Beyond Stockfish and Leela

John Hartmann

What is a chess engine?

Search

```
239         SearchManager* mainThread = (is_mainthread() ? main_manager() : nullptr);
240
241         Move pv[MAX_PLY + 1];
242
243         Depth lastBestMoveDepth = 0;
244         Value lastBestScore     = -VALUE_INFINITE;
245         auto  lastBestPV        = std::vector{Move::none()};
246
247         Value  alpha, beta;
248         Value  bestValue     = -VALUE_INFINITE;
249         Color  us            = rootPos.side_to_move();
250         double timeReduction = 1, totBestMoveChanges = 0;
251         int    delta, iterIdx                        = 0;
252
253         // Allocate stack with extra size to allow access from (ss - 7) to (ss + 2):
254         // (ss - 7) is needed for update_continuation_histories(ss - 1) which accesses (ss - 6),
255         // (ss + 2) is needed for initialization of cutOffCnt.
256         Stack  stack[MAX_PLY + 10] = {};
257         Stack* ss                  = stack + 7;
258
259         for (int i = 7; i > 0; --i)
260         {
261             (ss - i)->continuationHistory =
262               &continuationHistory[0][0][NO_PIECE][0];  // Use as a sentinel
263             (ss - i)->continuationCorrectionHistory = &continuationCorrectionHistory[NO_PIECE][0];
264             (ss - i)->staticEval                    = VALUE_NONE;
265         }
266
267         for (int i = 0; i <= MAX_PLY + 2; ++i)
268             (ss + i)->ply = i;
269
270         ss->pv = pv;
```

Evaluation

- In the past, this was handcrafted – various numbers attached to position themes
- Today, NNUE (efficiently updatable neural networks) are used in every top engine.

Chess Engines in 2020

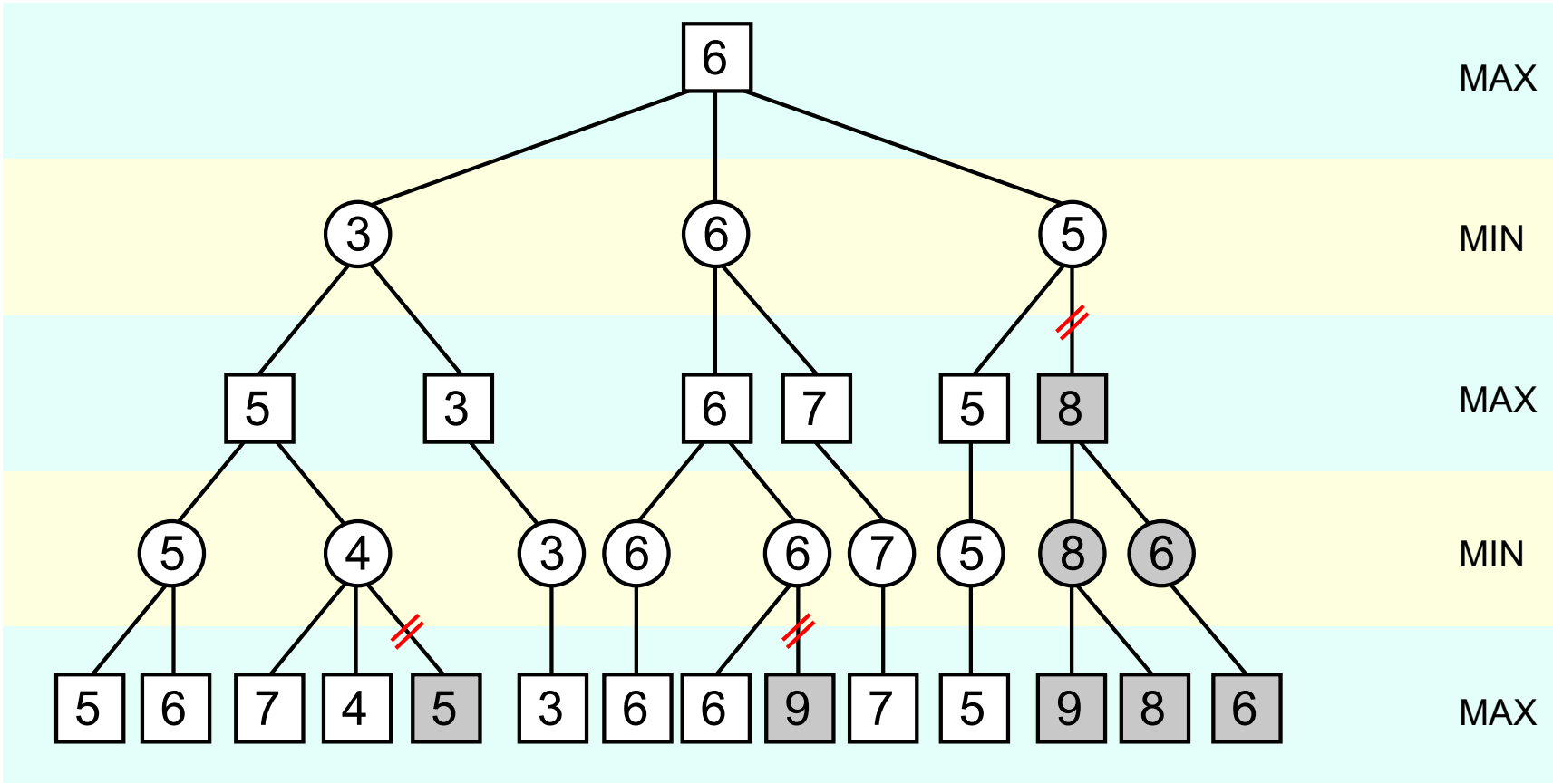
	Alpha Beta	MCTS
Handcrafted	Stockfish classical, Komodo, Houdini.	Komodo MCTS
Neural Network	Stockfish NNUE	Leela and Alpha Zero… *sort of*.

Chess Engines Today

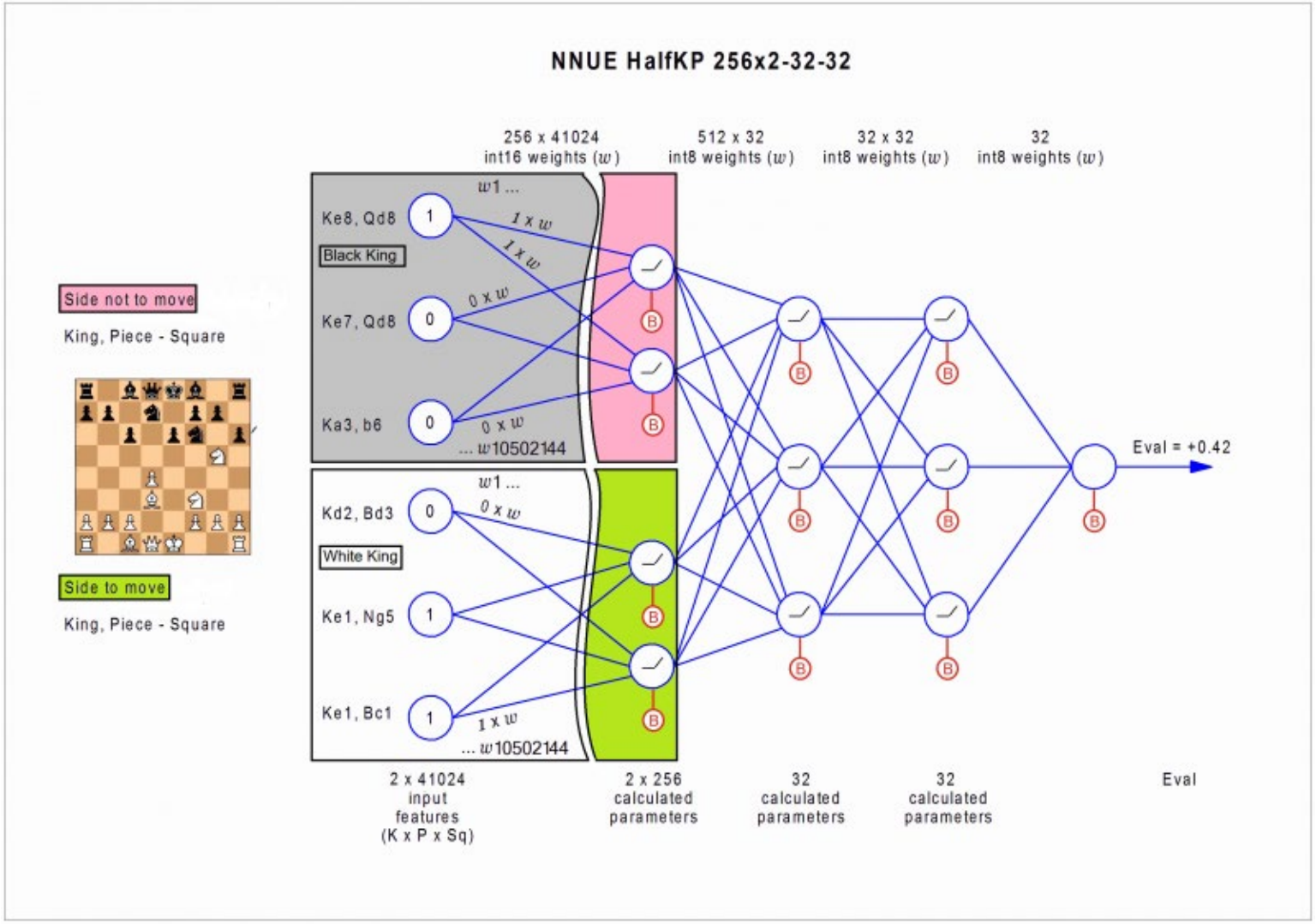
	Alpha Beta	MCTS
Handcrafted	Stockfish classical, Komodo, Houdini.	Komodo MCTS
Neural Network	Stockfish, Torch, Dragon, etc.	Leela … *sort of*.

STOCKFISH

Alpha-beta Search

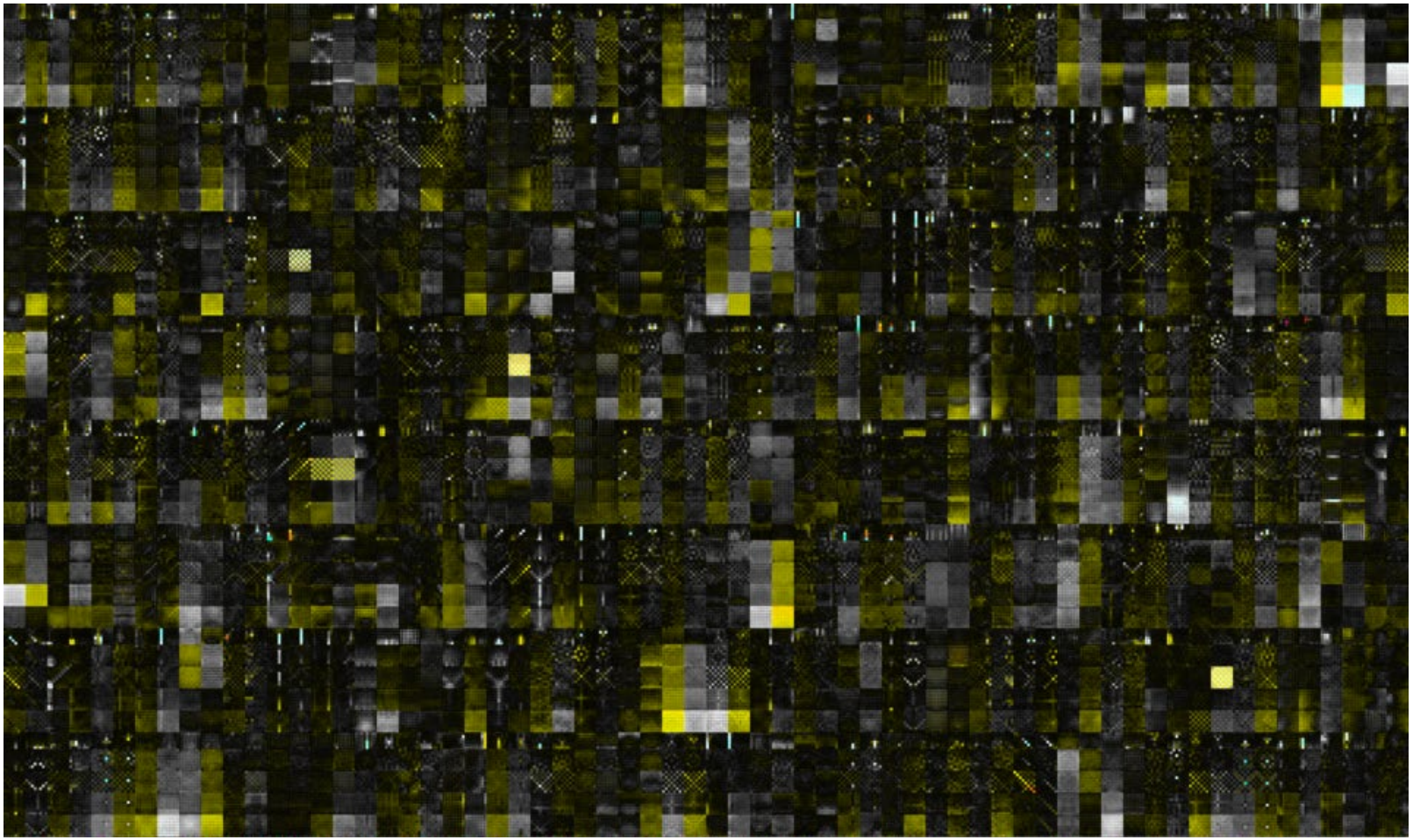


NNUE



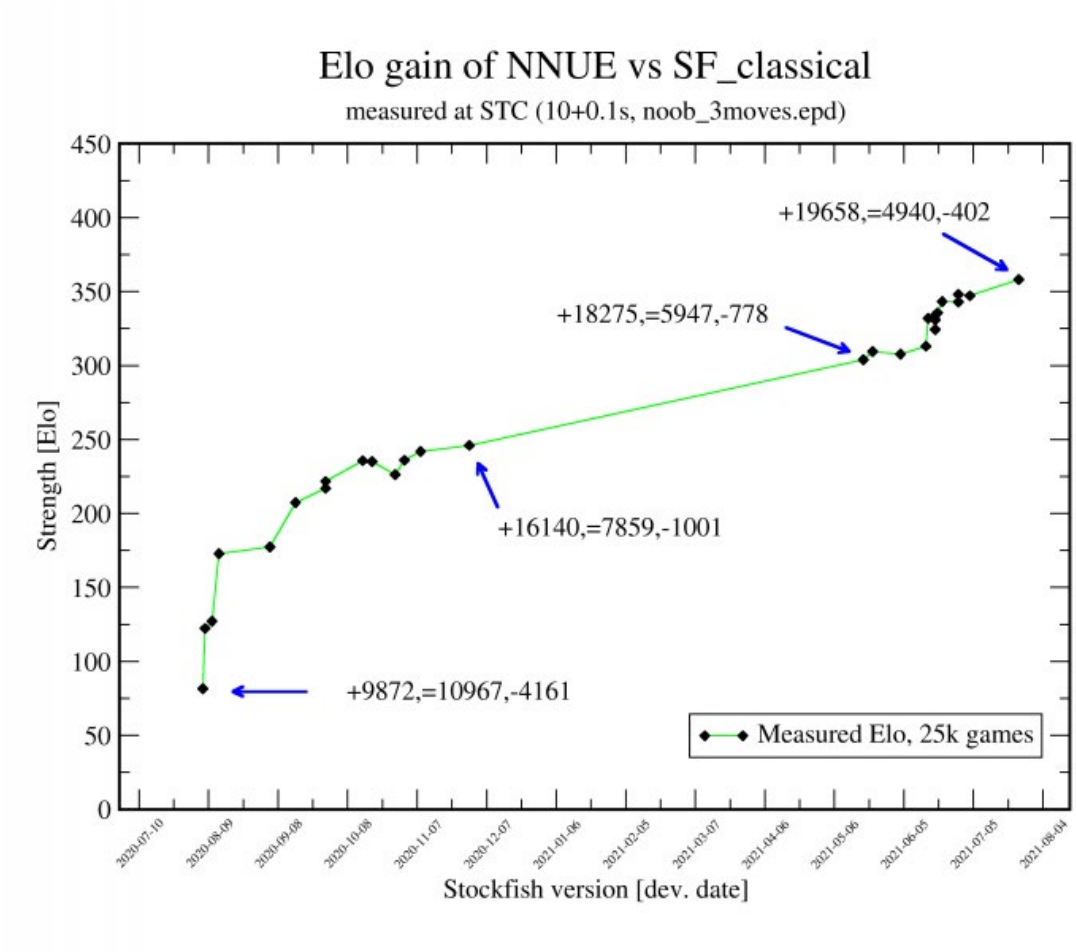
https://www.chessprogramming.org/Stockfish_NNUE

NNUE ‘visualized’



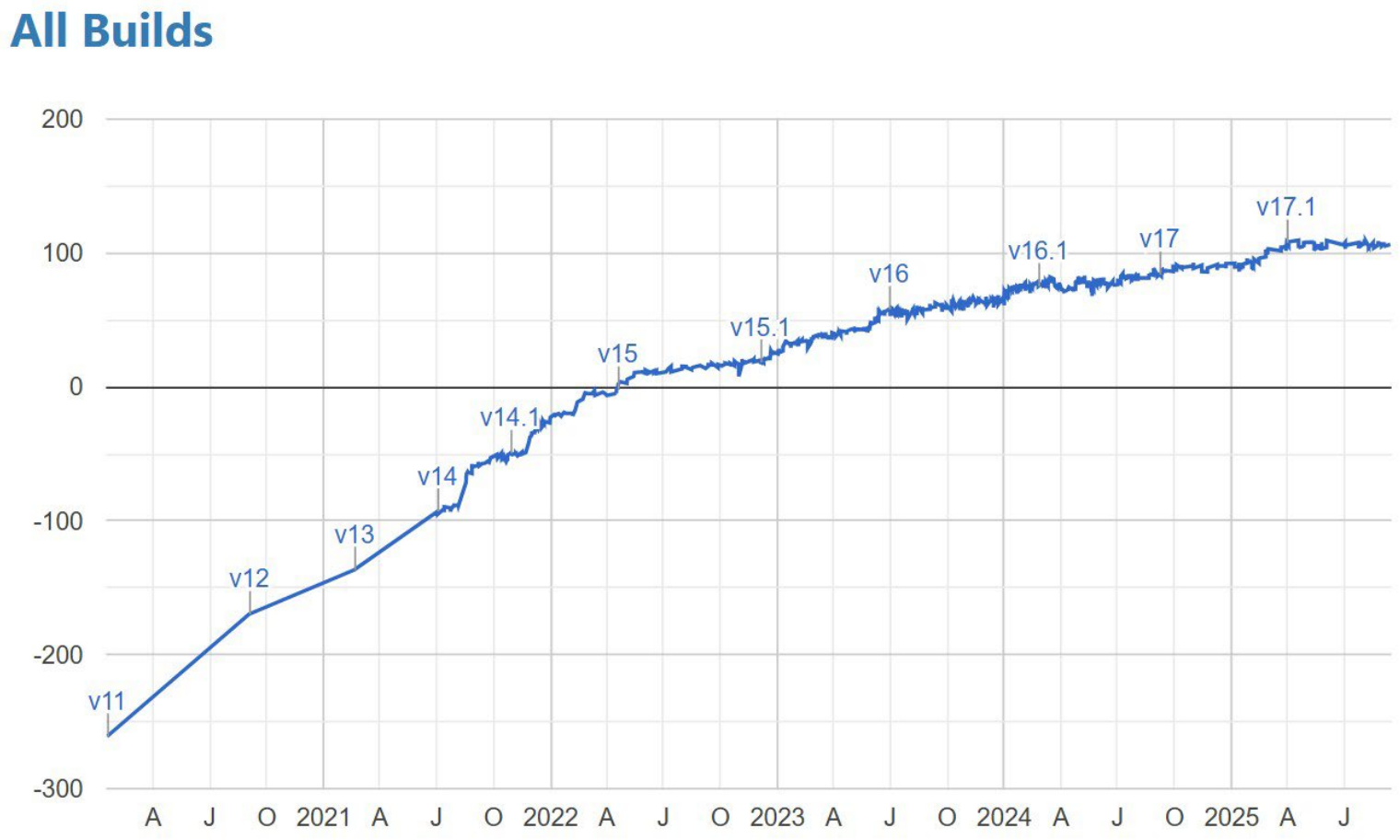
https://en.wikipedia.org/wiki/Stockfish_(chess)

The result – immense strength



https://www.chessprogramming.org/Stockfish_NNUE

The result – stagnation?



https://nextchessmove.com/dev-builds

LEELA

The Alpha Zero Revolution

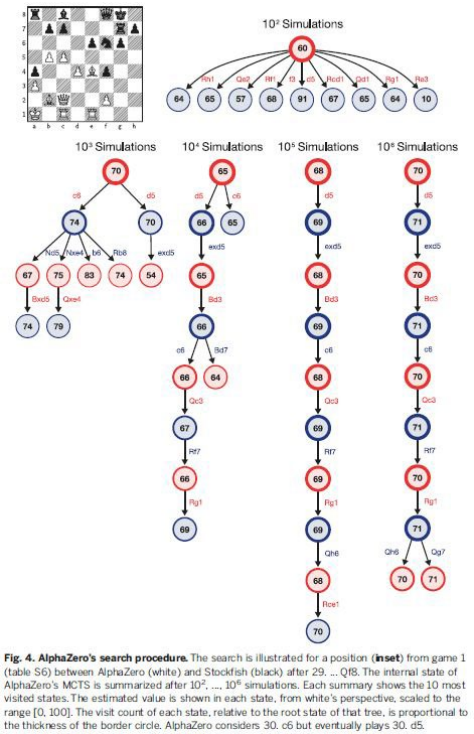
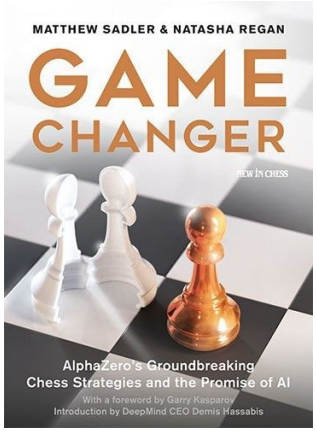


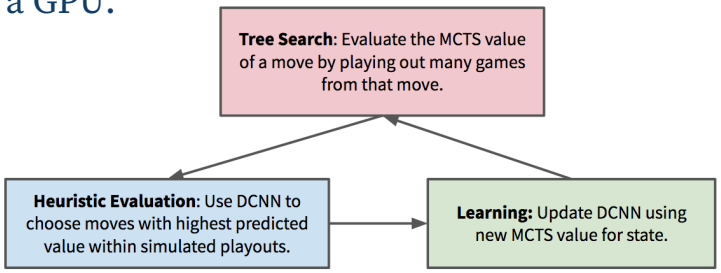
Fig. 4. AlphaZero's search procedure. The search is illustrated for a position (**inset**) from game 1 (table S6) between AlphaZero (white) and Stockfish (black) after 29. ... Qf8. The internal state of AlphaZero's MCTS is summarized after 10^2, ..., 10^6 simulations. Each summary shows the 10 most visited states. The estimated value is shown in each state, from white's perspective, scaled to the range [0, 100]. The visit count of each state, relative to the root state of that tree, is proportional to the thickness of the border circle. AlphaZero considers 30. c6 but eventually plays 30. d5.

Graphic from Silver et al., *Science* 362, 1140–1144 (2018) 7 December 2018

The announcement in 2017 of Alpha Zero's dominant victory of Stockfish shocked the chess world.

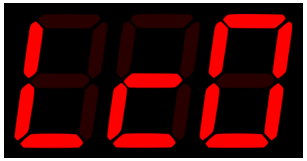
Alpha Zero uses a self-trained (via reinforcement learning) neural network for evaluation, and a 'Monte Carlo' type search (MCTS / UTC), using game playouts from different nodes to search the game tree. It assesses positions probabilistically and needs a GPU.





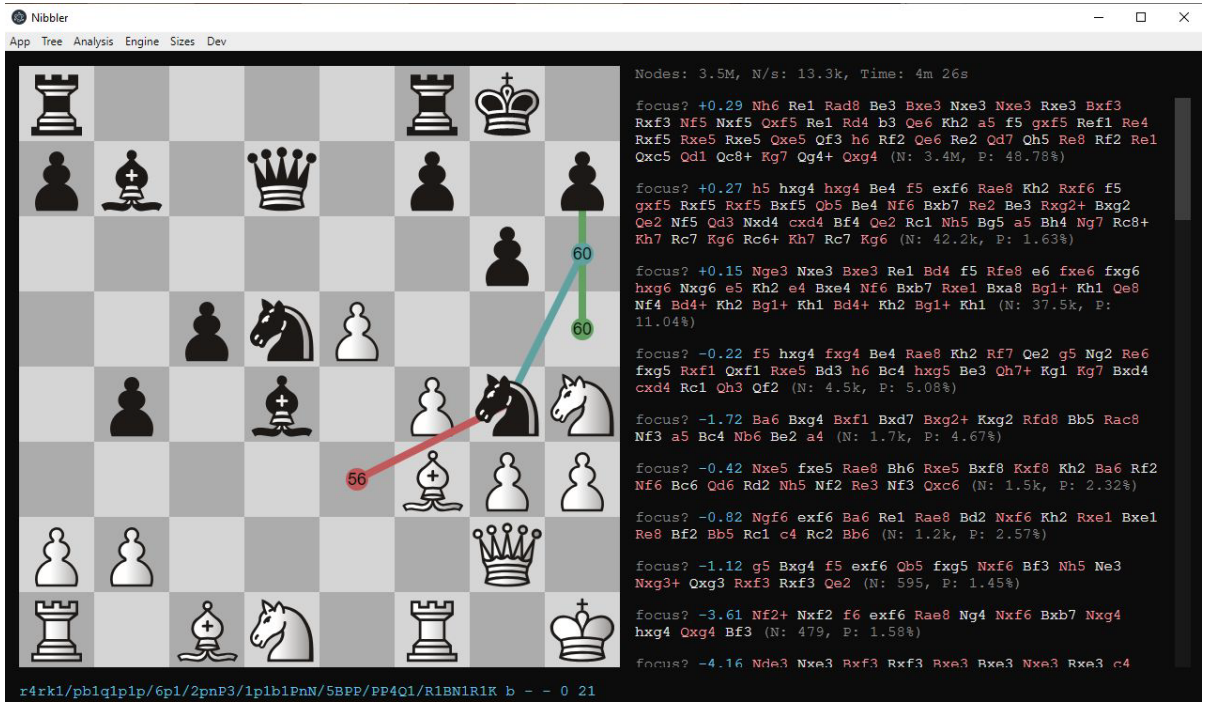
https://nikcheerla.github.io/deeplearningschool/2018/01/01/AlphaZero-Explained/

Leela Chess Zero: The **Real** Game Changer



Almost immediately after the first Alpha Zero preprint appeared, chess enthusiasts – including Gary Linscott, who was a key Stockfish developer – began working to create an open-source chess engine based on the Alpha Zero model and pseudocode.

Like Alpha Zero, Leela Chess Zero is entirely self-trained. (No human knowledge was fed into the learning process.) It learns by playing millions of games against itself, using donated computer time from players around the world. It also uses a type of Monte Carlo Tree Search (UCT) to navigate the game tree.



Leela (0.26.1, 256x20) evaluating the position in Korobov-Shankland (FIDE Online Oly, 2020) after 21. h3 in Nibbler

Leela data drives SF dev

- Beginning in 2021 Stockfish evaluation networks were trained on Leela data; this continues to the present day.
- Leela data is valuable to devs because:
  - It's plentiful – billions of FENs for training
  - It's of good quality
  - Some people have found ingenious ways to filter it for optimal use ("binpacks")

The result

- After intense creative evolution, ideas for new NNUE designs / architectures have slowed. The basics are well known.
- Many use the same data and tools to generate NNUE files – what separates Stockfish is still search.
- Ubiquity of Stockfish (and Leela) permeate – why use other engines?

CPU Engine testing

	Rating	Data	Trainer		Notes
Stockfish 17.1	3500	Leela	Orig	https://github.com/official-stockfish/stockfish	
Torch 3	3411	Orig / Leela	Orig	chess.com	private
Reckless 0.8.0	3411	?	bullet	https://github.com/codedeliveryservice/Reckless	no multi-PV
Obsidian 16	3400	Leela	bullet	https://github.com/gab8192/Obsidian	
PlentyChess 6	3373	Orig	bullet	https://github.com/Yoshie2000/PlentyChess	
Alexandria 8	3339	Leela	bullet	https://github.com/PGG106/Alexandria	
Berserk 13	3331	Orig?	Grapheus	https://github.com/jhonnold/berserk	
Dragon 3.3	3323	Orig	Orig	https://komodochess.com/	$$
Integral 7	3311	Orig	bullet	https://github.com/aronpetko/integral	
Caissa 1.20	3292	Orig	Orig	https://github.com/Witek902/Caissa	
Rubichess 20240112	3238	Orig?	?	https://github.com/Matthies/RubiChess	
Ethereal 14.25	3238	Orig	Orig	https://github.com/AndyGrant/Ethereal	$$
CS Tal 2	3196			https://github.com/ChrisWhittington/Chess-System-Tal-NNUE-2	
Titan	3191	Leela	bullet	https://github.com/jeff-pow/Titan	
Wasp 7.00	3048	Orig	Orig	https://waspchess.stanback.net/	
Patricia 5	3041	Orig	Orig	https://github.com/Adam-Kulju/Patricia	no multi-PV

2+1, 1 thread, 1024 hash, TCEC openings

Leela Engine testing

Rank	Name	Elo	+	-	games	score	oppo.	draws
1	stockfish17-windows-x86-64-bmi2	3497	38	37	100	68%	3378	51%
2	stockfish17.1-windows-x86-64-bmi2	3470	37	37	100	64%	3378	55%
3	torch-v2-windows-avx2-popcnt	3424	15	15	700	56%	3378	53%
4	Obsidian150-avx2-pext	3414	39	39	100	55%	3378	46%
5	torch-v3-windows-avx2-pext	3411	15	15	700	55%	3378	56%
6	Obsidian160-avx2-pext	3390	41	41	85	52%	3378	54%
7	lc0	3378	9	9	1785	44%	3423	54%

2+1, 6 thread CPU, 4096 hash, TCEC openings

Torch

A private *Chess.com* engine by Andrew Grant et al; uses half original and half Leela data to create one network that drives the engine. Currently =2nd in my testing, but I have not tested v4 yet (which recent was sent to testers.) Also provides OpenBench framework for devs to accelerate testing.

Open source stars

Reckless

- Very new engine, but big jumps in strength.
- Written in Rust
- Uses Bullet trainer; not sure about data

Obsidian

- 18-year-old author Gabriele Lombardo
- Leela data, original trainer
- https://www.chessdom.com/meet-gabriele-lombardo-author-of-the-chess-engine-obsidian/

Open source stars

Berserk

- Original data, Grapheus trainer (unique to Berserk?)

PlentyChess

- Original data, bullet trainer
- Initial sense is that evals are very different than SF, Obsidian, etc.

Patricia

- Specifically trained to be aggressive; three nets.
- Weaker than most NNUE, but stronger than SF 11.5!
- **"I am checking games with Patricia engine. It's totally crazy! I haven't seen any engine play like this" - GM Vidit Gujrathi** (world ranking #26, Indian Chess Olympiad gold medalist in 2024), 2025/07/27 on Lc0-discord

What's the big idea?

- Everyone knows Stockfish. Many know and use Leela.
- If you want to generate new opening ideas, you have to find new tools "outside the box."
- Some of these might be worth your time in investigating, *especially* those that use original data or go for style instead of strength.