Internet Chess server.

`random` Randomizes play by perturbing the evaluation score slightly. The degree of perturbation is adjustable.
Note: not implemented in this version. Neither in v5

`hash` on - enables using the memory hash table to speed search
off - disables the memory hash table

`memory N` Sets the hash table to permit storage of N MB.

`null` on - enables using the null move heuristic to speed search
off - disables using the null move heuristic

`xboard` on - enables use of xboard/winboard
off - disables use of xboard/winboard
(Same as `gnuchess --xboard`)

`depth N` Sets the program to look N ply (half-moves) deep for every search it performs. If there is a checkmate or other condition that does not allow that depth, then it will not be

`level MOVES MINUTES INCREMENT`
Sets time control to be MOVES in MINUTES with each move giving an INCREMENT (in seconds, i.e. Fischer-style clock).

`load`
`epdload` Loads a position in EPD format from disk into memory.

`save`
`epdsave` Saves game position into EPD format from memory to disk.

`switch` Switches side to move
Note: not implemented in this version.

`solve FILENAME`
`solveepd FILENAME`
Solves the positions in FILENAME

`remove` Backs up two moves in game history

`undo` Backs up one move in game history

`show` Requires a subcommand:
board - displays the current board
time - displays the time settings
moves - shows all moves using one call to routine
escape - shows moves that escape from check using one call to routine
noncapture - shows non-capture moves
capture - shows capture moves
eval [or score] - shows the evaluation per piece and overall
game - shows moves in game history
pin - shows pinned pieces
Note: 'show eval' and 'show pin' not implemented in this version.

`test` Requires a subcommand:

movelist - reads in an epd file and shows legal moves for its entries

capture - reads in an epd file and shows legal captures for its entries

movegenspeed - tests speed of move generator

capturespeed - tests speed of capture move generator

eval - reads in an epd file and shows evaluation for its entries

evalspeed tests speed of the evaluator

Note: not implemented in this version.

`analyze` Switches program into analysis mode, this is primarily intended for communicating analysis to an external interface using the Xboard chess engine protocol. It enables "force", "post", and "hard", at the same time, whilst altering the output format of post to conform with the engine protocol.

`graphic` Enables graphic mode. In this mode, the board is displayed using the Unicode chess symbols (Codepoints from U+2654 to U+265F). The graphic mode requires a terminal with Unicode support.

(Same as `gnuchess --graphic`)

`nographic`

Disables graphic mode. The board is displayed using the good-old classic view based on plain ASCII.

3.4 Environment variables

If `GNUCHESS_PKGDATADIR` is defined, it will be taken as the path for the config file and for the book, in case the files are not found in the current directory. If it is not defined, they will be taken from the package data directory, in case the files are not found in the current directory.

3.5 Configuration file

3.5.1 Structure

A file called `gnuchess.ini` is used to define configuration options. The file is applicable provided `--uci` is not used. The file is looked in three places according to the following precedence:

1. The directory where the program was started
2. Environment variable`GNUCHESS_PKGDATADIR`
3. The package data directory stated at configure time

Sections are composed of `variable = value` lines.

Note: There can be spaces in variable names or values. Do not use quotes.

`[Adapter]` section

This section is used by the adapter only. The engine is unaware of these options. The list of available options is detailed below in this document.

`[Engine]` section

This section contains engine UCI options. The PolyGlot-based adapter does not understand them, but sends the information to the engine at startup (converted to UCI form). You can add any UCI option that makes sense to the engine (not just the common options about hash-table size and tablebases).

Note: use INI syntax, not UCI. For example `OwnBook = true` is correct. It will be replaced by the adapter with `setoption name OwnBook value true` at engine startup.

Standard UCI options are `Hash`, `NalimovPath`, `NalimovCache` and `OwnBook`. Hidden options like `Ponder` or `UCI_xxx` are automatic and should not be put in the INI file.

3.5.2 Options

These should be put in the `[Adapter]` section.

`Log` Default: false

Whether the adapter should log all transactions with the interface and the engine. This should be necessary only to locate problems.

`LogFile` Default: `adapter.log`

The name of the log file. Note that it is put where the program was launched from, not into the engine directory.

WARNING: Log files are not cleared between sessions, and can become very large. It is safe to remove them though.

`Resign` Default: false

Set this to "true" if you want the adapter to resign on behalf of the engine.

NOTE: Some engines display buggy scores from time to time although the best move is correct. Use this option only if you know what you are doing (e.g. you always check the final position of games).

`ResignMoves`

Default: 3

Number of consecutive moves with "resign" score (see below) before the adapter resigns for the engine. Positions with only one legal move are ignored.

`ResignScore`

Default: 600

This is the score in centipawns that will trigger resign "counting".

`ShowPonder`

Default: true

Show search information during engine pondering. Turning this off might be better for interactive use in some interfaces.

`KibitzMove`

Default: false

Whether to kibitz when playing a move.

`KibitzPV` Default: false

Whether to kibitz when the PV is changed (new iteration or new best move).

`KibitzCommand`
Default: tellall

xboard command to use for kibitzing, normally "tellall" for kibitzing or "tellothers" for whispering.

`KibitzDelay`
Default: 5

How many seconds to wait before starting kibitzing. This has an affect only if "KibitzPV" is selected, move kibitzes are always sent regardless of the delay.

`Book` Default: false

Indicates whether the adapter should use a book. This has no effect on the engine own book (which can be controlled with the UCI option `OwnBook` in the `[Engine]` section). In particular, it is possible to use both a PolyGlot book and an engine book. In that case, the engine book will be used whenever PolyGlot is out of book. Remember that PolyGlot is unaware of whether the engine is itself using a book or not.

`BookFile` Default: `book.bin`

The name of the (binary) book file. Note that PolyGlot will look for it in the directory it was launched from, not in the engine directory. Of course, full path can be used in which case the current directory does not matter.

If the file is not found in the current directory, it will be looked for in *GNUCHESS_PKGDATADIR* if the variable is defined, or in the package data directory otherwise.

Note that there is no option to control book usage. All parameters are fixed when compiling a PGN file into a binary book (see below). This is purposeful and is not likely to change.

Using a book does not require any additional memory, this can be important for memory-limited tournaments.

These are UCI options for the `[Engine]` section:

`NullMove Pruning`
(Always/Fail High/Never)

Default: Fail High

"Always" actually means the usual conditions (not in check, etc ...). "Fail High" adds the condition that the static evaluation fails high. Never use "Never" (ever)! OK you can use "Never" to test a Zugzwang problem.

I expect that this option has little effect (assuming the first two choices only). It was only added because most engines do not use the fail-high condition.

`NullMove Reduction`
(1-3 plies)

Default: 3

3 is rather aggressive, especially in the endgame. It seems better than always using 2 though.

`Verification Search`
: (Always/Endgame/Never)

  Default: Endgame

  This tries to solve some Zugzwang-related problems. It is expected to hardly have any effect in games. The default value should be sufficient for most-common Zugzwang situations.

`Verification Reduction`
: (1-6 plies)

  Default: 5

  5 guarantees that the cost of verification search is negligible in most cases. Of course it means Zugzwang problems need a lot of depth to get solved, if ever! With such a reduction, verification search is similar to Vincent Diepeveen's "double null move".

`History Pruning`
: (true/false)

  Default: true

  A bit dodgy, but fun to experiment with. It should help in blitz, but it's possible it actually hurts play in longer games.

`History Threshold`
: (percentage)

  Default: 60%

  This is the thing, as it affects the search tree! Lower values are safer, and higher values more aggressive. THIS VALUE HAS NOT BEEN TUNED! There is a good chance Fruit's strength can be improved by changing this option.

`Futility Pruning`
: (true/false)

  Default: false

  Very common but controversial. Makes the engine a tiny bit better at tactics but slightly weaker positionally. It might be beneficial by a very small amount, but has not been tested in conjunction with history pruning yet.

`Futility Margin`
: (centipawns)

  Default: 100

  This value is somewhat aggressive. It could lead to problems in the endgame. Larger values prune less but will lead to fewer positional errors.

`Delta Pruning`
: (true/false)

  Default: false

  Similar to futility pruning. Probably safer because it is used mainly during the middlegame. Has not been tested with history pruning either.

`Delta Margin`
(centipawns)

Default: 50

Same behaviour as futility margin. This one is probably safe.

`Quiescence Check Plies`
(0-2 plies)

Default: 1

Fruit tries safe (SEE >= 0) checks at the first plies of the quiescence search. 0 means no checks at all (as in most older engines). 1 is the same as previous versions of Fruit. 2 is probably not worth the extra cost. It could be interesting when solving mate problems though.

`Evaluation options`
(percentage)

Default: 100%

These options are evaluation-feature multipliers. You can modify Fruit's playing style to an extent or make Fruit weaker for instance by setting "Material" to a low value.

"Material" is obvious. It also includes the bishop-pair bonus. "Piece Activity": piece placement and mobility. "King Safety": mixed features related to the king during early phases "Pawn Structure": all pawn-only features (not passed pawns). "Passed Pawns": ... can you guess?

The following options were used in PolyGlot v1.4, but are deprecated in GNU Chess:

`EngineName`
Default: GNU Chess

This is the name that will appear in the xboard interface. It is cosmetic only. You can use different names for tweaked versions of the same engine.

If no "Engine Name" is given, the UCI name will be used.

`EngineDir`
Default: .

Full path of the directory where the engine is installed. You can use "." (without the quotes) if you know that PolyGlot will be launched in the engine directory or the engine is in the "path" and does not need any data file.

`EngineCommand`
Put here the name of the engine executable file. You can also add command-line arguments. Path searching is used and the current directory will be "EngineDir".

NOTE: Unix users are recommended to prepend "./"; this is required on some secure systems.

3.5.3 Workarounds

Work arounds are identical to options except that they should be used only when necessary. Their purpose is to try to hide problems with various software (not just engines). The default value is always correct for bug-free software.

These workarounds are unlikely to be used or meaningful for GNU Chess.

`UCIVersion`
: Default: 2

  The default value of 2 corresponds to UCI+. Use 1 to select plain UCI for engines that have problems with UCI+.

`CanPonder`
: Default: false

  The adapter now conforms to the documented UCI behaviour: the engine will be allowed to ponder only if it (the engine) declares the `Ponder` UCI option. However some engines which can actually ponder do not declare the option. This work around lets The adapter know that they can ponder.

`SyncStop`
: Default: false

  When a ponder miss occurs, the adapter interrupts the engine and immediately launches a new search. While there should be no problem with this, some engines seem confused and corrupt their search board. `SyncStop` forces the adapter to wait for the (now useless) ponder search to finish before launching the new search.

`PromoteWorkAround`
: Default: false

  Some engines do not specify a promotion piece, e.g. they send "e7e8" instead of the correct "e7e8q". This work around enables the incorrect form (and of course promotes into a queen).

3.6 Output files

GNU Chess produces several output files:

`Adapter log file`
: This file is written by the adapter. The name of the file is specified by the following option in `gnuchess.ini`:

  `LogFile = FILENAME`

  The file is produced if option `--uci` is not specified and if the following variable is set in `gnuchess.ini`:

  Typically, this file is named `adapter.log`, but any other name will do the job.

  `Log = true`

`log.nnn`
: This file is written if command `name` was requested. The contents are the opponent's name and the game in coordinate algebraic notation.

`game.nnn`
: This file is written if command `name` was requested. The contents are the opponent's name and the game in portable game notation (PGN).

`players.dat`
: This file is written if command `name` was requested. The contents are the statistics of games. This is the format of each line:

  `opponent-name wins loses draws`

`gnuchess.debug`

This file contains internal information that is useful for debugging purposes. For this file to be written, it is necessary to define preprocessor directive `DEBUG`. Hence, when installing the program, instead of

```
./configure
make
make install
```

Use the following commands:

```
./configure CPPFLAGS=-DDEBUG
make
make install
```

When analysing a bug, this file could be very helpful. Users are encouraged to provide it.

3.7 XBoard chess engine

Running the program with the "–xboard" command line parameter sets it to produce output acceptable to and accept input suitable for XBoard and WinBoard, the graphical display front-ends with mouse interface.

For historical reasons the option "xboard" does not need to be preceeded by "−", however we would encourage the new syntax.

How to run XBoard with GNU Chess as chess backend:

```
xboard -fcp 'gnuchess --xboard'

xboard -fd . -fcp './gnuchess --xboard'
```

3.8 UCI chess engine

For GNU Chess to behave as a pure UCI chess engine, execute the following command:

```
gnuchess --uci
```

In this mode, configuration file `gnuchess.ini` is ignored.

3.9 Internet

For GNU Chess to run in Internet, Zippy is required. See Zippy documentation in the XBoard/WinrBoard distribution:

`http://www.tim-mann.org/`

this is an example of how to run GNU Chess on FICS using XBoard as frontend and Zippy as connector:

```
xboard -zp -ics -icshost freechess.org -icshelper timeseal -fcp 'gnuchess --xboard'
```

4 Book

See options `Book` and `Bookfile` in Running GNU Chess - Configuration file - Options

5 Tests

GNU Chess 6 has been tested on the Free Internet Chess Server (`http://www.freechess.org`) with XBoard.

6 Auxiliary file formats

`.bin` binary book format

`.pgn` game listing like `1. e4 e5 2. Nf3` etc.

`.epd` epd-style format using FEN notation. See tests subdirectory for example.

`log.nnn` record of an entire game from computer's viewpoint (thinking, etc.)

`game.nnn` record of an entire game, similar to .pgn but auto-generated

The `.bin` file format is a simple binary format for the compiled book which is read by the program when it is using book. See the book section for more detail.

EPD and PGN require little introduction. These are the uniformly accepted standards for position recording and game recording.

Note that `log.nnn` and `game.nnn` files are written at the end of a game when you use the `name` command to give the computer your name. It is highly recommended to do this since the resulting two files that match in a monotonically-increasing extension numbered suffix may be used for reporting bugs and keeping track of your games.

7 History

The first version of GNU Chess was written by Stuart Cracraft back in 1984. Versions from 2 to 4 were written by John Stanback. Version 5 was written by Chua Kong-Sian. Version 6 was written by Fabien Letouzey.

8 Known problems

8.1 Adapter

These are known problems of PolyGlot v1.4 as described by Fabien Letouzey.

The addition of Chess960 support lead to a change in internal-move representation for castling. This slightly affected the opening-book format. We recommend that you recompile books with this version.

Several users reported engines losing on time. The playing conditions always mixed playing on an Internet server with pondering.

It is not yet clear what the source of the problem is, but there seems to be a forever incompatibility between the xboard and UCI protocol regarding a complex pondering/remaining-time relation.

9 Developers

9.1 Background of v6

Since version 5.07 of the program was released in 2003, GNU Chess has basically remained unchanged. Then Fabien Letouzey developped Fruit, which proved to be a stronger chess engine and shaked the computer chess world. At some point in time, he kindly assigned copyright to FSF, which allowed us to make Fruit 2.1 (the latest free version) the base for GNU Chess v6.

Therefore, it can be fairly said that Fabien Letouzey is remarkably the main contributor to GNU Chess v6.

9.2 Guidelines followed to develop GNU Chess v6

The idea was to use Fruit 2.1 as the base for GNU Chess v6, keeping external interfaces backwards compatible. This is particularly important since GNU Chess has been world-wide used for long.

GNU Chess can be used in two modes: interactively using the command-line interface on a text console, and used as backend engine from a graphical frontend. Both cases rely on almost identical grammar. The former is described in GNU Chess help.

When GNU Chess is used as a pure chess engine, for instance, as backend for XBoard, it uses the Chess Engine Communication Protocol (aka XBoard protocol), which is described here:

`http://home.hccnet.nl/h.g.muller/engine-intf.html`

Since Fruit uses the Universal Chess Interface (UCI), keeping the former interface was the main challenge of GNU Chess v6. UCI is described here:

`http://wbec-ridderkerk.nl/html/UCIProtocol.html`

GNU Chess can also run as a UCI chess engine, if the `--uci` command-line option is specified.

9.3 Chess engine protocol adapter

UCI is very different from the Chess Engine Communication Protocol. PolyGlot is a UCI-to-xboard adapter developed by Fabien. It connects a UCI chess engine to an xboard interface such as WinBoard. UCI2WB is another such adapter (for Windows).

Standalone PolyGlot can be used, along with Fruit, as chess engine for chess frontends, suchs as XBoard. In that case, PolyGlot and Fruit run as two independent single-threaded processes. PolyGlot starts first, and it forks Fruit. Both processes get communicated by by means of pipes: PolyGlot captures Fruit's standrad input and output.

PolyGlot tries to solve known problems with other adapters. For instance, it detects and reports draws by fifty-move rule, repetition, etc.

PolyGlot 1.4 has been adapted and incorporated to GNU Chess v6 as chess engine protocol adapter. It connects Fruit-based GNU Chess engine to the good old GNU Chess frontend.

9.4 Structure of the source code

We want to keep three loosely-coupled modules in GNU Chess v6:

- frontend
- adapter
- engine

The main program contains the frontend and starts two addtional threads, one for the adapter, and another one for the engine. The three components comprise a chain, thus there are two links:

- frontend <–> adapter
- adapter <–> engine

The links are based on pipes. There is no need for additional synchronization mechanisms such as mutex. The changes in PolyGlot and Fruit are minimal, since they were already using the same mechanism to communicate with each other.

Another technical problem was the fact that GNU Chess v5 was written in C, whereas PolyGlot and Fruit are written in C++. In GNU Chess v6 the main program, the adapter and the engine are in C++, but the frontend remains in C.

Source code was placed under a single `src` in GNU Chess v5, as usual. Three additional directories have been created for GNU Chess v6, so the code is organized in four directories:

`src`
: Contains the `main` and a source file used to create the pipes that communicate the three modules: frontend, adapter and engine.

`src/frontend`
: Contains the frontend. The code here is inherited from GNU Chess v5, with some modifications mainly in `cmd.c` and a new file `engine.c` which addresses the message passing through modules.

`src/adapter`
: Contains the chess protocol adapter, based on PolyGlot 1.4. Minor changes, wrt the baseline.

`src/engine`
: Contains the chess engine, based on Fruit 2.1. Minor changes wrt the baseline.

The sources in each `src` subdirectory are compiled as a static library. When link with the sources in top `src`, they result into the binary `gnuchess`. The module libraries are named after their respective directory name:

```
libfrontend.a
libadapter.a
libengine.a
```

9.5 C/C++ coexistence

The source code of the frontend comes mainly from version 5.07, which was written in C. There is no problem in having both C and C++ in the same program, and so have they lived together in all versions from 6.0.0 to 6.1.0. However, in version 6.1.1 and subsequent versions the frontend is compiled in C++. This change has been introduced to avoid compilation warnings in output.cc (former output.c) as a result of the introduction of Unicode literals.

PolyGlot and Fruit have a common origin, which means that there is a lot of shared code. The point is that the shared code is very similar but not identical, which would make hard an eventual unification. In order to avoid massive name clash, all the code has been wrapped in namespaces, one for the adapter, one for the engine. The frontend remains in the default namespace.

9.6 Chess engine

9.6.1 Fruit overview

Fruit was designed to help with the study of game-tree search algorithms, when applied to chess. It is now released as a chess engine, which is a somewhat different category of programs. Therefore the source code contains entire files and also functions that are either not used by the engine, or could be replaced with a much simpler (although somewhat less efficient) equivalent.

As a chess engine, Fruit combines a "robust" search algorithm with a "minimalist" evaluation function. The latter is not a design choice, and will hopefully change in the future.

The following description is only a very incomplete description. Please consult the source code for an absolute definition.

The search algorithm was designed to accommodate with heavy forward-pruning eccentricities (such as search inconsistencies).

9.6.2 Board data structure

Fruit uses the 16x12 board. Although this structure is not very popular, it can be seen as simply combining 10x12 (mailbox) with 16x8 (0x88).

0x88 was picked in Fruit because of the small memory requirements of vector calculations (much smaller tables). It is possible that Fruit uses bitboards for pawns in the future.

9.6.3 Search algorithm

The main search algorithm is a classical PVS with iterative deepening. Search enhancements such as a transposition table and null-move pruning are also used (see below).

A few details in the PVS implementation are not-so-standard and are there to supposedly enhance the stability of the search (like reducing the consequences of search inconsistencies). For example the re-search window after a scout fail high of score "value" (with value > alpha) is [alpha,beta], not [value,beta]. As another example, I only allow null move when the static evaluation fails high (i.e. eval() >= beta). Whether these features improve the strength of the engine is an open question.

The main search function is full_search() in search_full.cpp

9.6.4 Transposition table

Fruit uses 4 probes and replaces the shallowest entry. Time stamping is used so that entries from previous searches are considered available for overwriting.

Enhanced Transposition Cutoff (ETC) is also used 4 plies (and more) away from the horizon.

9.6.5 Null move

Fruit uses R=3 recursive null move, even in the endgame.

In Fruit, a precondition to using null move is that the static eval fails high. One of the consequences of this is that no two null moves can be played in a row (this is because the evaluation is symmetrical). This is a usual condition but notice that in Fruit the null-move condition is "pure" (independent of move paths). The fail-high condition was selected for other reasons however.

Also, a verification search is launched in the endgame.

9.6.6 Move ordering

The move ordering is rather basic:

- transposition-table move
- captures sorted by MVV/LVA
- promotions
- killer moves (two per level, no counters)
- history moves (piece-type/to-square table, with "aging").

9.6.7 Evaluation function

The evaluation function includes:

- Material
- Mobility
- Drawish-material heuristics
- Separated passed-pawn evaluation from the pawn hash table. Interaction with pieces can be taken into account
- Pawn-shelter penalty; with king placement this forms some sort of a simplistic king-safety feature
- Incremental move generation
- Futility and delta pruning (not tested in conjunction with history pruning and hence not activated by default)
- Move ordering (bad captures are postponed)
- History pruning (not tested seriously yet enabled by default)

9.6.8 Speed

Fruit is not fast (in nodes per second) given the little it is calculating. Some "optimisations" could be undone in order to make the code shorter and more flexible.

10 Translations

English is the native language of GNU Chess, as usual in the GNU Project. Starting on version 6.1.0, GNU Chess features internationalization support based on GNU `gettext`. The Translation Project contributes message strings translated into other languages. See the Translation Project page for GNU Chess (`http://translationproject.org/domain/gnuchess.html`) for the currently available translations.

During the installation of the operating system, the language to be used is selected. This selection is made system-wide. An individual user can select his or her own language by setting the 'locale' through environment variables. Typically, a command like this (in `bash`) will set German as the language to be used:

```
export LANGUAGE=de_DE
```

See details on this at `http://www.gnu.org/software/gettext/manual/gettext.html#Users`

Appendix A GNU Free Documentation License

Version 1.3, 3 November 2008

0. PREAMBLE

   The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

   This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

   We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

   This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

   A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

   A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

   The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

   If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

   If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

   If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

   It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

   You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

   A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

   You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

   The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

   In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

   You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

   You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

    The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See `http://www.gnu.org/copyleft/`.

    Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

    "Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

    "CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

    "Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

    An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

    The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with... Texts." line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Concept index