Turing's Paper Machine (Turochamp) and the reconstructed Turochamp for playing a chess game on your computer.

Alan Turing

About the development of the original Turochamp:

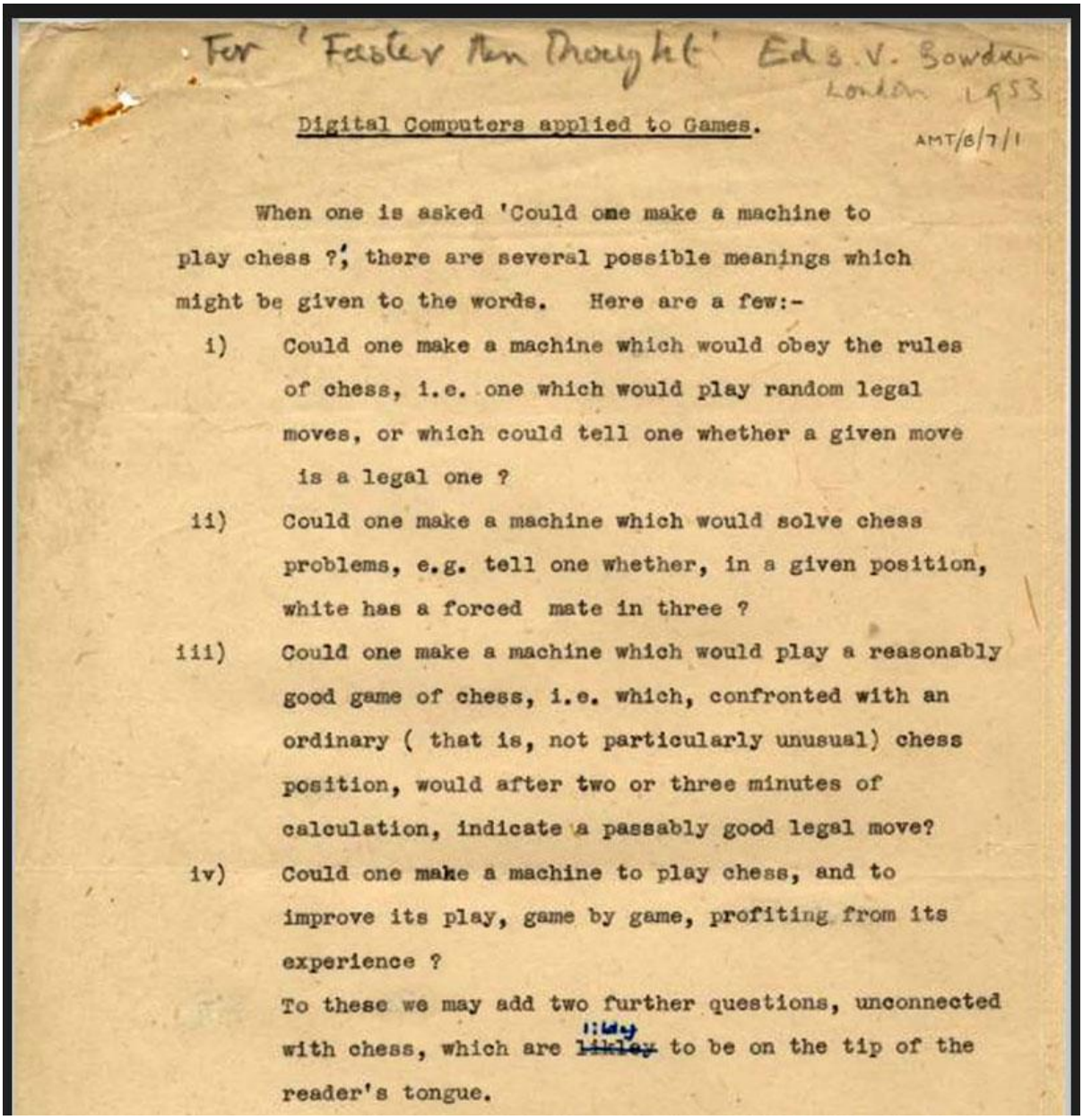
The first chess program was written by Alan Turing, although written a few years *before* computers had been invented

In 1948 Turing, assisted by David Champernowne, wrote the instructions that would enable a future machine to play chess. They called the program ***Turochamp****,* but it popularly became known as “Turing's paper machine”. Turing's goal was to make a machine which would play a game of chess, Turing (and Champernowne) used the following piece values:
pawn=1, knight=3, bishop=3½, rook=5, queen=10 and the following positional evaluation functions:

1. **Mobility.** For the Q,R,B,N, add the square root of the number of moves the piece can make; count each capture as two moves.
2. **Piece safety.** For the R,B,N, add 1.0 point if it is defended, and 1.5 points if it is defended at least twice.
3. **King mobility.** For the K, the same as (1) except for castling moves.
4. **King safety.** For the K, deduct points for its vulnerability as follows: assume that a Queen of the same colour is on the King's square; calculate its mobility, and then subtract this value from the score.
5. **Castling.** Add 1.0 point for the possibility of still being able to castle on a later move if a King or Rook move is being considered; add another point if castling can take place on the next move; finally add one more point for actually castling.

6. **Pawn credit.** Add 0.2 point for each rank advanced, and 0.3 point for being defended by a non-Pawn.
7. **Mates and checks.** Add 1.0 point for the threat of mate and 0.5 point for a check.

Here is a scan of Turing's Faster-Than-Thought paper..



For 'Faster than Thought' Ed B.V. Bowden London 1953

Digital Computers applied to Games. AMT/B/7/1

When one is asked 'Could one make a machine to play chess ?', there are several possible meanings which might be given to the words. Here are a few:-

i) Could one make a machine which would obey the rules of chess, i.e. one which would play random legal moves, or which could tell one whether a given move is a legal one ?

ii) Could one make a machine which would solve chess problems, e.g. tell one whether, in a given position, white has a forced mate in three ?

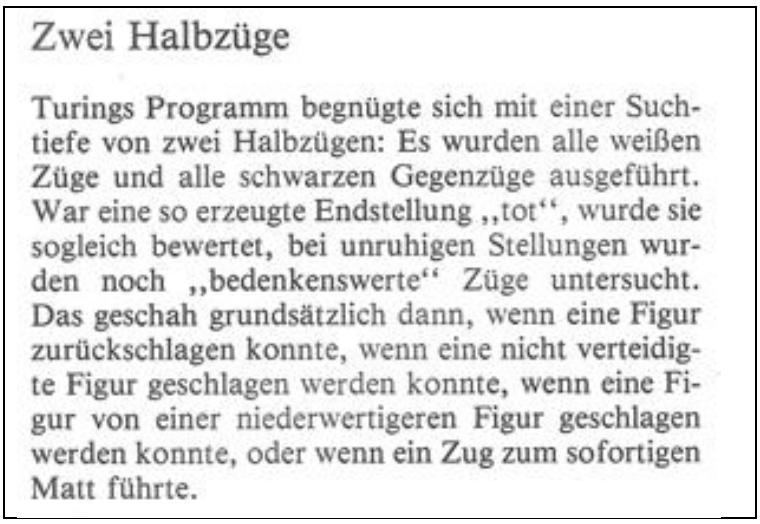
iii) Could one make a machine which would play a reasonably good game of chess, i.e. which, confronted with an ordinary (that is, not particularly unusual) chess position, would after two or three minutes of calculation, indicate a passably good legal move?

iv) Could one make a machine to play chess, and to improve its play, game by game, profiting from its experience ?

To these we may add two further questions, unconnected with chess, which are ~~likely~~ likely to be on the tip of the reader's tongue.

Turochamp incorporated important methods of evaluation, and also the concepts of selectivity and dead position, despite it is unclear how this was "implemented" in the game playing experiments. Champernowne later said: “they were a bit slapdash about all this and must have made a number of slips since the arithmetic was extremely tedious with pencil and paper”. In a CCC forum post, Frederic Friedel mentioned a **search depth of up to three plies**.
https://www.chessprogramming.org/CCC

In CSS december 1986, Frederic Friedel, Die "Papiermachine" von Alan Turing, In this article he mentioned **only 2 plies**, but with the possibility to examen some more moves:



Zwei Halbzüge

Turings Programm begnügte sich mit einer Suchtiefe von zwei Halbzügen: Es wurden alle weißen Züge und alle schwarzen Gegenzüge ausgeführt. War eine so erzeugte Endstellung „tot", wurde sie sogleich bewertet, bei unruhigen Stellungen wurden noch „bedenkenswerte" Züge untersucht. Das geschah grundsätzlich dann, wenn eine Figur zurückschlagen konnte, wenn eine nicht verteidigte Figur geschlagen werden konnte, wenn eine Figur von einer niederwertigeren Figur geschlagen werden konnte, oder wenn ein Zug zum sofortigen Matt führte.

The first registrated computer chess game.

There is a saved game from 1952 against a colleague, Alick Glennie, Turing acted as a human CPU, using paper and pencil, requiring more than half an hour to calculate a move. Turochamp's search algorithm was simple: **Calculate every possible move to 2 plies**, and then continue with all forced moves, checks, and recaptures until no more were possible. Its evaluation was likewise simple, with a handwritten function much like every other pre-AlphaZero engine. Evaluations included terms for Piece values, Piece mobility, Piece defense and King safety.

(**Note**: 2 plies search or 3 plies search? When Frederic Friedel contacted Donald Michie, who had been with Turing at Bletchley, describing the problem in general and the most significant deviations the ChessBase Turing program produced after the prescribed **three-plies search** (which Turing used in his paper-and-pencil game).

Furthermore the thinking time of his opponent is unknown.

Turing's paper machine – Alick Glennie, Manchester 1952: 1.e4 e5 2.Nc3 Nf6 3.d4 Bb4 4.Nf3 d6 5.Bd2 Nc6 6.d5 Nd4 7.h4 Bg4 8.a4 Nxf3+ 9.gxf3 Bh5 10.Bb5+ c6 11.dxc6 0-0 12.cxb7 Rb8 13.Ba6 Qa5 14.Qe2 Nd7 15.Rg1 Nc5 16.Rg5 Bg6 17.Bb5 Nxb7 18.0-0-0 Nc5 19.Bc6 Rfc8 20.Bd5 Bxc3 21.Bxc3 Qxa4 22.Kd2 Ne6 23.Rg4 Nd4 24.Qd3 Nb5 25.Bb3 Qa6 26.Bc4 Bh5 27.Rg3 Qa4 28.Bxb5 Qxb5 29.Qxd6 Rd8 0-1.

You can replay this game in the chess animation on this site:

https://www.schaakcomputers.nl/computerschaak/chessx1.php?item=1&merk=Turochamp,%20Turings%20papermachine

In the early 1950s Turing began Turochamp on a Ferranti Mark 1 computer Although Turing began working on implementing and programming Turochamp to computer, the Ferrenti Mark 1, in 1950 at the University of Manchester, his work was not finished due to his tragic and untimely death in 1954.

About the development of the reconstructed Turochamp:

Nobody seemed to have written an actual computer program using Turing's algorithm. In 2004 the programming team at the ChessBase software company decided to create a chess engine based on Turing's algorithms. The basis was Turing's description and the sample game against Glennie, Although the original algorithm has since been lost, it was reconstructed by Ken Thompson and Mathias Feist. They built a standard chess program interface for the ChessBase *Fritz* program, so that the Turing engine could be tested.

The ChessBase team had a problem: the engine refused to duplicate all of Turing's moves as recorded in the Glennie game. It deviated in ten places .For instance the problem of the very first move: Turing played 1.e4, the ChessBase come up with 1.e3.
One wonders if Turing made 20 calculations, each taking a number of minutes, before he executed the first move, 1.e2-e4. It is one of the two most common moves that start a chess game

Problaby Alan Turing did not care about details; he was interested in the general principle. Champernowne, who had assisted Turing during the 1952 game said : "In the actual experiment I suspect we were a bit slapdash about all this and must have

made a number of slips since the arithmetic was extremely tedious with pencil and paper."

An important point is that Turing, working with paper and pencil, clearly used his intuition to come up with the moves. And without knowing it he used Alpha-Beta pruning, a technique that was invented by Allen Newell and Herbert A. Simon half a decade later, when machines could actually execute code.

Mathias Feist wrote "Turing's calculation for e2-e4 is correct, which means he did not really calculate e2-e3. As a chess player he knew that this move is not logical, because in the initial position king safety doesn't play a role. He probably looked at it and thought: everything is the same for both moves, except e2-e3 is 0.2 points worse because the pawn moves just one square. So he discarded it without further calculation."

There were also some problems with the quiescence search ("captures had to be followed up at least to the point where no further capture was immediately possible"), as applied by Turing in the Glennie game.

There are a number of other deviations in the game, more towards the end, when Turing was apparently tiring. The game clearly demonstrates how tedious and error-prone human calculation in such a situation can be. On a modern computer the ChessBase Turing engine needs less than five milliseconds to calculate the values for all legal moves in a given position, and to choose the best one to play. Turing took fifteen to twenty minutes.

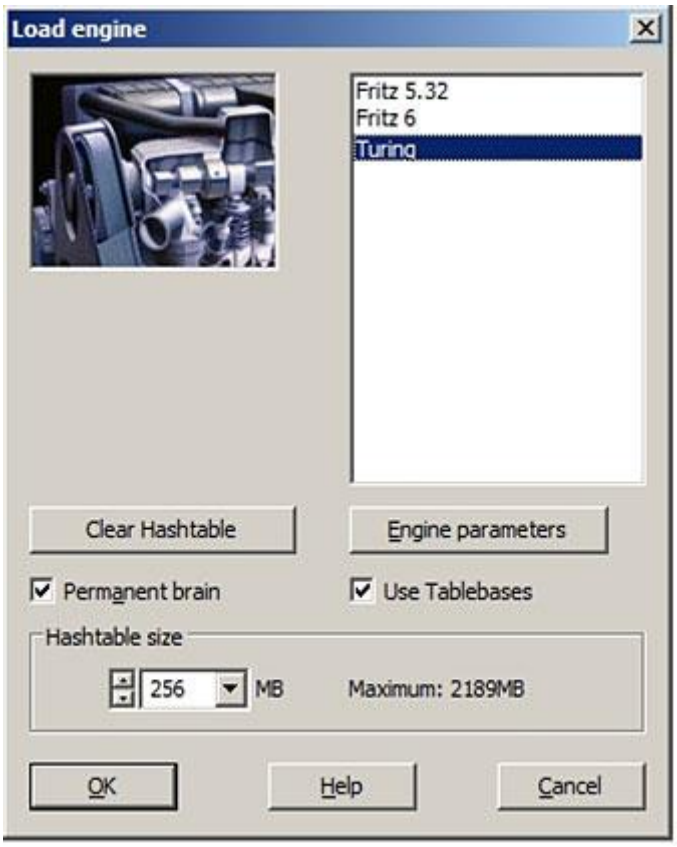
At the ChessBase site https://en.chessbase.com/post/reconstructing-turing-s-paper-machine it is said **The reconstruction of the algorithm and digitalisation in 2000** (?, ment is 2012 I think) **was faithful in that the program still works by calculating just 2 moves ahead.** It is to say 2 moves is 4 ply. Here was problaly ment 2 ply.

In June 2012 Frederic Friedel travelled to Manchester to attend the Alan Turing Centenary Conference. In Manchester he assisted Garry Kasparov in a commemorative lecture on Turing's chess involvement. The two spoke about the reconstruction of the paper machine, and Kasparov actually played a game against it: Video lecture by Garry Kasparov and Frederic Friedel

There Kasparov said: "I think that at five seconds a move it could beat most amateurs…

Turochamp (Turing) reconstructed in practice.

So, after downloading and imported in the Fritz chess programm, Turochamp reconstructed under the name "Turing", you have the possibility to play wit it by yourselve or against another computer.



Playing against the reconstructed Turochamp is astonishing. You must certainly be alert and concentrated. I didn't find specific information about the programm, but this are my main experiences:

- A much deeper search than by the original. When we take over **the 2 plies "A strategy"** a **6/7 plies search ("B strategy")** in the middle game at tournament level (3 min./move) is normal. In the endgame the search depth varies from 7 to 11 plies.
- An opening book is available for immediate respons.
- It thinks in the opponent's time.
- Playing perfectly within the given time limits.
- You can watch the numbers of possible moves of the investigated ply search depth.
- Many possibilities for level choice.
- Does not react properly on a pre-installed plies search.
- Unknown or given elo rating.

You can watch several games on www.schaakcomputers.nl or more specific on https://www.schaakcomputers.nl/computerschaak/chessx1.php?item=1&merk=Turochamp,%20Turings%20papermachine I think an estimated elo rating of 1450 is realistic.

In practice (see above) Turochamp reconstructed calculated untill 11 ply search!
It could be the A/B strategie, that is to say 2 ply A-strategy brute force and beginning at ply 3 a B-strategy?

What to say about the strenght of this reconstructed version? Where the notifications of Alan Turing, from about 1950, enough to let it play that strong (let us say elo 1450) on a modern computer we have nowadays, unless the high speed and some extra possibilities?

On a modern computer the ChessBase Turing engine needs less than five milliseconds to calculate the values for all legal moves in a given position, and to choose the best one to play.

Oktober 2022, Luuk Hofman

References:

- https://en.chessbase.com/post/reconstructing-turing-s-paper-machine
- 12-1986, CSS, Frederic Friedel, Die "Papiermachine" von Alan Turing
- 1983, Dr. H.J. van den Herik. Computerschaak, schaakwereld en kunstmatige intelligentie.
- Internet search, divers.
- Experiences with own games played with the reconstructed Turochamp.