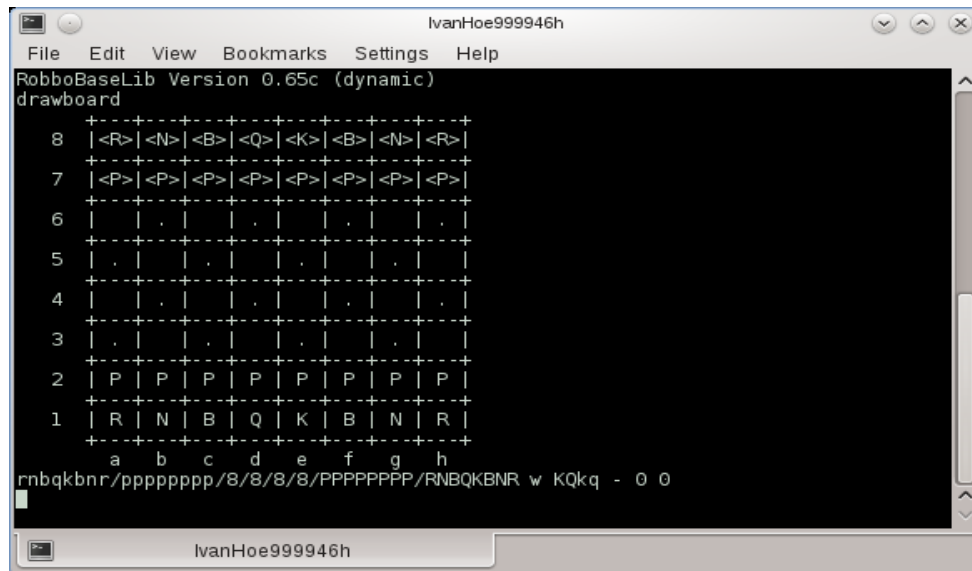


Introduction



IvanHoe is the strongest open-source chess engine available. IvanHoe is free to use and distribute. The program has no graphical interface, but rather a console application that communicates with a chess graphical user interface (GUI) such as [XBoard](#) (Linux), [SCID](#) (Linux/Windows), and [Arena](#) (Windows) via standard Universal Chess Interface (UCI) protocol.

IvanHoe has its own [endgame tablebases](#) ([RobboBases](#)) which are superior to everything out there today. The bitbase tablebases contain the game-theoretical value (win, loss, or draw) of each possible move in each possible position up to 6 pieces.

IvanHoe has matured a great deal since its inception. It began as **IPPOLIT**, a single processor chess engine, then **RobboLito** which added tablebases support, to **Igorrit**, the first multi-processor support, and today, a host of features such as multi-pv, monte-carlo, Chess960, etc. IvanHoe uses a decrementing versioning scheme; for example, 999984 is newer than 999987, and 999950k is newer than 999950h.

IvanHoe is a bitboard chess engine optimized for 64-bit architecture with native support for both 32/64 bit Linux and Windows Operating Systems. The *src* directory contains source code for the engine which is public domain, and *Linux* and *Windows* directories have binaries/executables and libraries.

Naming IvanHoe logo to the same name as the binary/executable will allow some GUIs to display it during use.



Windows users may need to install a small library (Microsoft Visual C++ Redistributable Package) free from Microsoft in order to run the program:

32-bit: <https://www.microsoft.com/download/en/details.aspx?displaylang=en&id=5555>

64-bit: <https://www.microsoft.com/download/en/details.aspx?displaylang=en&id=14632>

RobboBases

Downloading

RobboBases compressed and split with [7-Zip](#) are available at <http://chess.cygnitec.com/RobboBases>.

1. Create a directory named *RobboBases* and two sub-directories (*RobboTripleBase* and *RobboTotalBase*) under RobboBases.
2. Create nine sub-directories (2, 3, 4, 5, Z, 33, 42, 51, and 6Z) under RobboTripleBase directory.
3. Create eight sub-directories (3, 4, 5, Z, 33, 42, 51, and 6Z) under RobboTotalBase directory.
4. Download each archive to the corresponding directory.
5. To extract, right-click the archive (001 for split files), and click *Extract Here*.

Note: As of IvanHoe 999946, older RobboTripleBases are no longer supported. Newer RobboTripleBases support began with IvanHoe version 999950. The older format are supported by IvanHoe versions 999987 to 999947, and there are no 6-man sets. Remember that IvanHoe uses a decrementing versioning scheme.

RobboBases 6-man (33, 42, 51, and 6Z) sets are available at:

<http://ippolit.wikispaces.com/RobboTripleBases>

<http://ippolit.wikispaces.com/TotalBases+Download>

Loading And Using

RobboBaseLib.so (Linux) or RobboBaseLib.dll (Windows) must be in working directory of the engine. Another option is to set RobboBaseDynamicLibraryFile full path pointing to the library file when located elsewhere.

Sample path settings:

RobboTripleBaseDirectory X:\chess\tablebases\RobboBases\RobboTripleBase

RobboTotalBaseDirectory X:\chess\tablebases\RobboBases\RobboTotalBase

RobboTripleBaseDirectory /media/chess/tablebases/RobboBases/RobboTripleBase

RobboTotalBaseDirectory /media/chess/tablebases/RobboBases/RobboTotalBase

More configurations pertaining RobboBase usage are described in UCI Options section.

Arena and *ChessGUI* normally retain path pointing to the tablebases.

Fritz: This GUI will not remember the directories, so create a new engine, and with show hidden files/folders enabled, browse to

C:\Users\yourname\AppData\Roaming\ChessBase\Engines.UCI

or it might be

C:\Documents and Settings\yourname\Application Data\ChessBase\Engines.UCI

Open the UCI engine file with your favorite text editor, and modify it to mirror sample below with correct directories:

[ENGINE]

Name=IvanHoe 9.46b x64

Author=Yakov Petrovich Golyadkin, Igor Igorovich Igoronov, Roberto Pescatore, Yusuf Ralf Weisskopf, Ivan Skavinsky Skavar plus Decembrists (all)

Filename=X:\chess\engine\ivanhoe\IvanHoe999946a.exe

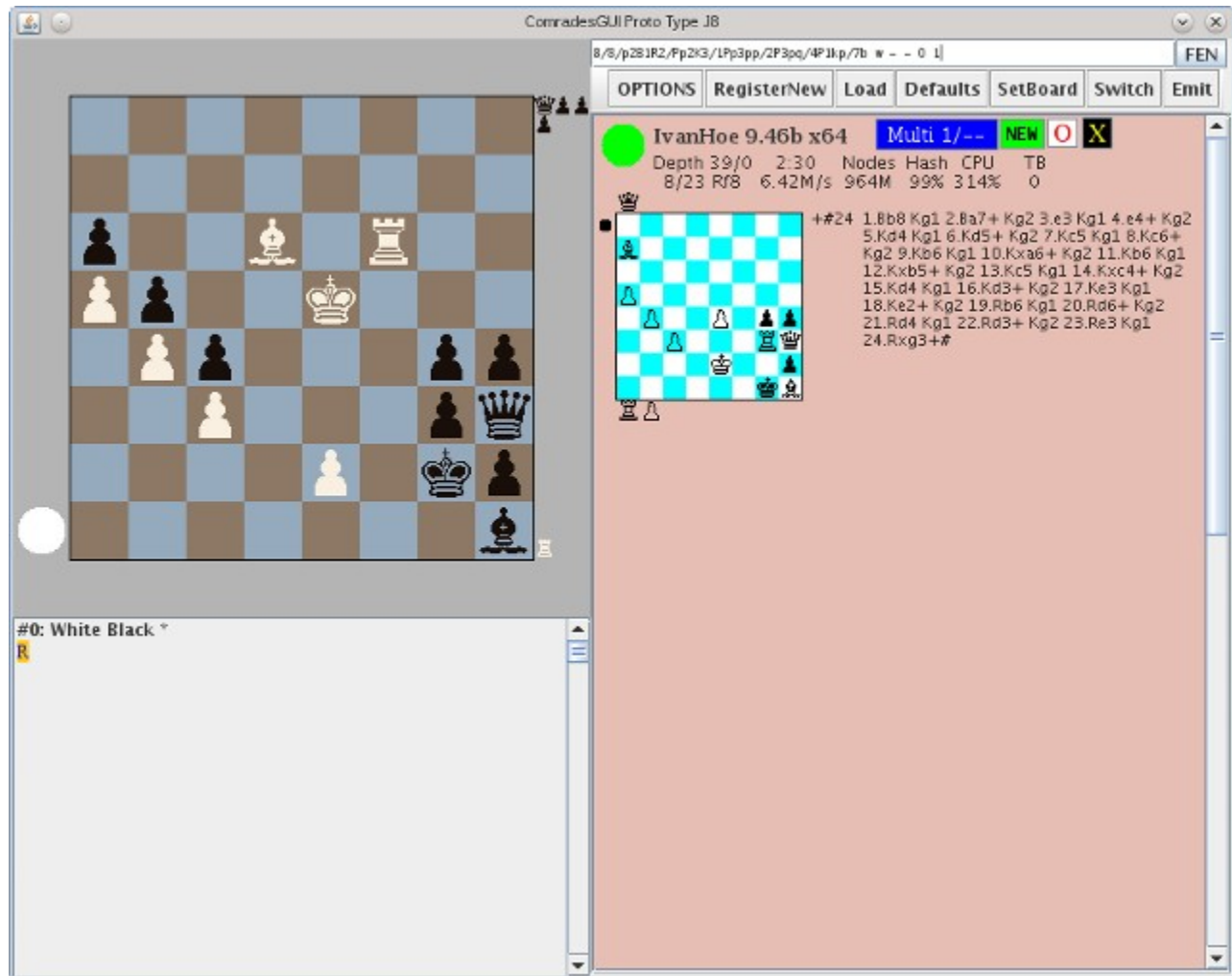
[OPTIONS]

RobboTotalBaseDirectory=X:\chess\tablebases\RobboBases\RobboTotalBase

RobboTripleBaseDirectory=X:\chess\tablebases\RobboBases\RobboTripleBase

Visit <http://ippolit.wikispaces.com/RobboBases> for more info.

ComradesGUI



Instructions for ComradesGUI (Java GUI)

ComradesGUI requires [Java Runtime Environment](#) (JRE).

Double click the jar file to load the GUI or pass **java -jar ComradesGUI.jar** argument in command line.

Proto Type.

First Value : attend IvanHoe, and too Monte Carlo. Directed at analysis (human)

Now includes Inventive Comrades Intercommunicatory (ICI) in first.

ToDo:

- Logging, onto disk

- Command Line Options on start to Communicator

- Thrive : MultiPV redundant transpose ?

- To better: attend StringBuffer over String (+)

Conditions:

Do not name two communicators the identical.

Do not change Path to new version with alternate UCI option list, make new Communicator.

Do not delete Communicator when Instance is here.

Attend JSpinner change before On/Off push in Monte Carlo. (order)

Applies:

Attend from RegisterNew (button), then Load.

Turn on/off with red/green light. Right click for Monte Carlo if so.

SetUp mode from Menu, push Yellow to exit, so too "C" to clear. See rules for SetUp for RobboExplorative (java).

Edit castles and en passant by hand in FEN box, not SetUp mode.

DoubleClick "C" for SetUp to newgame, Random for Chess960 (box).

The attire is not failsafe in current. Do not pretend stupid.

A "pv" exceeding 256 tokens will crash.

Tricks:

Hold CTRL-ALT when selecting from Defaults to direct delete.

Double click west of a8 to reverse board.

Underpromote concerning: click pawn on 7th by right button, then 3 buttons RBN.

In SetUp, double-click C (upper left) twice to NewGame, even Chess960.

MoveTree:

Click to expand. Double-Click to jump. Arrows to move (up/down/left/right). MoveTree at infancy now.

Keys (Up/Down/Left/Right/Space) from Newbuy to traverse.

Mores:

Report bugs (all) at [IPPOLIT](http://ippolit.wiki) wiki.

Confer wise ideas in there too, but refrain heavy tidings.

Note: Logging refrains at 16 Mb, desisting 4mb from top.

1. Buttons

OPTIONS - Set Options for ComradesGUI, Fonts and Colors

RegisterNew - Register new Communicator via File Chooser

Load - Load new Communicator via Registry List

Defaults - Change defaults for Communicator

SetBoard - enter set Board mode

Switch - PGN Selector, too: Load PGN file (all)

Emit - Emit PGN from all (window), attending Tree

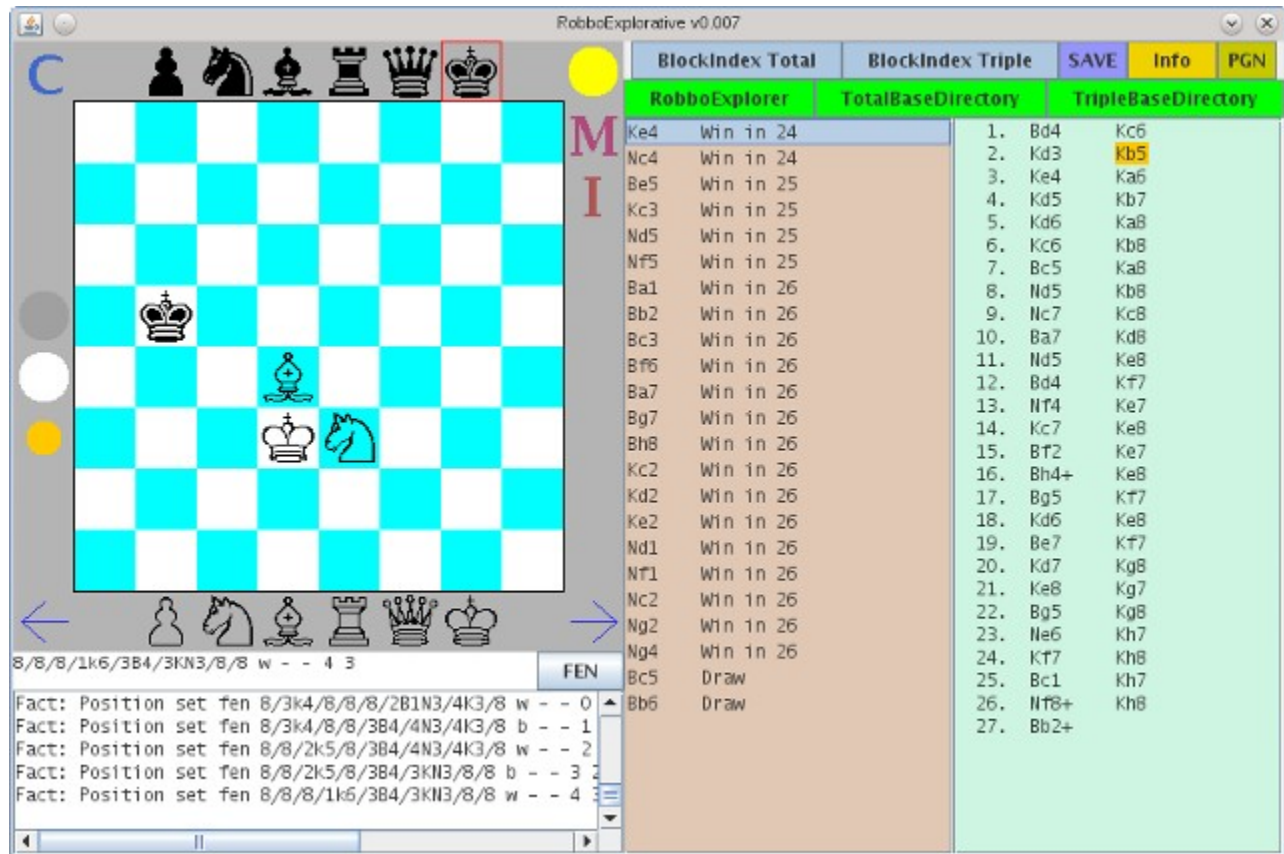
Yet: Shift-Click, only MainLine (Newbuy)

Also: Switch PGN via tabs (in tree).

CTRL-Tab, game back

Shift-Tab, game forward

RobboExplorative



Instructions for RobboExplorative (Java GUI).

RobboExplorative requires [Java Runtime Environment](#) (JRE).

Double click the jar file to load the GUI or pass **java -jar RobboExplorative.jar** argument in command line.

For then, Click upon RobboExplorer to tell (file chooser) where RobboExplorer sits.

The same holds for TotalDirectory and TripleDirectory in their base. RobboBaseLib file must also be in the same directory as the RobboExplorer binary.

These can be saved in RE.config to read upon start.

This is the example RE.config file.

```
=====
TotalDirectory /home/chess/RobboTotalBase/
TripleDirectory /home/chess/RobboTripleBase/
RobboExplorer /home/chess/IvanHoe-Beta-999950r/RobboExplorer
=====
```

Now see "java" know needs to where RE.config is too, in mandatory. The look is direct. The SAVE button does the save of RE.config for you now.

The right-click on the SAVE button brings up the InformPanel to say what files give each piece collection. This is for conformance.

Also: first loading is not immediate unless Bases are in disk cache. Apply patients. Progress notes in TextInfo box (lower left). The RobboExplorer button must turn Green to invite. See below to BlockIndex.

To use RobboExplorative click on pieces then click them on the board. We do not have drag and drop for expense.

The right-click (if so) of a piece will move it (jumping).

Example: click on WhiteQueen at bottom. Click on e5. Click on d3. Two queens now. Right-click on e5. Click on g8. WhiteQueen jumps e5 to g8.

Twice right-click is an other delete option. The first is empty square, from left of either pawn (black or white).

The White/Black indicator is on the left. The Clear board ("C") is on the upper left. The Yellow button in upper right calls RobboExplorer on data. The FEN button can also be pushed for input (validated).

So then, with TotalDirectory or TripleDirectory ready (even both) then click Yellow button with position on board. The responding data appear at right. Then too, moves are doubleclick-able or via up/down arrows plus Return (Enter). With TripleBases there is only three types but TotalBases will sort by length.

The M below the yellow button is to splice the mainline for the move list. Warning: this consumes time in instances.

The I below the M, is for information on the files for the board. They can appear in a new frame.

To export, as PGN the discussion, implore button (PGN) for the this.

The final convenience is Block indexing that accrues greater speed to load. This applies to power users and is easy to shoot your foot to Yakutsk if files find rearrangement.

The Bugs you find you can to say on [IPPOLIT](#) wiki.

Large (Huge) Pages

Windows

1. Choose **Start > Control Panel > Administrative Tools > Local Security Policy**.
2. In the left pane of the Local Security Settings window, expand **Local Policies** and choose **User Rights Assignment**.
3. In the right pane of the Local Security Settings window, choose **Lock pages in memory** and choose **Action Properties**. The Local Security Setting dialog box opens.
4. In the **Local Security Setting** dialog box, click **Add User or Group**.
5. Enter the appropriate user name, then click **OK** to close the Select Users or Groups dialog box.
6. Click **OK** to close the Local Security Policy Setting dialog box. The Local Security Settings window shows the user assigned to the policy.

Linux

1. `vm.nr.hugepages`
Attach to `/etc/sysctl.conf` one line
`vm.nr_hugepages=1024`

This demands 1024 2Mb pages with "sysctl -p" or reboot for read "man sysctl" to see.

Or in the system

`# echo 1024 > /proc/sys/vm/nr_hugepages`

Ensure the number 1024 is your proper size in desire.

The overlimit subdues the system!

2. **MEMLOCK limit**
Attach to `/etc/security/limits.conf` two lines for 20GB in the limit
`* soft memlock 20971520`
`* hard memlock 20971520`

Now read "man limits.conf" for these are per login.

The limit size can be too big and not annoy.

Check "ulimit -a" to see memlock limits in the actual.

3. **shmget standards**

Contrive with `/proc/sys/kernel/shmmax` to lift for size in the choice (20GB).

`# echo 21474836480 > /proc/sys/kernel/shmmax`

Again too big has no annoy. Purview "man shmget" NOTES in the more.

4. **Arriving in IvanHoe**
The UCI TryLargePages demands to use yours.
The problem is to do so for Hash and PawnsHash.
The size for Hash is better at 1GB pages but not for PawnsHash unless your computer is big.
The simultaneous split is not seen by us.
So the 2MB size for the Large Pages is what we see now.
Always disengage IvanHoe via quit to ensure deremedy of LargePages.Yet: the GUI has the fault too. The SIGKILL eludes the catch.
5. **Testing results (1GB+64Mb+64Mb)**
NPS 8477000 vs NPS 9167000 (8%)

Large (huge) pages can increase performance by up to 15 percent. It does require frequent reboot however.

UCI Options

This has a description with the UCI options for IvanHoe currently (999946-Beta).

Option: Hash

This takes values from 1 to 65536 in megabytes, though requires too values to regard height bits. We cannot put this information on the option itself for the various GUI insists we determine it as "Hash". Warning: upon a value who is too large, the access will be slowed.

Option: PawnsHash

This takes values from 1 to 1024 in megabytes, and only works for binary powers (e.g: 1, 2, 4, 8, 16, 32, 64, 128, ...). The name is wrong, as king location is also in pawns. Rising this can lead to more hash hits with pawn evaluation. Again making it too large will slow the access in the memory system.

Option: PawnsHashDynamic

This automatically sets PawnsHash to 1/16th of Hash.

Option: PVHash

This takes values from 1 to 1024 in megabytes, and only works for binary powers. The name is wrong, as king location is also in pawns. Increase this if TryPVinAnalysis is off and the PVs are too short.

Option: EvalHash

This takes values from 1 to 1024 in kilobytes!, and only works for binary powers. Do not apply this too large! Enough to reach large pages has sufficiency.

Personal Advise: 1024 Hash, 128 PawnsHash, 128 PVHash, 2048kb EvalHash (large).

Option: Threads

This counts the multicore CPUs. The current maximum is 64 as the SMP workings are untested for more. CPUs are automatically detected.

Option: Ponder

This demands to the UCI whether to ponder. We have tested Ponder and it now works as intended. So fast moves are played upon desire from ponderhit. They can be bad occasionally, but all is true in generally. The Easy Move logic sees our attention here, where we think are problems.

Option: VerifyNullMove

This verifies null move. The default is true as that is better. Turning this off cannot see much gain and loses in zugzwang.

Option: AlternativeTimeUsage

This turns on the AlternativeTimeUsage. We have made no thorough tests for the application here.

Option: AllowInstantMoveFromHash

This allows instant moves to be made from hash when in ponder off mode. The value of this is when a move is clear.

Option: BufferTime (milliseconds)

This lists in milliseconds how much time to borrow. This will not be used. For a 1 minute flat game, 1000 millisecond (1 second) is OK.

Option: OutputDelay (milliseconds)

This lists in milliseconds how long for waiting until output is emitted.

Option: MultiCentiPawnPV

This limits the gap of worse moves with MultiPV. For making this 100 centipawns will eliminate moves worse than that much behind.

Option: RandomCount

This turns on the randomizer effect unless it is zero. The number of random numbers to add is controlled here. The maximum is 8.

Option: RandomBits

This determines how much bits to use for each random component. The value can be 1 or 2 or 3. With one random bit, the value is -1 or 0 or 1, and with 2 it has from -3 up to 3, and with 3 it has from -7 to 7. Each random is added according to the count of RandomCount. To see this the value of RandomCount is 4 and RandomBits is 2 should add 4 values from -3 to 3 for each evaluation. We use RandomCount as 4 or 8 and RandomBits as 1 for testing.

Option: UCI_White_Bishop_Pair_Scale (cp)**Option: UCI_White_Pawn_Scale** (cp)**Option: UCI_White_Knight_Scale** (cp)**Option: UCI_White_Light_Bishop_Scale** (cp)**Option: UCI_White_Dark_Bishop_Scale** (cp)**Option: UCI_White_Rook_Scale** (cp)**Option: UCI_White_Queen_Scale** (cp)**Option: UCI_Black_Bishop_Pair_Scale** (cp)**Option: UCI_Black_Pawn_Scale** (cp)**Option: UCI_Black_Knight_Scale** (cp)**Option: UCI_Black_Light_Bishop_Scale** (cp)**Option: UCI_Black_Dark_Bishop_Scale** (cp)**Option: UCI_Black_Rook_Scale** (cp)**Option: UCI_Black_Queen_Scale** (cp)

These are user fun options with rescaling pieces. We have not applied them internally.

Option: MaterialWeighting**Option: KingSafetyWeighting****Option: PawnsWeighting****Option: StaticWeighting****Option: MobilityWeighting**

These are more user fun options for rescaling. The units are all in 1024s.

Option: AlwaysAnalyze

This option allows GUI compatibility for some and propels the companion of MultiPV mode when playing a game.

Option: TryPVInAnalysis

This option demands an expansion of the PV in analysis mode. If this is not on the PV can be truncated from hash hits.

Option: FixedAgeAnalysis

This option delimits the AGE count in the hash table when in analysis. This is useful when applying forward and backward analysis as in the contrary, the AGE is incremented upon every position from the GUI. However you need to be careful and apply "ucinewgame" to shoal the hash when the higher depths entries become boggy, or when applying an independent position.

Option: SendCurrmove

This option demands the currmove to have been sent even when game mode is on.

Option: DoHashFull

This option implies to send hashfull at the second updates. There is a little overhead.

Option: GetFEN

This utility endeavors for IvanHoe to compute the FEN as an info string in UCI.

Option: TimeImitateOpponent**Option: TimeMoreWhenLosing****Option: TimeMoreWhenWinning****Option: TimeEasyFactor****Option: TimeEasyFactorPonder****Option: TimeBattleFactor****Option: TimeOrdinaryFactor****Option: TimeAbsolutePercent****Option: TimeDesiredMillis****Option: TimeBookExitMoves**

These options control the standard time usage. Our defaults seem decent. The DesiredMillis sets up how much time to use, except in movestogo mode when that is apparent. The default 40 uses 40/1000 or 1/25 of the time back in the desired time. The factors then say how much of the desired time to use in situations. The absolute percent caps the amount that can be used in worst scenarios. The book exit moves demands to use extra time on the first moves when the book was left. We put this as 0 and the matter is not much.

Option: ExtraExtendInCheck

This option when on will extend an extra half-ply when in check. The default is off.

Option: SplitAtCUT, SplitDepthCUT, SplitDepthALL, SplitDepthPV

These options appear with -DUSER_SPLIT, and allow the user to control the multicore mode for more.

RobboBase Options:

Option: RobboBaseDynamicLibraryFile

This is for libraries .so (Linux) and .dll (Windows). It should only be modified when the library file is located elsewhere other than the engine's current working directory.

Option: RobboTripleBaseDirectory

This sets the RobboTripleBase directory to a string, and loads the TripleBases.

Option: UnloadDeregisterInform

UnLoadTriple unloads the RobboTripleBases from the directory that is set.

DeRegisterTotal deregisters the RobboTotalBases from the directory that is set.

Option: RobboTotalBaseDirectory

This sets the RobboTotalBase directory to a string, and loads the TotalBases.

Option: RobboTotalBaseCacheSize

This sets the RobboTotalBase cache size. The values are 1 to 1024 megabytes in binary. The value should be minimal for the TotalBase use finds itself only at root positions. The exception is when building the TotalBase lot, though that access is separate. So 1 megabyte seems fine for play and analyzing here.

Option: DynamicLoadTripleBaseCacheSize

This sets the size for dynamic Triple bases in cache. Unless said all with 5 pieces or more go here. The size depends on your work. With analysis of endgames, 256MB or more counts wise.

Option: TripleHashSize

This takes values from 1 to 4096 in megabytes, and only works for binary powers. It stores score for TripleBase positions probed which complements DynamicLoadTripleBaseCacheSize. It can however decrease NPS when set too high.

Option: TripleWeakProbeDepthHalfPly

This sets the depth limit to weak probes. When depth exceeds it, a weak probe is incurred. Negative values are possible too, and can have value. Increasing too large turns off TripleBases in essence. Weak probes are not handily free in cost, though worry is low generally. There is also a plus with statistics in having weak probes attend.

Option: TripleDefiniteProbeDepthHalfPly

This sets the depth limit with TripleBase probes. When depth exceeds it, a TripleBase probe is incurred. Probe from disk will attend much time. Guideline: 40 half ply sums 20 ply, at 1 millionth of tree if split binary. Note: Probe still can technically fail if cache tendency is arraigned (SMP defect).

Option: TripleDefiniteProbeHeight

This sets the height for definite TripleBase probes. When height is as small as value, definite TripleBase probe is met. The control is switched for more at root. This complements your above option.

Option: LoadOnWeakProbe

This sets whether to load TripleBases in the background during a search when a probe is made. If this is not set, your TripleBases must be in memory to function.

Option: RobboTripleBulkLoadThisDirectory**Option: RobboTripleBulkDetachThisDirectory**

This moves an entire directory of RobboTripleBases into memory (RAM). The cache is not filled by these. The 5s fill over 500MB. The directory should be minor, not root. Example is "value 5" or "value 345Z" for those. Multiple directories are separated using pipe (|). It can also be "value /media/disk/RobboTripleBase/5" or "value /media/disk/RobboTripleBase/345Z" using full path. 2, 3, and 4 are put in RAM automatically.

Option: RobboTripleBulkLoadThisName**Option: RobboTripleBulkDetachThisName**

This moves one file for RobboTripleBases to be used from RAM other than disk. There is care, as TripleBases use recurrings to call them more. The situation send a time in difficult. Unless there is necessary, the WeakProbe access deems enough.

Option: MultiPV

This sets the MultiPV number. This will work in game mode additionally if AlwaysAnalyze is made too.

Option: UCI_Chess960

This allows support for Fischer Random Chess. Some GUIs handle it like ponder.

Optional options:

DebugTimeManagement: to appear for debugging with time management

Further utilities in -DUTILITIES:

eval: lists an evaluation of the position, but still not prepared finally

benchmark [go string]: achieve a benchmark upon 16 positions, with UCI string appended seeing the default for "go movetime 1000"

perft [n]: count a perft for the position to n moves

perft-check [n] [c]: count a perft for the position to n moves and ensure the answer is c

drawboard: draws a board in the style of Crafty

verify-triple [a] [b] [c] [d]: utility internal function for the verifier with triple bases which need the loading and the TotalBase registry, and notation here is the internal accounting of pieces so [7] [14] [9] [0] is wQ against bR+bP

The ZOG MP does not turn on unless you compile it and then it hangs.

MonteCarlo:

UCI Version: "**go montecarlo** [options]"

Options:

cpus # : default has 1

min # : default has -15000

max # : default has 15000

length # : default has 65535

depth # : default has 9

moves [list]

This searches the moves with the moves [list] who comes last, churning it on [cpus #] of cpus. Each iteration runs for [length #] ply at each move searching [depth #] ply. When score exceeds [max #] or recedes [min #] the termination occurs too. The default is 9 ply searches and no termination until game end.

position fen 4r3/4b3/3p1k2/2pP4/2P5/1P5K/6R1/6N1 w - - 0 0

go montecarlo cpus 4 min -25 max 325 length 40 depth 10 moves g2a2 g2e2 g2f2 g1f3 g2g3

The output is "MCresult [move] [score] [cpu id]" as

MCresult g2g3 64 0

MCresult g2e2 99 3

MCresult g2f2 -30 1

The ComradesGUI shall decide it easier to access. The internal move selector from [list] is now random. To be better, weight upon results.

A huge **Thank You** to the developers!

Visit <http://ippolit.wikispaces.com/> for more info.