Djinn User Guide

by Tom Likens (Copyright © 2003-2004)

Fear only one thing, fear the Djinn.

-the WishMaster

Introduction

The Djinn[1] chess engine is an ongoing hobby of mine that I have finally decided to share with the world. It is freeware, meaning its value is likely equivalent to its cost. For programming simplicity, it is also a command-line program (i.e. It has no graphical-interface of its own), but it does support the Xboard/Winboard protocol. Since there are a number of free and commercial interfaces that support this protocol, this should pose no real problem. The program is available in both a Windows version and a Linux version. The main development occurs under Linux, but supporting both versions ensures that the code stays honest. In addition, a number of debugging tools are only available under Windows, so that is another reason to support both versions.

The Djinn engine is a modern chess program. What this has come to mean, in the nomenclature of the field, is that it utilizes most of the algorithms and techniques considered *state-of-the-art*. Currently, it lacks book learning and symmetrical multi-processing (SMP) capabilities, but most other features are present. Learning is next on the TODO hit list, while SMP may or may not ever happen, depending on my schedule and user interest.

This engine is not a Crafty clone. I have looked at the Crafty source but the code is 100% mine except for Eugene Nalimov's endgame tablebase access code, which frankly I shudder to even think about writing[2]. I am currently not releasing the source code, but that too could change. If I do it will most likely be under the GNU Public License or GPL as it is commonly known.

Features

Djinn offers a reasonable range of features that users have come to expect from a modern chess program. It is not a commercial offering but freeware, so many of the features offered by a program such as The ChessMaster are *not* available. I make no apology for this since I also have a life and full time job. A short list of Djinn's features include.

- Opening Book support
- User Configurability
- Support for EPD test suites (both standard ASCII and HTML graphical output)
- Nalimov Endgame Tablebase support
- Xboard/Winboard compatibility (Protocol Version 2+)
- Analysis (under Winboard)

---

[1] Note, Djinn is pronounced as *Jinn*. The D is silent.

[2] Eugene's code is amazing stuff and, if you ever feel like hurting your brain, you should try and understand it some afternoon when you have a bit of free time.

- Resigns when losing
- Recognizes all legal draws, including the fifty-move rule
- Negascout alpha/beta search
- Quiescence search
- Null move cutoffs
- Extended futility pruning
- Internal Iterative Deepening
- Hash tables
    - Normal transposition table
    - Pawn evaluation hash table
    - Evaluation hash table
- History Heuristic
- Killer moves

The question of playing strength always arises. My guess is that it is roughly equivalent to the last version of the old 4.x GnuChess series. Not too embarrassing, but not world champion either. It should give you a decent game and if you're not paying attention, it might even surprise you!

Command-Line Options

If you invoke the program with the *-help* option, it will display the following message. Most of these options are self-explanatory.

Note: *Command-line options take precedence over all other configuration options and settings.*

NAME
        djinn

SYNOPSIS
        djinn [-book path+file name] [-egtb path] [-help] [-ics [auto|man] [-log [on|off]]
        [-test <input.epd> <output.res> <seconds/position>] [-xboard] [-xbook] [-xegtb][-xlog]

DESCRIPTION
        djinn is a freeware chess program that runs under both Windows
        and Linux. It can be run at the command line or under any
        graphical interface that supports the Xboard protocol.

        -book <full path name of the book file>
                If an opening book is available then its location and name can be supplied on
                the command line. In addition, its location can be supplied via the program's
                configuration file or the environment variable **DJINN_OPENING_BOOK**.

        -egtb <endgame tablebase directory path>
                Sets the path to the Nalimov endgame tablebases, if they are available.
                This path can also be set via the configuration file or the environment variable
                DJINN_TB_PATH.

        -help
                Display this message.

        -ics [kibitz | whisper | quiet]
                Use this option when playing on an internet chess server. It controls the analysis
                displayed by the engine during a search. If the engine is "**kibitzing**" then any
                messages sent by the engine (normally, the current analysis of the position) will be
                shown to all observers of the game, *including the opponent*. If we are "**whispering**"
                the message will be sent only to observers and not the opponent.

        -log [on | off]
                If enabled the program will log all of its output to the file "djinnlog.xxx" where
                xxx is a number between 000 - 999. Rollover occurs if there are more than 1000
                log files present.

        -test <input.epd> <output.res> seconds/move
                Runs an EPD test in batch mode. Note, that the ".epd" postfix *is* required.

        -xboard

Enables Xboard/Winboard mode. This option has been deprecated since the engine will recognize that it is running under Winboard without it.

-xbook

Do not load the opening book. This can be used to disable the use of the opening book even if it is specified on the command-line, configuration file etc.

-xegtb

Do not use the endgame tablebases. Similar to the "-xbook" option. It will disable use of the endgame tablebases.

-xlog

Do not create a log file of the engine's activity. Note, this takes precedence over the -log option given above.

Configuration File

User preferences can be saved permanently to a configuration file. This file is read whenever the program is started. The configuration file can be named either *djinnrc* or *djinn.rc.* These names are currently hardcoded into the engine and must not be changed. On the other hand, the location of the configuration file is much more flexible. Djinn will look for its configuration file in one of three possible locations, in descending order of precedence.

1. The current working directory

2. If the environment variable **DJINN_RC_PATH** is set, it will be used

3. In the user's home directory (under Linux this is determined by the **HOME** environment variable)

Currently, the following parameters can be set via the configuration file. An example configuration file should be included in the zip file.

General Options

- Should the program's output be sent to a log file
- Path to the directory for storing all log files
- Should an opening book be used if available
- Path+name of the opening book to use
- Opening book style (solid, freq, random or first)
- Size of the main hash table
- Size of the pawn hash table
- The use of endgame tablebases can be enabled or disabled
- Path to the Nalimov tablebases
- Size of the endgame tablebases memory cache
- The point-of-view of the score (default: all scores shown are from white's perspective)
- Resignation cutoff value (in pawns)

Engine Personality Options

- King safety

Opening Book

The engine, like most chess programs, supports an opening book. The book is a binary file that is read upon program initialization. An opening book is available or, if you are enterprising, easily generated from a list of PGN[3] encoded chess games. To create an opening book you need to generate an ASCII file with the following format:

1. Lines that begin with a '#' are considered comments and ignored

2. Moves with a '?' after them are marked as blunders

3. Lines of play are listed as:


```
e4 c5 Nf3 Nc6 Bc4 e6 Nc3 Nf6 O-O Be7 d4 d5
e4 c5 Nf3 Nc6 Bc4 e6 Nc3 Nf6 O-O Be7 d4 d5
e4 c5 Nf3 Nc6 Bc4 e6 Nc3 Nf6 O-O Be7 d4 d5
Nf3 Nf6 c4 b6 g3 c5 Bg2 Bb7 O-O g6 b3 Bg7 Bb2 O-O e3 e6
Nf3 Nf6 c4 b6 g3 c5 Bg2 Bb7 O-O g6 d4 cd Qd4 Bg7 Nc3 d6
Nf3 Nf6 c4 c5 Nc3 Nc6 d4 cxd4 Nd4 e6 g3 Qb6 Nb3 d5 cd Nd5
Nf3 Nf6 c4 c5 Nc3 Nc6 g3 d5 cxd5 Nxd5 Bg2 Nc7 O-O e5 Ne1 Be6
```


Note: Leading white space is ignored when the file is parsed.

4.   Moves can also take the form:

    {e4,6000} {e5,-1201}

 The second value represents the learned value, (a pawn is equal to 1000). Currently, the learned values are not used, but this is scheduled to change in the next release. Positive scores are good for white, while negative scores are good for black. The ranges are +/-6000, which corresponds to +/- 6 pawns. Moves with a '?' appended to them are considered blunders and are <u>never</u> played, regardless of the opening book style setting. In addition, exclamation marks may also be appended to moves, to indicate that they are especially good. Djinn does not use this information in its present incarnation, but that is likely to change in future versions.

Once the ASCII version of the opening book has been prepared, the creation of the binary version is relatively straightforward. Once the engine is running, the command

build-book <ASCII-input-file> <binary-output-file>

will start the process. The engine's input parser is fairly liberal, but some moves may still not be interpreted correctly. In order to minimize the errors, the engine sets the initial position for each line in the opening book. As new moves are read, they are simultaneously played on the engine's internal board and validated for legality. If an error occurs the offending move and corresponding board position are printed to the standard output (usually, the screen or console window) and saved to the log file.

The resulting binary file should be copied to another directory for safekeeping. The *path+name* of the binary opening book can be set in one of three ways.

---

3 Portable Game Notation or PGN, is a common standard for representing chess games.

1. The environment variable **DJINN_OPENING_BOOK** can be set

2. The configuration file parameter "**opening-book=**" may be set

3. The **-book <path+name>** command-line option can be set when the program is started

Opening Book Move Selection

There are four opening book styles available to the user. This option can be set in the configuration file or changed by the user after the engine has been started.

- **solid** (*default)*- No move will be selected that was played less than 25% of the time
- **freq** - This causes the program to select a random move that is weighted towards the move played most frequently in the opening book
- **random** - Will cause the program to play a truly random move. The only exception is that moves marked as blunders will be skipped.
- **first** - The program will play the first legal move it encounters in the opening book for the current position.

Analysis Mode

The opening book can also be used when analyzing a position. For example, selecting the "**Analysis Mode**" option under Winboard from the initial position results in the following information being displayed by the program.

```
depth=0 +0.00 (e4 79% d4 11% Nf3 7% c4 3% g3 1% f4 1% Nc3 1% b4 1% g4
1% b3 1% d3 1% a3 1% a4 1% h4 1% e3 1% h3 1% c3 1% )
Nodes: 0 NPS: 0
Time: 00:00:00.01
```

Graphical Interfaces

There a number of graphical interfaces that support the Xboard/Winboard protocol. In order of decending popularity:

1. Winboard/Xboard written and maintained by Tim Mann, is the original and many would maintain, the best chess interface around. It allows for machine/machine matches, chess engine tournaments and interfacing to a variety of Internet chess servers. It is robust and used by literally thousands of people around the world everyday. There are also a multitude of web pages and forums devoted exclusively to its usage.

   http://www.tim-mann.org/xboard.html

2. Arena is free Windows-only chess interface that is gaining quickly in popularity. Its interface is slick and rivals the best commercial interfaces.   It also allows engine vs. engine matches and has too many features to list. The best advice is to download it and give it a try.

   http://www.playwitharena.com

Command Summary

Command	Description	Type
`accepted`	This command is sent back by Xboard (protocol N) to acknowledge that a feature requested by the engine is supported.	xboard
`analyze`	This xboard command can be used to analyze the current position indefinitely. This mode is similar to *force* mode except that the engine continually ponders on its next move. The following commands are supported while in analysis mode: 1. Any legal move 2. **undo** – Backup one move and start analyzing the last position 3. **new** – Reset to the starting position, continue analyzing 4. **setboard** <FEN> – Setup the new position, continue analysing 5. **exit** – Exit analysis mode 6. **bk** – Show any book moves from this position 7. **hint** – Show the predicted move from this position	both
`black`	Changes the color of the program to black. If the color is actually changed then the current, position becomes the first (i.e. non-reversible), position in the game tree.	xboard
`bk`	Use this command to show the current state of the opening book. Under xboard it displays a dialog box that displays the status of the opening book and the name of the book file.	both
`book [on|off]`	Enables or disables the program's use of its opening book.	user
`book-style <xxx>`	Determines the way the program will select a move from its opening book. The options are: • **solid** [default]– No move will be selected that was played less than 25% of the time. • **freq** – This causes the program to select a random move that is weighted towards the move played most frequently in the opening book for the current position. • **random** – Will cause the program to play a truly random move. • **first** – The program will play the first legal move it encounters in the opening book for the current position.	user

Command	Description	Type
build-book *in out*	Use this command to build a book file for the program. The ASCII input file should take the following form: 1 e4 e5 2 Nf3 Nc6 ... 1 d4 d5 2 c4 Nf6 ... The output file (which is a binary file that is not readable) can be loaded using either the **-book** command line option or the **opening-book** parameter in the configuration file.	user
computer	This is sent by Xboard whenever the opponent is another computer. It is currently ignored.	xboard
easy	Turn off pondering (permanent brain)	both
edit	This function handles the modification of the current position. It supports the following Xboard compatible commands: <u>XBOARD supported commands</u> c Change the current piece color Pa4 (e.g) Add pawn of the current color on a4 # Clear the entire board . (dot) Exit edit mode <u>Additional commands</u> xa4 (e.g.) Clear the square a4 When edit mode is entered the current color for all new pieces is always white. Edit mode does NOT alter the side that had the right to move when edit mode was invoked. After an edit command is complete, if the king and rooks are on their home squares then castling assumed to be available for them. All en'passant captures are considered illegal regardless of the current position of the pawns. The count for the *fifty-move* rule is set to zero and for *draws-by threefold-repetition* no prior positions are deemed to have occurred. When exiting the position is verified for legality.	xboard
egtb-cache <size>	This command resizes the cache used by the endgame-tablebases. As always bigger is better. egtb-cache [524288 | 1024kb | 1M] The example command sets the cachee size to 524288 bytes, 1024 kilobytes or 1 megabyte (1048576 bytes).	user

Command	Description	Type
eval [???]	Displays the evaluation for the current position. Where [???] can be: **all** - Returns the complete evaluation **kings** - King safety, opening and middlegame **kings-end** - King safety in the endgame **queens** - Queens evaluation only **rooks** - Rooks evaluation only **bishops** - Bishops evaluation only **knights** - Knights evaluation only **pawns** - Pawn evaluation only **opening** - Print the opening evaluation **middlegame** - Show the middlegame evaluation", **endgame** - Show the endgame evaluation", **square** - Rule of the square evaluation", **white-6th** - Doubled white pawns on the 6th rank **black-6th** - Doubled black pawns on the 6th rank	user
eval-test *in out*	Runs a FEN test suite. Each position is scored using only the static evaluation. In addition, each position is score four ways 1) normally 2) translated about the x-axis 3) translated about the y-axis 4) translated about both the x/y axis Any position that shows a significant variance in the four scores is flagged as a possible problem. Also see the 'test' command.	user
exit	Exit analysis mode	both
flip	Flips the position around the x-axis, so that A1 <-> A8, A2 <-> A7, ... H1 <-> H8 etc.	user
flop	Flips the position around the y-axis, so that A1 <-> H1, A2 <-> H2 etc.	user
flip-flop	Performs a A1 <-> A8 flip followed by an A8 <-> H8 transformation. The end result is that A1 is exchanged with H8, A2 becomes H7 etc.	user
force	In force mode the program does not play either side, instead it acts as a referee. It accepts moves and checks their legality, but makes no moves of its own.	both
go	Forces the program to start searching.	both
hard	This turns on pondering (permanent brain)	both

Command	Description	Type
hash <size>	This command resets the size of the main hash table. hash [524288 | 1024kb | 1M] The example command sets the hashtable size to 524288 bytes 1024 kilobytes and 1 megabyte (1048576 bytes).	user
help ["command']	If entered without any arguments, this command displays a list of all the commands. If followed by a command name, it will display information for that command only.	user
hint	This command displays a potentially good move to the opponent. It is used under Winboard to implement pondering.	both (mainly xboard)
ics "hostname"	This command lets the engine know it is interfacing to an internet chess server.	xboard
ics-seek	This command is used when the program is playing on ICC in automated mode to seek a new game.	xboard
level <m> <t> <inc>	This is the same command gnuchess uses to set the time. The parameter <m> represents the number of moves in the game, <t> is the initial time in the game. Finally, <inc> represents the Fischer-clock increment. This value is added back to the clock after each move is made when <m> moves have been made.	both
load <FEN>	Use this command to load a file containing a Forsythe-Edwards Notation (FEN) encoded chess as defined in the PGN standard. The .fen extension at the end of the file's name is optional.	user
name "Opponent"	Informs the engine of the opponent's name.	both
new	Resets the program to start a new game. All of the hash tables are cleared.	both
next	Use this command to load the next Forsythe-Edwards-Notation (FEN) encoded chess position. This command is only useful if a file containing multiple FEN positions was loaded previously.	user
noop	This command performs (NO-OP)eration. Added for Xboard/Winboard compatibility only.	xboard
otime	This command sets the opponent's internal clock in centiseconds (1/100th of a second) increments.	xboard
perf <N>	This command tests the move generator by repeatedly calling it on the current position for (N * 1000) iterations. If N is omitted then the default is 100 (i.e. the move generator is called is called 100,000 times).	user
perft <-v> [d]	Enumerates all moves to depth [d] for side from the current position. This funtion performs a mini-max search without any use of the alpha-beta enhancement and therefore will be slow if the depth is set to deep. It is used to debug the move generator. The -v option enables verbose reporting.	user

Command	Description	Type
perft-epd *d in out*	This command enumerates all the moves for a selected test suite to a depth of \<d>. The output file \<output.cnt> contains not only the total count, but also the count for each individual position.	user
phash \<size>	This command resets the size of the pawn hash table. hash [524288 | 1024kb | 1M] The example command sets the hashtable size to 524288 bytes 1024 kilobytes and 1 megabyte (1048576 bytes).	user
ping \<N>	This xboard command is used to synchronize the output of xboard (protocol version 2+) with the engine. When the engine sees this command it should respond with "pong \<N>" response.	xboard
post	Forces the program to display the principal variation while it is searching for a move. This option is on by default (except when Xboard support is enabled- then it is off).	both
protover \<2,3,..>	This message is sent after the "*xboard*" command if the version of Xboard is using the 2+ protocol.	xboard
quit/bye	Terminates the program	both
random	Included for Xboard compatability only, it is currently ignored.	none
rating xxxx yyyy	This command is sent in ICS mode. The first number is the engine's rating and the second number is the opponent's rating	xboard
redraw	Redraws the current board.	user
rejected \<FEATURE>	This command is sent back by Xboard (protocol 2+) to indicate that a feature requested by the engine is (unfortunately) NOT supported.	xboard
remove	Take back the last TWO moves played (if possible). The caveats for the '**undo**' command also apply to this command	both
resign \<m> \<n>	This command sets the resignation threshold for the program, where \<m> is multiplied by the value of a pawn to derive the threshold, (this value should be an integer). \<n> represents the number of consecutive program moves that exceed the threshold.	user
result { xxxx }	This is sent by xboard/winboard when a game is completed.	xboard
sd \<DEPTH>	Causes the program to search to a fixed depth \<DEPTH> ignoring the time required to complete the search. This command should be used cautiously since the search could take a very long time to complete depending on the depth selected and the position.	user
setboard \<FEN>	Use this command to load a Forsythe-Edwards Notation (FEN) encoded chess position at the command prompt.	both

Command	Description	Type
st <TIME/MOVE>	This command sets the seconds the program will spend searching each of its move.	user
swap <move>	This command tests the static exchange evaluation (or SEE as it is often called). This command looks at the consequences of a ***specific*** move to a square. You need to provide the move in SAN, (i.e. Rxg5, fxe4, Qxe5, f4 etc.) The command will return a +/- (or 0) score if the side moving will win/lose (or exchange even) material on the square.	user
swap-all <c> <sq>	This command tests the static exchange evaluation routine. You need to provide the color to move and the square to perform the captures on (this should be A1 through H8). The command will return a +/- (or 0) score if the side with the right to move will win/lose (or exchange even) material on the square.	user
swindle [on|off]	This command is only useful if the endgame tablebases are loaded. If the engine detects an EGTB draw at the root then in swindle mode it will select the draw move that prolongs the game the longest in the hopes that the opponent will make a mistake and convert the draw into a loss.	user
test *in out*	Runs an EPD test suite. Note, each position is scored using the normal search. Also see the '**eval-test**' command.	user
test-html *in out*	Similar to the '**test**' command. The difference is that running this command creates a directory called <output> in the current working directory. Inside this directory are HTML analysis files for each test position. Opening the file "*index.html*" provides browser access to all the positions.	user
time <n>	This command sets the program's internal clock in centiseconds (1/100th of a second) increments.	xboard
undo/takeback	Take back the last move made. If you use this command to retract one of the program's moves then you must issue the 'go' command to force the program to start playing again. Note, this command stops both clocks.	both
white	Changes the color that the program plays to white. If the program was previously playing white it is left unchanged. If the color changes then the program counter is reset so that the current position is the first position (i.e. it cannot be undone).	xboard
write-book -v file	Use this command to write out the engine's opening book in ASCII format. If you add the -v verbose qualifier then additional information is written to the output file.	user
write-html <file>	Use this command to save the current position to file <file> in HTML graphical format.	user
xboard	This message is sent by Xboard when it starts the engine for the first time	xboard

Bugs Reports, Comments and/or Suggestions

If you find a bug or would just like to drop me an email concerning the engine, my email address is: tlikens@charter.net I would also like to hear from you if you have any suggestions for new features or improvements to the program.

Acknowledgments

■ Robert Hyatt for his unrelentingly concise and clear responses to all my queries on CCC and for his chess program Crafty.

■ Eugene Nalimov for creating his endgame tablebases and giving me, and countless others, permission to use his access code.

■ The multitude of people who have patiently answered my chess programming questions on the web and in personal emails over the years.

■ To Jan. She knows why.