

Trainerarbeit im Rahmen der B-Trainer-Ausbildung des Niedersächsischen Schachverbandes 2022

Prüfer: FM Bernd Laubsch, Volker Janssen, GM Dr. Karsten Müller

Abgabe: 13.04.2022

Engine an – Kopf aus

Probleme und Chancen bei der Verwendung von Schachengines

von Tobias Vöge

Inhaltsverzeichnis

1. Einleitung.....	2
2. Technische Vorklärungen	
2.1 Alpha-Beta-Engines	
2.1.1 Minimax-Prinzip.....	4
2.1.2 Alpha-Beta-Pruning.....	5
2.1.3 Bewertungsfunktion.....	6
2.2 „Künstliche Intelligenzen“	
2.2.1 Prinzipien einer KI.....	6
2.2.2 Monte Carlo Tree Search System.....	8
2.3 Hybride Ansätze.....	13
3. Sachgemäße Enginenutzung	
3.1 Grenzen von Engines	
3.1.1 Probleme bei Studien und Partien.....	14
3.1.2 Mensch gegen Maschine.....	21
3.2 Der sinnvolle Gebrauch	
3.2.1 In der Forschung.....	31
3.2.2 Im Training.....	34
4. Fazit.....	39
5. Quellenverzeichnis.....	40

1. Einleitung

„Dann fragen wir halt einfach die Engine, was richtig ist“ auf der einen und „das ist doch Engineschach!“ auf der anderen Seite sind Sätze, die jeder Schachspieler im Laufe seiner Karriere von Trainern oder Mitspielern zu hören bekommt oder sogar selbst äußert. Zeugt das erste Zitat noch von einem Glauben, alle schachlichen Probleme könnten augenblicklich durch einen mechanischen Allwissenden gelöst werden, so kann letzteres je nach Situation durchaus als Beschimpfung des scheinbar dummen oder realitätsfremden Blechkopfs gesehen werden.

Dieses Spannungsfeld weist auf Probleme hin, die durch ein falsches Bild von der Einbeziehung elektronischer Hilfsmittel im Schach vermittelt werden. Die korrekte Nutzung einer Engine kann in der Forschung wie auch im Training meiner Meinung nach bereichernd und motivierend wirken. Falsche Nutzung hingegen kann das Gegenteil erzeugen.

Das Thema hat gerade in der heutigen Zeit eine stärker werdende Relevanz, da durch die Digitalisierung selbst Einsteiger schnell mit der Engine in Kontakt kommen – in der Regel ohne Anleitung. Es ist aber auf der anderen Seite auch für professionelle Spieler nicht wegzudenken, wie man etwa an den mit Computerunterstützung ausgefeilten Eröffnungsrepertoires junger emporschnellender Großmeister sieht.

Die Untergliederung und Funktionsweisen von Alpha-Beta-Engines, Engines, die mit Hilfe von neuronalen Netzen agieren und Hybriden wird im Kapitel 2: „Technische Vorklärungen“ behandelt. Es werden jeweils die Hauptprinzipien behandelt und bewusst auf eine historische Entwicklungsgeschichte verzichtet.

Kapitel 3: „Sachgemäße Enginenutzung“ soll anhand von Beispielen konkretisieren, wo Grenzen und Chancen im Umgang mit Engines liegen. An einigen Stelle werden schachliche Beispiele verwendet, deren Auflösung jeweils direkt danach geschieht. Wer die Aufgaben selbst lösen möchte, muss sich also zwingen, kurz innezuhalten und über die Stellung nachzudenken. Um die Lösungen mancher Aufgaben zu verstehen, ist das Wissen eines soliden Vereinsspielers (etwa 1800+ Elo) von Nöten. Theoretisch sollten die wesentlichen Punkte aber auch ohne die entsprechenden Beispiele verständlich sein.

Nicht behandelt wird der illegale Einsatz von elektronischen Hilfsmitteln zur Leistungssteigerung. So ist z.B. „unsachgemäße Nutzung“ nicht gleichbedeutend mit

„im Rahmen eines Turniers illegal“, sondern bezieht sich auf einen hinderlichen Effekt bei der Wahrheitsfindung in einer Stellung oder bei pädagogischen Aktionen.

Das Ziel der Arbeit ist es, mit Halbwissen bezüglich der Funktionen von Engines aufzuräumen und ein paar Hinweise für die pädagogisch und schachlich-wissenschaftlich Sinn bringende Verwendung von Engines zu liefern. Adressaten sind vor allem interessierte Schachspieler sowie Trainer. Die behandelten Themen können meiner Ansicht nach gut als Unterrichtsinhalte verwendet werden, deshalb gibt es an manchen Stellen mehr Beispiele als vielleicht rein wissenschaftlich notwendig wären.

Begrifflich hat sich die bereits erwähnte englische Bezeichnung „Engine“ (auf Deutsch: „Motor“) in der heutigen Zeit für alle Schachprogramme auch im deutschsprachigen Raum weitgehend durchgesetzt, weshalb ich den Begriff als Fachterminus beibehalte. Der sprachlichen Einfachheit halber wird im Folgenden nicht gegendert, wenn etwa von „Schachspieler“ die Rede ist, ist gleichsam auch die Schachspielerin gemeint.

2. Technische Vorklärungen

2.1 Alpha-Beta-Engines

Schach ist ein Spiel mit perfekter Information, sodass es theoretisch möglich wäre, alle Varianten, wie sich eine Partie entwickeln könnte, herauszufinden und jeweils den definitiv besten Zug zu wählen. Im Universum gibt es allerdings bekanntermaßen nur etwa 10^{86} Atome – im Vergleich zu etwa 10^{120} mögliche Schachpartien massiv weniger. Auch mit Quantencomputern, die allen Speicherplatz der Welt hätten, wäre es also nicht möglich, einen Laplaceschen Dämon für Schach zu entwickeln.

Da eine allwissende Engine als reiner Speicherort der Wahrheit nicht umsetzbar ist, muss das Ziel auf der möglichst korrekten Wiedergabe des Resultats liegen, das durch die besten Züge beider Seiten erreichbar ist. Bewertungen wie „Weiß steht besser“ (inklusive einer Zahl) bilden also nur eine Tendenz ab. Diese und den bestmöglichen Zug (wobei er sich durch die Stellungsbewertung ergibt) plus Variante anzugeben ist die Funktion von Engines.

2.1.1 Mini-Max-Prinzip

Für die Errechnung der optimalen Variante nehmen wir an, dass beide Spieler ab einer fraglichen Partiestellung immer den bestmöglichen Zug für sich wählen. Eine positive Zahl bei der Stellungsbewertung zeigt weißen, eine negative schwarzen Vorteil an. Bei 0 ist die Stellung ausgeglichen. Das Mini-Max-Prinzip bedeutet nun, dass jeweils eine Variante dadurch ausgewählt wird, dass abwechselnd Weiß (als Maximum-Spieler) den Zug mit der höchsten Stellungsbewertung auswählt und Schwarz (als Minimum-Spieler) den Zug mit der geringsten.

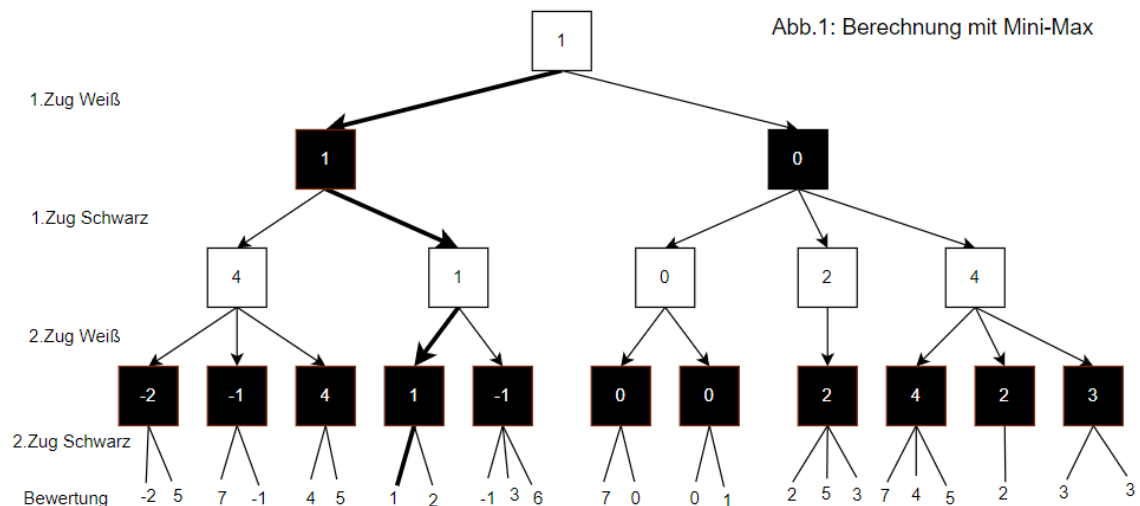
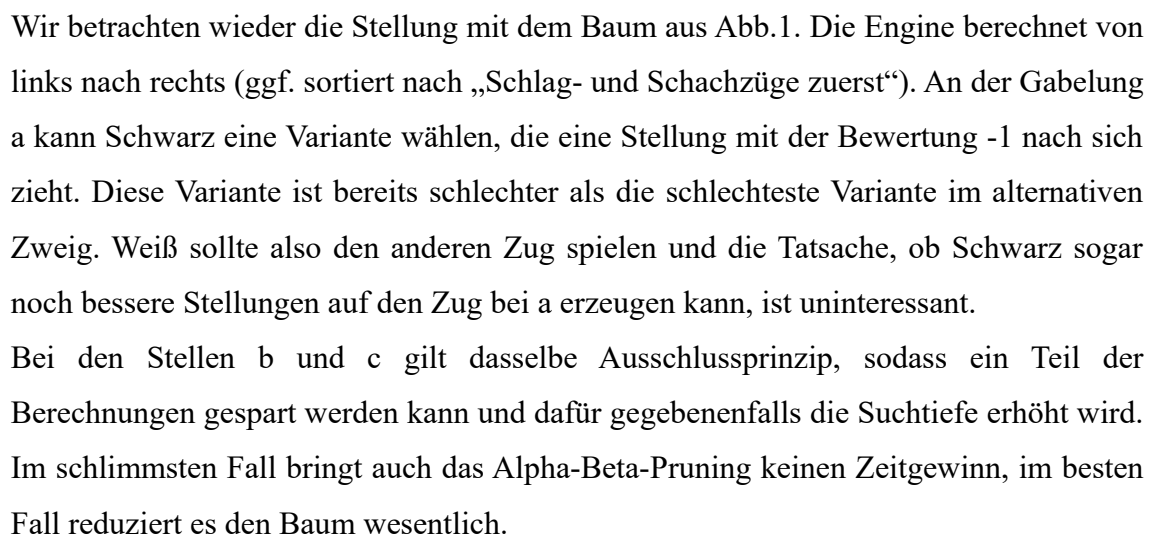


Abbildung 1 zeigt einen solchen Mini-Max-Baum mit einer Tiefe von vier Halbzügen. Die Idealvariante ist mit fetten Pfeilen gekennzeichnet. Die Quadrate heißen in der Fachsprache Knoten. Vereinfacht wurde angenommen, dass jeweils nur ein bis drei Züge ab jeder Stellung möglich sind, wie dies in manchen Endspielen der Fall ist.

Nun ist aber nicht jede Stellung ein solches Endspiel und bei einer Streuung von durchschnittlich etwa 30 bis 35 möglichen Halbzügen pro Stellung kann man sich vorstellen, dass die unvorstellbar hohe Zuganzahl selbst für eine Maschine nicht in der gewohnten Tiefe zu bewerkstelligen ist.

Das Problem eines zu großen Baumes ist einerseits über die Rechentiefe begrenzt, andererseits kennen wir bereits aus der menschlichen Variantenberechnung Methoden, um den Rechenaufwand möglichst effizient zu gestalten.

Eine effiziente menschliche Variantenberechnung funktioniert häufig mit einem Ausschlussverfahren. Auch die Engine macht sich eine solche Methode zu eigen, namentlich das „Alpha-Beta-Pruning“ (deshalb auch „Alpha-Beta-Engines“ für Engines des traditionellen Ansatzes).



5

2.1.3 Bewertungsfunktion

Lediglich für eine Reihe von Endspielen mit überschaubarem Figuren- und Zugpotenzial liegt in der Tablebase eine absolut wahre Bewertung vor.¹ Für alle anderen Stellungen muss wie bereits erwähnt eine ungefähre Tendenz angegeben werden. Für die Bewertungsfunktion untersuchen traditionelle Engines Merkmale, die der Mensch vorgegeben hat.

Diese Merkmale bilden Material, Bauernstruktur, Königssicherheit, Figurenaktivität und einige weitere.² Sie werden je nach Partiephase anders gewichtet und für das Programm verständlich konkretisiert. So wird etwa für das Material der relative Figurenwert beider Parteien verglichen oder für die Figurenaktivität werden Läufer auf offenen Diagonalen oder Springer auf Vorpostenfeldern besser bewertet. Je nach Gewichtung der einzelnen Merkmale kommt die Engine zu einem internen Ranking und gibt am Ende eine entsprechende Bewertung in Bauereinheiten aus.

Die differenzierte Gewichtung der verschiedenen Bewertungsmerkmale ergibt sich traditionell aus der Absprache mit starken menschlichen Spielern und manuellen Versuchen. In diesem Punkt unterscheiden sich die Alpha-Beta-Engines untereinander. Da der Vorgang des Ausprobierens und Verbesserns ein methodisch nicht sehr vielfältiges Aufgabenfeld und stattdessen vor allem Fleißarbeit darstellt, verwundert es nicht, dass bei manchen Engines dieser Prozess automatisiert wurde. Hierin findet sich der Unterschied zwischen Alpha-Beta-Engine und KI.

2.2 „Künstliche Intelligenzen“

„Tja. Ich habe mich immer gefragt, wie es wohl wäre, wenn Außerirdische auf der Erde landen und uns zeigen würden, wie man richtig Schach spielt. Jetzt fühle ich mich, als wäre es passiert.“

Peter Heine Nielsen

2.2.1 Prinzipien einer KI

Im Match AlphaZero gegen Stockfish 8 im Jahre 2016 passierte etwas Unglaubliches. Durch einen neuen, eher statistisch orientierten Ansatz schaffte es eine Engine komplett ohne Eröffnungs- oder Endspielbuch, dafür mit neuronalem Netzwerk, die beste

¹ Fun fact: Das längste Matt (ohne angewendete 50-Züge-Regel) aus einem Siebensteiner dauert bei optimalem Spiel 549 Züge. Vgl. tb7.chessok.com/probe/3/61. Diese und alle weiteren Internetquellen wurden zuletzt aufgerufen am 18.03.2022.

² Für Details vgl. <https://hxim.github.io/Stockfish-Evaluation-Guide/>.

traditionelle Engine zu bezwingen. Dabei rechnete AlphaZero gerade einmal 80.000 Stellungen pro Sekunde und Stockfish 70.000.000. Wie lässt sich dies erklären?

Zuvor sei noch angemerkt, dass das autodidaktische Programm über mehr Rechenleistung verfügte und der Zeitmodus (eine Minute pro Zug) sowie das fehlende Eröffnungsbuch bei beiden ungünstig für den Fisch waren. Deshalb ist das Ergebnis (28 Siege, 72 Remis, 0 Niederlagen für AlphaZero) mit Vorsicht zu genießen. In einem weiteren Duell Anfang 2018 gelang einer verbesserten AlphaZero-Version mit 155-839-6 ein weiterer Achtungserfolg.³

Außerdem macht die Tatsache, dass AlphaZero mehr Partien gewann, es nicht zwangsläufig zu einer besseren Engine, sondern in erster Linie zu einem besseren Spieler. Wenn man betrachtet, welche Stellungsbewertungen besser zu den tatsächlichen Ergebnissen passten, liegt Stockfish vorne. AlphaZero hatte seine Stellungen im ersten Match in diversen Situationen überschätzt, während der Fisch die Stellung häufig wahrheitsgemäß im Ausgleichssinne empfand.

Aber um auf die Frage zu antworten, warum das Match so verlief: AlphaZero hatte heimlich geübt. Neun Stunden lang. In dieser Zeit spielte das Programm 44 Millionen Partien gegen sich selbst – also mehr als 1.000 Partien pro Sekunde.⁴

Wie bei Anfängern, die gerade einmal die Regeln kennen, gab es zu Beginn sehr viele zufällige Züge ohne erkennbare Muster. Die meisten Partien wurden nach 50 Zügen abgebrochen und für Remis erklärt, um unnötigen Rechenaufwand zu vermeiden.

Die Partien, die trotz verkürzter Zugzahl entschieden wurden, nutzte AlphaZero als Orientierungshilfe. Es bevorzugte in den folgenden Simulationen Züge, die a) eine hohe Wahrscheinlichkeit hatten, der beste Zug der Stellung zu sein b) erfolgversprechende Varianten nach sich zu ziehen schienen (evaluiert durch das „value network“) und c) bisher wenig simuliert worden waren.

Das „value network“ („Werte-Netzwerk“) bezeichnet ein komplexes System, dessen Sinn es ist, Ergebnisvorhersagen auf Basis weniger Stellungen zu treffen. Das System bewertet eine Stellung – anders als bei Alpha-Beta-Engines – nicht aufgrund statischer menschengegebener Merkmale, sondern aufgrund von Verhältnissen dieser Merkmale in einer konkreten Stellung untereinander. Grob gesagt werden z.B. nicht die

3 Vgl. für mehr Details zu den Matches www.chess.com/de/news/view/google-s-alphazero-besiegt-stockfish-in-einem-100-partien-vergleich-3971 und www.chess.com/article/view/computers-in-chess-good-or-evil-part-two.

4 Verfechter der didaktischen Sinnhaftigkeit von Ultrabullet (15 Sekunden/Partie) finden hier vielleicht eine gute Argumentationsfigur.

Figurenaktivität und das Material untersucht, sondern vielmehr wie die Figurenaktivität den Wert der Figuren verändert. Das System wird durch die Ergebnisse der Simulationen beeinflusst und erschafft so eine Art „Intuition“.

Das Vorgehen führt dazu, dass nicht wie bei Alpha-Beta-Engines alle Äste gleich berechnet werden, bis sie eine angestrebte Tiefe erreichen oder durch Pruning ausgeschlossen werden können und ähnelt eher einer gezielten Suche. Auch wird bei der Stellungsbewertung zwischen den simulierten Varianten nicht der bestmögliche Wert genommen, sondern es wird der Durchschnitt gebildet. Diese Art der Generalisierung sorgt später für das besonders strategische Aroma der servierfertigen Züge.

Mit selbst „gelernter“ und optimierter Intuition und Bewertungsfunktion im Gepäck geht es nach der Trainingsphase für die KI in das Match. In diesem werden keine Simulationen mehr durchgeführt, sondern es kommt das Monte-Carlo-Tree-Search-System zum Einsatz. Näheres wird an einem Beispiel im folgenden Kapitel erläutert.

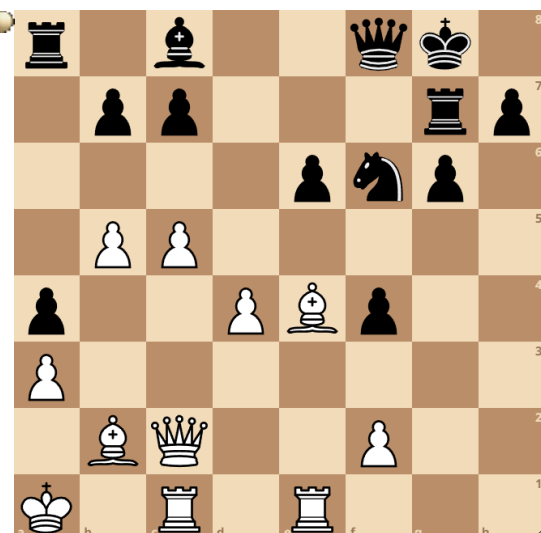
Damit nicht aufgrund von Wissen über die exakte gegnerische Version von bspw. AlphaZero der nächste Zug herausgefunden werden kann, ohne dass der eigene Zug gespielt wurde, wird zusätzlich ein kleiner Zufallsfaktor z.B. für die Eröffnung eingebaut. Unabhängig davon gewinnt ein Gedankenleser aber auch nicht zwangsläufig gegen einen Hellseher, sondern wird im Zweifelsfall selbst zum Hellseher.

2.2.2 Monte Carlo Tree Search System

AlphaZero-Stockfish, London 2018

In der Eröffnung hatte AlphaZero das Läuferpaar erhalten und G- und H-Bauern geopfert. Wir befinden uns jetzt im 30. Zug an einer kritischen Stelle.

Der gemeine Mensch könnte erst einmal über Th1 nachdenken, um die halboffenen Linien zu nutzen. Ein Großmeister würde vermutlich erst einmal nach der Stellungsbewertung vier bis fünf Kandidatenzüge aufzählen, in der



Reihenfolge, in der er sie für sinnvoll hält. Wie Alpha-Beta-Engines an die Stellung herangehen, wissen wir bereits (Kapitel 2.1). Was geht nun AlphaZero vor?

1. Baum (64 Knoten) als Tabellenausschnitt⁵

Zug	Zugwahrscheinlichkeit	Anteil der Knoten, die für diese Variante berechnet wurden	Bewertung (Erwarteter Score für Weiß)	Rang Zugwahrscheinlichkeit	Rang Anteil Knoten für die Suche	Rang nach Erwartung
30.Ld3	29,77%	19,40%	60,10%	1	1	17
30.Lf3	18,82%	13,40%	64,50%	2	2	13
30.c6	16,15%	10,40%	77,30%	3	3	2
30.d5	10,21%	7,50%	87,10%	4	4	1
30.Lg2	4,75%	4,50%	61,60%	5	5	14
30.f3	3,50%	4,50%	67,30%	7	5	7
30.Lh1	4,75%	3,00%	61,60%	5	7	14
30.Dd2	3,50%	3,00%	72,60%	7	7	3
30.Te2	0,41%	3,00%	65,90%	14	7	9
30.Dd3	0,35%	3,00%	65,90%	15	7	9
30.Lc3	0,22%	3,00%	50,80%	17	7	19
30.Tb1	0,08%	3,00%	61,60%	18	7	14
30.Kb1	0,07%	3,00%	57,00%	19	7	18
30.De2	1,90%	1,50%	67,30%	9	14	7
30.Tg1	1,40%	1,50%	65,90%	10	14	9
30.Dc4	1,20%	1,50%	68,70%	11	14	4
30.Th1	1,03%	1,50%	65,90%	12	14	9
30.Tcd1	0,65%	1,50%	68,70%	13	14	4
30.Dd1	0,26%	1,50%	68,70%	16	14	4
Gesamt	100,00%	100,00%	65,70%			

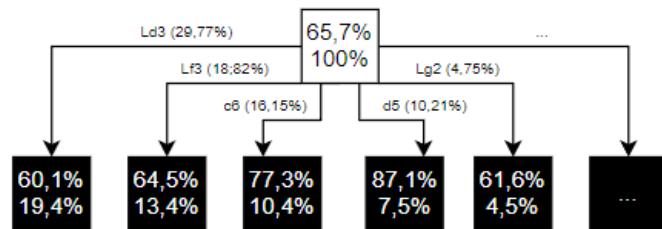
AlphaZero bewertet die Stellung mit 65,7% Punktwahrscheinlichkeit für Weiß. Diese Zahl ergibt sich aus dem Mittelwert aller Punkteerwartungen nach den einzelnen Zügen. Durch welche Ergebnisse genau die Punkteerwartung zustande kommt, ist nicht klar. Eine Stellung mit 100% Remispartien hätte die gleiche Erwartung wie eine mit der Hälfte Weiß- und der anderen Hälfte Schwarzsiegen. Zur Erinnerung: Alpha-Beta-Engines bewerten die Stellung nach der besten Variante und in Bauerneinheiten.

Die Zugwahrscheinlichkeit kennen wir aus dem „value network“, sie stellt die „Intuition“ dar, mit der die Züge auf den ersten Blick priorisiert werden. AlphaZero betrachtet natürlich in diesem ersten Baum alle möglichen Züge, in der Tabelle sind lediglich die 19 Züge angezeigt, die auch ein Mensch eventuell überlegen könnte, um den Umfang nicht unnötig zu vergrößern.

⁵ Sollte die Funktionsweise von AlphaZero einmal Thema in einem Gruppenunterricht werden, möchte ich anregen, die Tabelle im Stile der Fernsehserie „Familienduell“ (www.wikipedia.org/wiki/Familienduell) zu moderieren. „Wir haben 100% AlphaZero gefragt – nennen Sie einen möglichst intuitiven Zug in dieser Stellung“.

Im vorliegenden Fall besitzt 30.Ld3 die höchste Zugwahrscheinlichkeit, weshalb an diesem Zug die meiste Zeit gerechnet wurde. Die Evaluierung der Punkteerwartung zeigte jedoch, dass der Zug 30.d5 am stärksten einzuschätzen ist. Wie der Mensch wirft die Engine nun einen zweiten, tieferen Blick in die Varianten, die sich als würdig erwiesen.

Der erste Baum ist noch sehr flach. Der Kasten rechts soll verdeutlichen, dass es sich nur um einen Ausschnitt handelt und noch die anderen Äste dazu

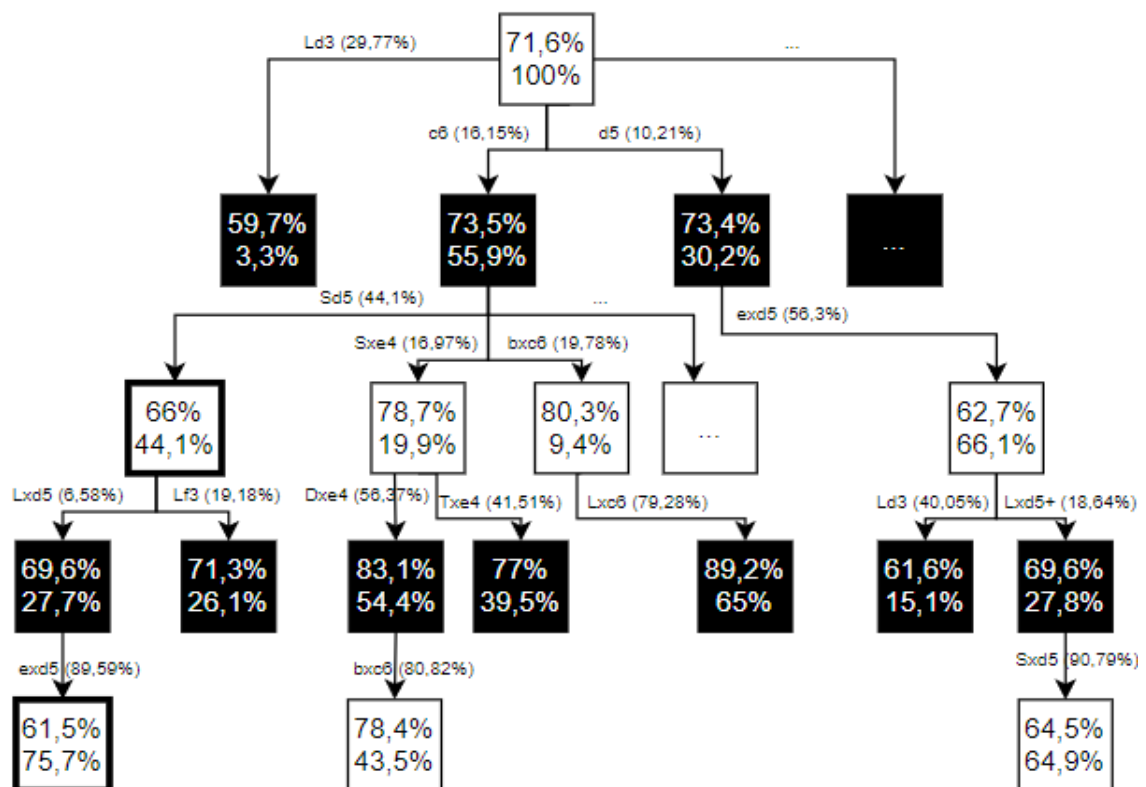


gedacht werden müssen. Die Zahl beim Zug ist die Zugwahrscheinlichkeit, die obere Zahl in den unteren Kästen bildet die Punkteerwartung und die untere Zahl unten den Anteil der Gesamtzeit, der an diesem Zug gerechnet wurde. Die Wurzel (Kasten oben) beinhaltet die gesamte Punkteerwartung und den Zeitanteil, der für alle Äste verwendet wurde (logischerweise 100%). Die hier dargestellten Bäume sind Momentaufnahmen, AlphaZero unternimmt keine solchen Schritte, sondern die Berechnung ist ein fließender Prozess.

2. Baum (256 Knoten) als Tabellenausschnitt

Zug	Zugwahrscheinlichkeit	Anteil der Knoten, die für diese Variante berechnet wurden	Bewertung (Erwarteter Score für Weiß)	Rang Zugwahrscheinlichkeit	Rang Anteil Knoten für die Suche	Rang nach Erwartung
30.Ld3	29,77%	12,30%	60,10%	1	2	8
30.Lf3	18,82%	7,70%	64,50%	2	3	5
30.c6	16,15%	7,30%	77,30%	3	4	2
30.d5	10,21%	53,80%	87,10%	4	1	1
30.Lg2	4,75%	1,90%	61,60%	5	5	6
30.f3	3,50%	1,50%	67,30%	7	7	4
30.Lh1	4,75%	1,90%	61,60%	5	5	6
30.Dd2	3,50%	1,50%	72,60%	7	7	3
Gesamt	100,00%	100,00%	74,10%			

Die Anteile der Züge auf die AlphaZero seinen Rechenaufwand verteilt, haben sich stark verändert. 30.d5, der eben durch seine hervorragende Punkteerwartung hervortrat, kommen über 50% der berechneten Knoten zu, er wird also tiefer berechnet. Mit der erneut erhöhten Bewertung erfüllt er die Erwartungen und bleibt der Favorit.



Der vereinfacht dargestellte Baum wird tiefer (und auf den ersten Blick unverständlicher). Der Hauptfokus liegt nun auf 30.c6, nach dem 30...Sd5 die „intuitiv“ wahrscheinlichste Fortsetzung ist. Darauf wurden die Züge 31.Lxd5 und 31.Lf3 etwa gleichviel betrachtet. Auf ersteren ist 31...exd5 der mit Abstand wahrscheinlichste Folgezug.

Es ist also ein Kopf an Kopf Rennen zwischen 30.c6 und 30.d5, welches erst im nächsten Baum zugunsten unseres Helden entschieden wird. Dieser erhält in über 90% der Aufmerksamkeit die Chance, sich unter Beweis zu stellen und überlebt sogar die tiefere Suche von Baum vier (65.536 Knoten) und fünf (4.194.304 Knoten). Da die Prinzipien klar sein sollten bzw. weitere Bäume eher verwirren, erspare ich sie an dieser Stelle.

Zum Vergleich: Während der Erstellung dieses Kapitels beurteilte mein Hybrid Stockfish 14 bei einer Tiefe von 46 und über 671.000 Knoten ebenfalls 30.d5 als den besten Zug (+1,4 Bauerneinheiten).

In der Partie folgte **30.d5 exd5 31.Ld3 c6 32.Dc3 Tf7 33.Tg1 Dg7** (Diagramm)

Die KI hatte an unserer untersuchten Stelle mit 4...Dh6 als besten schwarzen Zug gerechnet. Weiß gewann in der Folge die schöne Angriffspartie.⁶



Wie können wir das Vorgehen AlphaZeros zusammenfassen? Es hat deutlich weniger Skrupel, statische Stellungsmerkmale zu verschlechtern und stattdessen einer Art „Intuition“ zu folgen. Auf Basis dieser werden lohnenswert scheinende Varianten tiefer berechnet – ein Vorgehen, dass dem des Menschen eher ähnelt. Nachteile gegenüber traditionellen Engines gibt es vor allem, wenn die „Intuition“ trübt. Dies kann etwa in Stellungen mit sehr seltenen taktischen Motiven der Fall sein (siehe dazu Kapitel 3.1.1).

2.3 Hybride Engines

Der neuste Ansatz der Engine-Liebhaber besteht in einer Mischung aus Alpha-Beta-Engine und KI. Das neue Stockfish 14 (z.B. frei zugänglich über lichess.org) stellt einen solchen Hybriden dar und verbindet die herkömmliche rechenstarke Stockfish-Technik und das „intuitiv“ starke neuronale Netzwerk NNUE, sodass die Vorteile beider Seiten genutzt werden. Vorstellbar ist dies, wenn wir die Bewertungen in Bauerneinheiten beim Alpha-Beta-Pruning durch die Bewertungen des neuronalen Netzwerks ersetzen.

Ein weiterer hybrider Ansatz findet sich bei professionellen von Menschen geführten Enginewettkämpfen. Solche gewollten Zusammenspiele von Mensch und Engine sind in der Schachwelt auch als „Zentauren“ bekannt. Etwa auf der Seite infinitychess.com finden regelmäßig Wettkämpfe statt, in denen diverse Hybride zum Einsatz kommen. Namen wie EmanXXX oder lc0fatfritz-cuda hat vermutlich nicht jeder Nahschachprofi gehört, sind dort aber häufig anzutreffen.

⁶ Für die vollständige Partie, kommentiert von Sadler vgl. www.youtube.com/watch?v=bo5plUo86BU.

Prinzipiell spielt bei dieser Form von Hybriden eine KI (z.B. LeelaChessZero⁷) die Eröffnung mit einem Eröffnungsbuch und ab der Mitte der Partie übernimmt Alpha-Beta-Technik (z.B. Eman). So soll der strategische Stil in der ersten Partiephase genutzt werden und die pure Variantenberechnung im Endspiel. Frei nach dem Zitat

„Spiele die Eröffnung wie ein Buch, das Mittelspiel wie ein Zauberer und das Endspiel wie eine Maschine“

von Rudolf Spielmann.

Diese unbekannten Engines sind häufig nur über den Entwickler selbst zu erlangen, sodass der Zugang für die breite Masse wohl verwehrt bleibt. Außerdem sind sie eher für Engine- bzw. Zentaurenwettkämpfe als für menschliche Analysen gedacht.

3. Sachgemäße Enginenutzung

3.1 Grenzen von Engines

3.1.1 Probleme bei Studien und Partien

Es kann – wie bereits erwähnt – keine allwissende Engine geben, trotzdem scheinen Computerprogramme immerhin allen Menschen⁸ überlegen zu sein. Es gibt allerdings Probleme, die der Mensch anscheinend besser lösen kann als die Engine (im Folgenden ist immer Stockfish 14+NNUE gemeint). Betrachten wir ein paar Beispiele:

Die nebenstehende Stellung mit Weiß oder Schwarz am Zuge ist Remis, wenn Weiß lediglich mit seinem König umherzieht und keinen Bauern bewegt. Schwarz kann sich seinerseits nicht befreien und lediglich mit dem Läufer herumtanzen, bis die Partie durch Patt, 50-Züge-Regel oder dreifache Stellungswiederholung entschieden wird. Für den Menschen sollten diese Prinzipien

Diagramm 1a



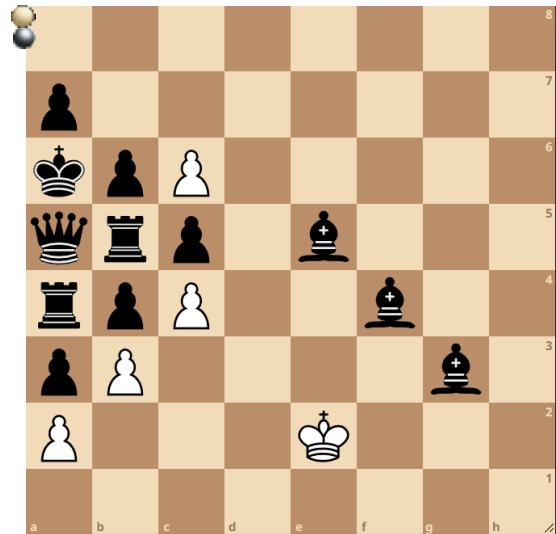
⁷ Lc0 ist der open-source-Nachbau von AlphaZero.

⁸ Auf 70kg übrigens etwa 7×10^{27} Atome. (Vgl. www.greelane.com/wissenschaft-technologie-mathematik/wissenschaft/how-many-atoms-are-in-human-body-603872/, 2019.)

einleuchten, sodass er schnell zur richtigen Bewertung kommt. Die Engine erreicht die Lösung nahezu augenblicklich durch ihre breite Variantenberechnung.

Diagramm 1b

Diagramm 1a wurde von Roger Penrose durch zwei zusätzliche Läufer editiert. Für den Menschen kein Problem, er findet aufgrund seiner Prinzipien (nur den weißen König ziehen, Schwarz kann sich nicht befreien) schnell zur korrekten Beurteilung. Bei einer Tiefe von 59 zeigt die Engine eine Bewertung von -19,9 an, hat also anscheinend nicht verstanden, dass diese Stellung theoretisch Remis ist. Im



Vergleich zu Beispiel 1a sorgen die beiden zusätzlichen Läufer für deutlich mehr Kandidatenzüge und somit mehr einzelne Varianten, die es zu berechnen gilt. Theoretisch ist die Stellung auch durch reine Variantenberechnung lösbar, praktisch liegen der Engine aber noch nicht die notwendigen Ressourcen (Zeit und „Rechenpower“) vor.

Diagramm 2

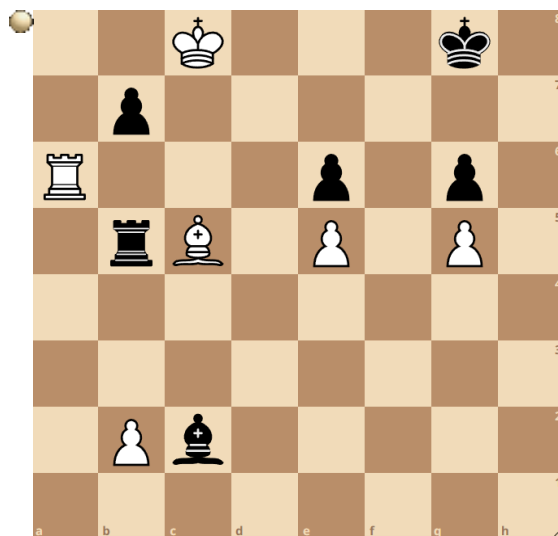
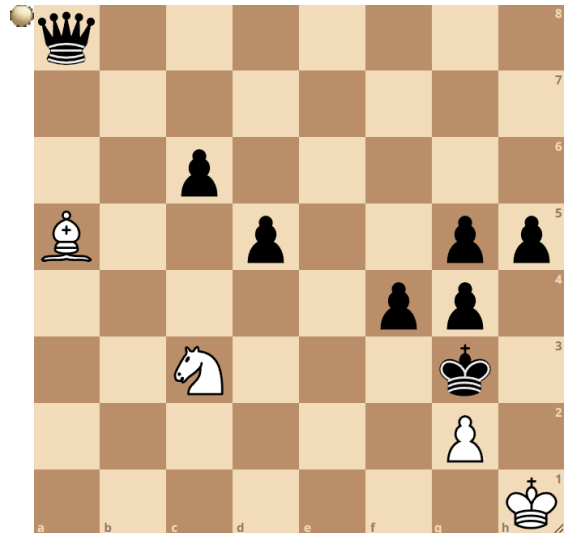


Diagramm 3



Diagramm 4

Auch bei den beispielhaft ausgewählten Diagrammen 2,3 und 4 hat die Engine zu kämpfen. Bei allen drei Studien zeigt Stockfish mindestens bis zu einer Tiefe von 35 schwarzen Vorteil.



Lösungen:

2) **1.Tc6!! bxc6 2.b4** erzeugt einen Vög'schen Turm⁹ und möchte mit Kc7-c8 eine Festung bauen. Die einzige schwarze Idee besteht darin, die Qualität zu opfern und mit dem König durch Zugzwang auszubrechen. **2...Kf7 3.Kc7 Ke8 4.Kc8 Txc5+ 5.xc5 Ke7 6.Kc7 Ld3 7.Kc8 Lb5 8.Kc7 La6 9.Kb6 Lc8 10.Kc7 11.La6 12.Kb6 13.Lc8 Kc7 14.Ka8!** (14.Ka7 Kc7 15.Ka8 Lb7+ 16.Ka7 Kc8–+) **Kc7 15.Ka7=** und Schwarz kann den weißen König nicht weiter abdrängen, ohne Patt zu setzen.

3) **1.La4+ Kxa4 (1...Kc4 2.Lb3+ Kb5 3.La4+=) 2.b3+ Kb5 3.c4+ Kc6 4.d5+ Kd7 8.e6+ Kxd8 9.f5=** und Weiß erreicht eine uneinnehmbare Festung.

4) **1.Se4+! Kh4 2.Sg3!! Df8** Schwarz muss Sf5# verhindern. (2...fxg3 3.Lb6 Da1+ 4.Lg1= Durch den Läufer auf der Diagonalen g1-a7 und zwei Pattideen erreicht Weiß eine Festung.) **3.Le1** droht Sf5## **3...fxg3 4.Lf2! gxf2** (z.B. 4...Da3 und es gibt wieder die Pattidee 5.Lg1=) **5.g3+=**

⁹ In Anlehnung an die Partie Gentemann-Vöge, Verden 2020, bezeichnet dieser Ausdruck einen Turm, der sich nicht mehr durch eigene Kraft aus einem festgesetzten Bereich mit für ihn schlechten Feldern befreien kann und mindestens einen gegnerischen Bauern bedroht.

Diagramm 5

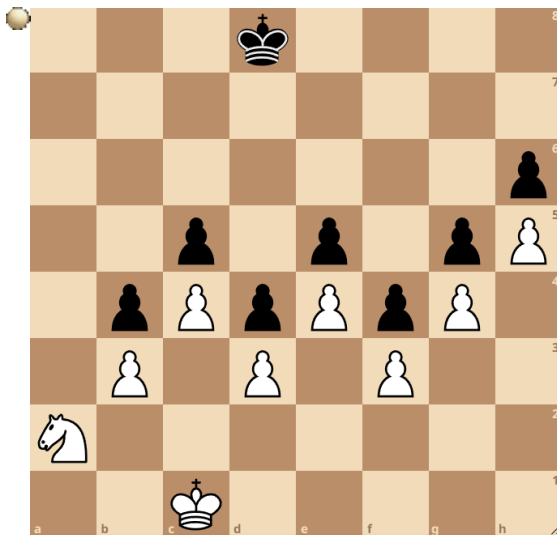


Diagramm 6

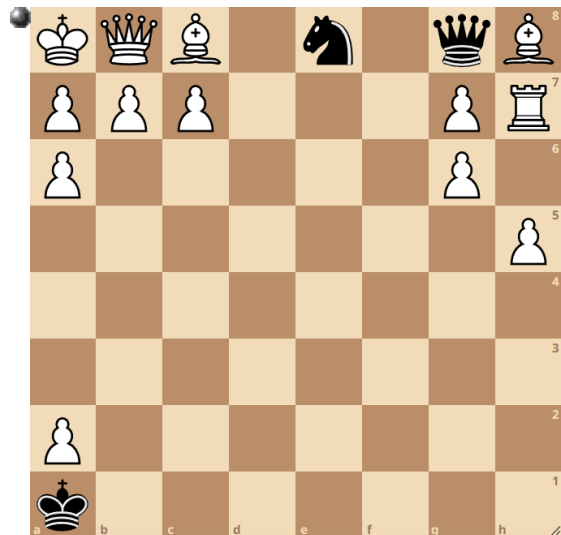
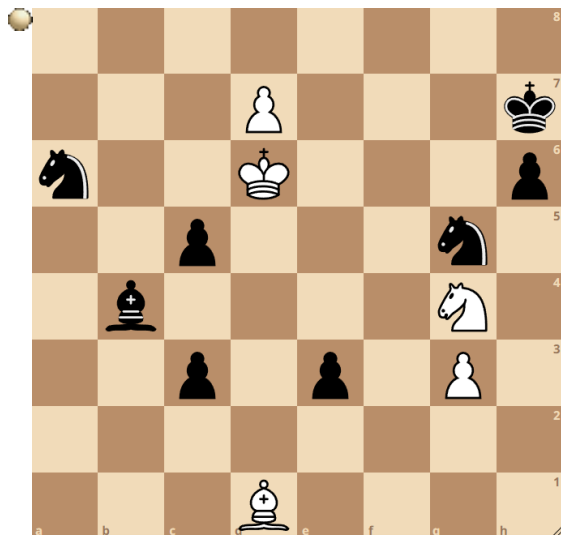


Diagramm 7



Die Komplexität bestand in den bisherigen Studien darin, langfristigen Festungen zu verstehen. Aber auch Gewinnmanöver können der Engine Probleme bereiten. Hierzu wieder drei Studien, welche der Blechkopf selbst bei Tiefe 50 nicht zu lösen vermag.

Lösungen:

5) Weiß muss seinen Springer nach f5 bringen. Dies schafft er über die nachfolgende Route. **1.Kb1 Kd7 2. Sc1 Kd6 3.Se2 Kd7 4.Sg1 Kd6 5.Sh3 Ke7 6.Sf2 Kd6 7.Sd1 Kc6 8.Sb2 Kc7 9.Sa4 Kc6 10.Ka2 Kd6 11.Sb6 Ke6 12.Sd5 Kf7 13.Sc7 Ke7 14.Sb5 Ke6 15.Sa3 Kd6 (15...bxa3 16.Kxa3 Kd6 17.Ka4 Kc6 18.Ka5+-) 16.Sc2 Ke6 17.Se1 Kf6 18. Kb2 Ke6 19.Kc2 Kd6 20.Kd2 Ke6 21.Ke2 Ke7 22.Kf2 Ke6 23.Kg2 Ke7 24.Kh3 Ke6 25.Sg2 Kf7 26.Sh4 Kg7 (26...gxh4 27.Kxh4 Kf6 28.g5+ hxg5+ 29.Kg4+-) 27.Sf5+ Kh7 28.Kg2+-** Schwarz verliert erst h6 und dann alle anderen Bauern.

6) Die einzige Figur, die Weiß potenziell ziehen kann, ist der Turm. Sobald der König nach g5 kommt, und Weiß am Zuge ist, muss ein Bauer gezogen werden, wobei nicht endlos viele Bauernzüge existieren und Weiß irgendwann in Zugzwang gerät. Um jeweils Kg5 im richtigen Moment spielen zu können, muss Schwarz im richtigen Moment ein Tempo verlieren. **1...Kb2 2.Th6 Ka3 3.Th7 Kb4 4.Th6 Ka5 5.Th7 Kxa6** Der König vollführt ein Dreiecksmanöver. Dieses ist nur möglich, wenn er über ein weißes Feld geht. Nur die Felder a6 und h1 kommen in Frage, da sonst der Lc8 Schach geben könnte. **6.Th6 Kb6 7.Th7 Kc5 8.Th6 Kd6 9.Th7 Ke5 10.Th6 Kf6 11.Th7 Kg5 12.a3 Kf4 13.Th6 Ke3 14.Th7 Kd4 15.Th6 Kc3 16.Th7 Kb2 17.a4 Kc3 18.Th6 Kd4 19.Th7 Ke3 20.Th6 Kf4 21.Th7 Kg5 22.a5 Kf6 23.Th6 Ke5 24.Th7 Kd6 25.Th6 Kc5 26.Th7 Kb4 27.Th6 Kxa5 28.Th7 Ka6 29.Th6 Kb6 30.Th7 Kc5 31.Th6 Kd6 32.Th7 Ke5 33.Th6 Kf6 34.Th7 Kg5 35.h6 Kf4 36.Le6 Sxc7#**

7) **1.Sf6+ Kg7 2.Sh5+ Kg6 3.Lc2+ Kxh5 4.d8D Sf7+ 5.Ke6 Sxd8+ 6.Kf5** Das Mattnetz ist gesponnen, jetzt muss nur noch der weiße Läufer Matt geben. **e2 7.Le4 e1S 8.Ld5 c2 9.Lc4 c1S 10.Lb5 Sc7 11.La4 Se2 12.Ld1 Sf3 13.Lxe2 Sa8 14.Lxf3#**

Die drei Studien können menschlich jeweils durch unterschiedliche Hauptmotive in überschaubarer Zeit gelöst werden. Im ersten Fall geht es um die Verbesserung der Figurenstellung, im zweiten um Zugzwang und im dritten ist vor allem das Mattnetz relevant. Für alle drei gibt es Prinzipien, denen nur der Fleischliche folgen kann. Zugzwang kann etwa durch einen Nullzug herausgefunden werden, das Prinzip des Dreiecksmanövers hilft bei der Verwirklichung.

Vor allem Aufgabe 7 ist interessant, da die Engine hier anscheinend nicht an der Tiefe scheitert, sondern fälschlicherweise Varianten ausschließt. Ich erkläre mir dies durch die extrem seltene doppelte Unterverwandlung.

Ein beliebtes französisches Rätsel lautet sinngemäß: In einem See wachsen Seerosen, die jeden Tag ihren Platzbedarf verdoppeln. An Tag 20 wird der See vollkommen zugewachsen sein. Wann füllen die Seerosen genau die Hälfte des Sees?

Lösung: Die Seerosen füllen den halben See an Tag 19. Der springende Punkt ist hierbei der Schluss vom Ziel auf den Weg. Sowohl eine Alpha-Beta-Engine, als auch eine KI als auch ein Hybrid bräuchten für diese Lösung dieses Rätsels Anfangswerte, von denen aus sie vorwärts rechnen könnten. Übertragen auf Schach ist Folgendes festzuhalten: Durch die Kenntnis von langfristigen Motiven und seine Abstraktionsfähigkeit kann der

Mensch wie bei den Seerosen „rückwärts denken“ und kommt so effizienter durch seinen breiteren methodischen Zugriff an die Lösung.

Aus Enginesicht ist das, was wir als richtig erkanntes Motiv/Prinzip kennen, nichts anderes als eine intelligente Verkürzung von Variantenberechnung. Das Problem liegt also nicht darin, dass die Engine mit bestimmten Motiven per se Probleme hat, weil es sie für sie nicht in einem realistischen Sinne gibt.¹⁰ Das Problem ist eher die Langfristigkeit und Überforderung durch zu viele Möglichkeiten.

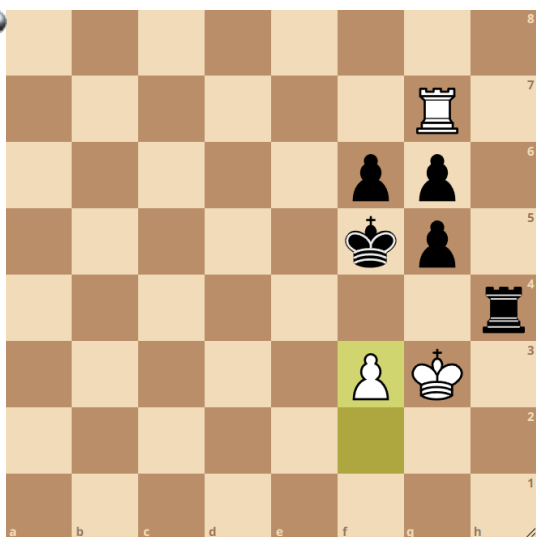
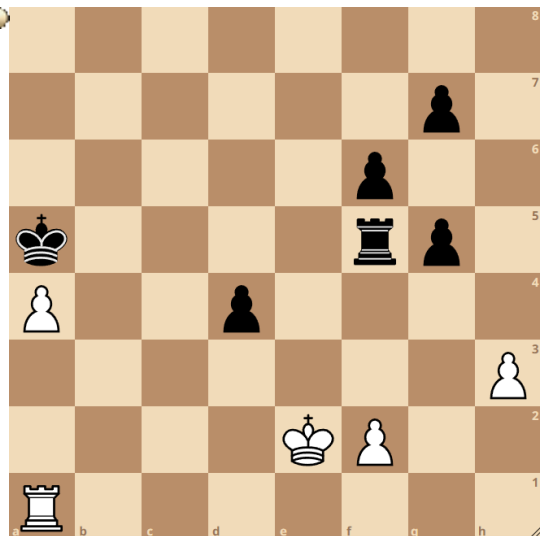
Abschließend noch drei Beispiele, welche die menschliche Überlegenheit in manchen Punkten gegenüber der Engine in real gespielten Partien demonstrieren.

Spielmann-Rubinstein, St. Petersburg 1909

Lösung: **1.Tc1** (1.Ta3? ist der Partiezug. ♟)

1...Tf4 2.Ta2 Th4 3.Kd3 Txb3+ 4.Kxd4 Th4+ 5.Kd3 Txa4 6.Te2 Tf4 7.Ke3 Kb6 8.Tc2 Kb7 9.Tc1 Ta4 10.Th1 Kc6 11.Th7 Ta7 12.Ke4 Kd6 13.Kf5 g6+! 14.Kxg6 Txb7 15.Kxh7 Ke5 16.Kg6 g4 0:1) **1...d3+ 2.Ke3 d2 3.Td1 Te5+ 4.Kd3 Td5+ 5.Kc3 Tf5 6.Txd2 Tf3+** Bis hier sind sich Mensch und Maschine noch einig. **7.Kc2!!**

Weiß muss den zweiten Bauern opfern, um Remis zu erreichen. (7.Td3? Auch ♟ Stockfish 14 begeht bei Tiefen über 52 diesen Fehler. 7...Txf2 8.Kb3 Tf5 9.Tc3 Te5 10.Td3 Te4 11.Td5+ Kb6 12.Td7 Kc6 13.Td3 13.Txg7 Th4 14.Kc2 Kc5 14.Tf3 14.Tc3+ Kd4 Tb4+ 15.Ka3 Tb7 16.a5 Tb1 17.Ka2 Tb5 18.Ta3 Kc6 19.a6 Tb8 20.a7 Ta8 21.Kb1 Kb6 22.Tb3+ Kxa7--+) **7...Txb3 8.Kd1 Kxa4 9.Ke2 Kb5 10.Td7 Th7 11.Kf3 Kc6 12.Tf7 Kd6 13.Kg4 Ke6 14.Ta7 Th4+ 15.Kg3 g6 16.Tg7 Kf5 17.f3=** (Diagramm) Schwarz kommt nicht voran.

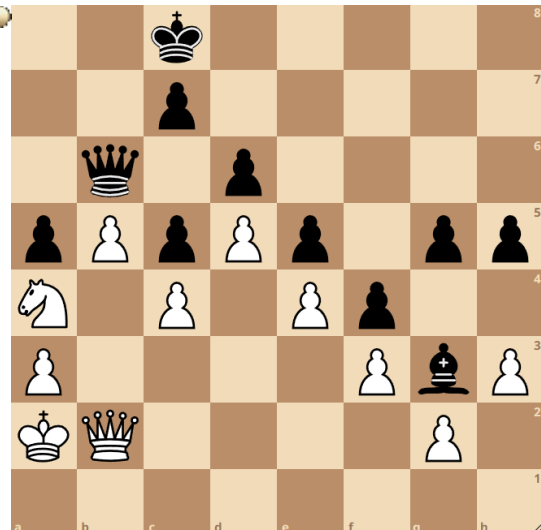


¹⁰ Die Realismusdebatte bei Motiven verschieben wir an dieser Stelle in die Ontologieabteilung der Schachwissenschaft.

Der König ist an den g6 gebunden und sobald der Th4 zieht, um g5-g4 vorzubereiten, zieht auch der weiße Turm an den Damenflügel, um nach g4 Schachs zu geben. Nach einer Tablebase von Guy Haworth ist diese Stellung bewiesenermaßen Remis.

Petrosian-Hazai, Schilde 1970

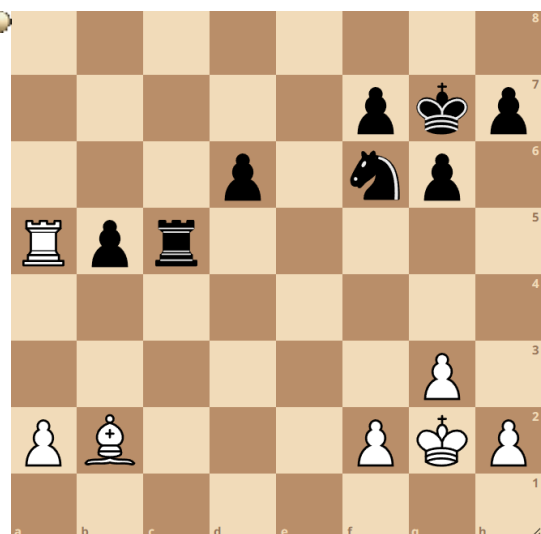
Lösung: Es liegt uns eine geschlossene Stellung vor, in der Weiß im ersten Zug die Dame gegen einen Springer gewinnen könnte. 1.Sxb6+? möchte die Engine spielen, führt jedoch zu einer Festung nach cxb6 2.h4 gxh4 3.Dc1 h3 4.gxh3 h4=. Der Gewinnplan, wie ihn Peter Prohaszka aufzeigen konnte, ist besser. **1.Db1 h4 2.Kb3 Kb7 3.Df1 Da7 4.Kc2 Db6 5.Kd3 Ka7 6.Db1 Lf2 7.Ke2 Lg3 8.Db3 Kb7**



9.Sb2 Da7 (9...a4 hilft auch nicht. 10.Dxa4 Lh2 11.Kf1 Lg3 12.Sd3 Lh2 13.Sc1 Kb8 14.Sb3 Kb7 15.Sa5+ Der Springer kommt in die gegnerische Stellung, gewinnt g5 und sorgt auf f5 dafür, dass der Läufer getauscht wird oder h4 fällt. Nach dem Läufertausch schafft Weiß einen Bauerndurchbruch an zwei Flügeln.) **10.Kf1 Lh2 11.g4 Lg3 12.Dc2 Db6 13.Sd3 Da7 14.a4 Db6 15.Dc3 Lh2 16.Kf2 Lg3+ 17.Kg2 Da7 18.Sxe5** Schwarz wir an seiner Achillessehne erwischt. **dxe5 19.Dxe5 Db6 20.Dxg5+-**

Giri, A.–Rodshtein, M., Eilat 2012

Die Engine möchte bei einer Tiefe von 43 im Diagramm zu Giri–Rodshtein 34.h4 spielen und rechnet weißem Vorteil nach einigem Getausche ins Turmendspiel. Für Giri war die Situation an dieser Stelle deutlich klarer, die richtige Idee gewann die Partie sofort.



Lösung: **34.a4! bxa4** (34...b4 35.Txc5 dxc5 36.Kf3 c4 37.Ld4 c3 38.Ke2 c2 39.Kd2+– Die Freibauern sind aufgehalten und der A-Bauer gewinnt.) **35.Txc5 dxc5** Weiß verfügt über eine ewige Fesslung. Sobald sich der schwarze König entfernt, fällt der Springer und Weiß gewinnt mit Mehrfigur. **36.f4 a3** (36...h6 holt die Kuh nicht mehr vom Eis, z.B. 37.Kf3 g5 38.f5+–) **37.La1+–**

Trotz der Beispiele kann man festhalten, dass die Engines sehr schnell besser wurden und werden. Bei der Recherche fand ich einige YouTube-Videos von 2020 oder später, die angeblich durch Engines unlösbare Aufgaben vorstellten. Die meisten wurden allerdings bei meiner Prüfung durch Stockfish 14 sehr wohl gelöst (häufig bereits durch gespeicherte Clouddaten). Es ist deshalb nicht auszuschließen, dass einige der Beispiele hier in der Zukunft mit einer noch leistungstärkeren Variantenberechnung oder einem größeren neuronalen Netz, welches zu einer verfeinerten „Intuition“ führt, durch Engines gelöst werden.

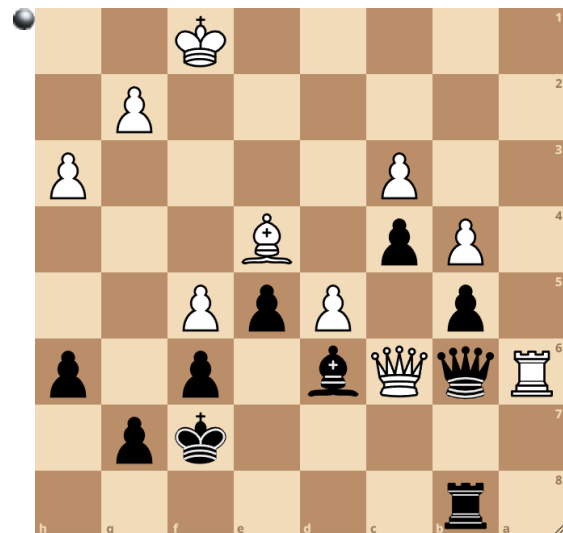
3.1.2 Mensch gegen Maschine

Das wohl bekannteste Duell zwischen einem menschlichen Großmeister und einem Computerprogramm ist das zwischen dem damaligen Weltmeister Garry Kasparow und IBMs DeepBlue 1997 in New York. Dieses Match ist deshalb so besonders, weil erstmals die Maschine unter Turnierbedingungen mit 3,5–2,5 den Sieg gegen einen Weltmeister davontragen konnte.

Kasparow, der im Jahr zuvor die Blechbüchse noch trotz Gegenwehr mit 4–2 in die Schranken gewiesen hatte, kämpfte u.a. mit einigen psychologischen Herausforderungen. Noch nie hatte ein Computer einen Weltmeister geschlagen, Schach schien zu kompliziert und das Monster zu beschränkt. Kasparow war zweifelsohne der beste menschliche Spieler zu seiner Zeit und hatte einen Ruf zu verlieren.

Andererseits war der Gegner bzw. dessen Verbesserung seit dem letzten Match nicht richtig einzuschätzen. Im Vorfeld war klar, dass die Maschine 200 Millionen Stellungen pro Sekunde berechnen konnte, eine für den Menschen unvorstellbare Leistung. War Schach kurz davor gelöst zu werden? Trat man überhaupt gegen einen schlagbaren Gegner an? Welche Schwächen genau hat der Gegner? Diese Fragen stellte sich die Alpha-Beta-Engine nicht, für Kasparow waren sie aber sicherlich relevant und beeinflussten sein Spiel mindestens unterbewusst.

Nachdem der Weltmeister die erste Partie (wohl auch aufgrund eines Bugs des Gegners) gewonnen hatte,¹¹ folgte in Runde 2 diese Stellung aus der Smyslow-Variante im Spanier. Was sollte Schwarz ziehen?



Lösung: Kasparow gab in dieser Stellung auf. Kurz zuvor war die Engine auf Db6+ mit dem König nach f1 gegangen, was für den Menschen in erster Linie kontraintuitiv war. Ob Kasparow davon in seiner Aufgabe beeinflusst wurde und dachte, dass die Engine gesehen hätte, dass Schwarz in dieser Stellung sowieso nichts mehr rettet, ist nicht auszuschließen. Sicher ist aber, dass nach **1...De3 2.Dxd6 Te8** Weiß die Figur zurückgeben muss, da z.B. nach 3.Lf3 ein Dauerschach folgt. **3.Dd7+** (3.h4 h5= erzeugt ebenfalls einen Dauerschachkäfig.) **3...Te7 4.Dc6 Dxe4 5.d6** Das resultierende Endspiel mit guten schwarzen Remischancen ist im Detail analysiert in Müller/Schaeffer.

Nach der Partie bezichtigte Kasparow IBM des Betrugs, indem er behauptete, ein starker Mensch hätte der Engine geholfen. In Anblick heutiger Cheatingbezichtigungen ein interessanter Vorwurf, wobei dieser Streitfall hier nicht aufgerollt werden soll.

In der dritten Partie spielte Kasparow 1.d3, um den Computer aus seinem Eröffnungsbuch zu bringen. Er brach dabei aber auch mit seinem eigentlichen aktiven Angriffsstil, die Partie endete Remis. Partie 4 und 5 brachten Kasparow sehr aussichtsreiche Stellungen, die er aber einmal aufgrund von Zeitnot und einmal wegen perfekter Verteidigung DeepBlues im Endspiel nicht verwerten konnte.

¹¹ Wer sich für die einzelnen Partien im Match interessiert, dem seien die Videoserie von Georgios Souleidis auf dessen YouTube-Kanal (www.youtube.com/watch?v=E1qatUrxpHs&list=PLXoFpqP9Yfb1ICChsu6EwMi3cfmrkviPo) und Müller/Schaeffer ans Herz gelegt.

Partie 6 sollte die Entscheidung bringen, Kasparow wählte eine Nebenvariante im Caro-Kann: 1.e4 c6 2.d4 d5 3.Sc3 dxe4 4.Sxe4 Sd7 5.Sg5 Sgf6 6.Ld3 e6 7.S1f3 h6?!



Lösung: Kasparow hatte darauf gesetzt, dass die materialistische Maschine sich das scharfe 8.Sxe6 nicht traute, da es keinen konkreten Vorteil in absehbarer Zeit bot.

8.Sxe6! De7?! (8...fxe6 9.Lg6+ Ke7 10.O-O Dc7 11.Te1=∞ Weiß spielt gegen den schwachen schwarzen König und hat Entwicklungsvorsprung. Objektiv besser als die Partiefortsetzung, aber auch nicht leicht zu spielen für Schwarz, gerade gegen eine Engine.) 9.O-O fxe6 10.Lg6+ Kd8 11.Lf4 b5 12.a4 Lb7 13.Te1 Sd5 14.Lg3 Kc8 15.axb5 cxb5 16.Dd3 Lc6 (16...a6 17.c4+– Weiß öffnet die Stellung und erhält einen verheerenden Mattangriff.) 17.Lf5 exf5 18.Txe7 Lxe7 19.c4 1:0

Zusammenfassend kann man sagen, dass Kasparows Anti-Computer-Strategie nicht wirklich aufging. Er versuchte zwar, den Gegner aus dessen bekannten Mustern zu locken, geriet dabei aber auch selbst in Stellungen, die nicht seinem Stil entsprachen oder schlichtweg nicht gut waren.

Ähnlich wie Kasparow, aber erfolgreicher ging Kramnik 2002 in Manama an den Kampf gegen die Engine Deep Fritz mit einer eigenen Strategie heran. Team Mensch hatte in der Vorbereitung Stellungen aus Partien gesammelt, in denen Fritz gegen andere Engines verloren hatte, um generelle Schwächen festzustellen. Kramnik behandelte Fritz eher wie einen guten Taktiker und schlechten Strategen und nutzte gezielt die Schwächen aus, wie er es vielleicht auch bei einem Menschen getan hätte.

In Partie 1 erzielte Kramnik aus einem Berliner Endspiel mit Schwarz ein solides Remis, auch in Partie 2 begann schnell die dritte Phase:¹²

1.d4 d5 2.c4 dxc4 3.Sf3 Sf6 4.e3 e6 5.Lxc4 c5 6.O-O a6 7.dxc5 Dxd1 8.Txd1 Lxc5 9.Kf1 (Diagramm) Kein notwendiger Zug, bringt Fritz aber vielleicht aus seinem Eröffnungsbuch. Kramnik suchte sich eine Variante mit einem strategisch



anspruchsvollen Endspielen heraus. Eine seiner Stärken und Fritz Schwächen. **9...b5?** Ermöglicht Weiß den Hebel a4. **10.Le2 Lb7 11.Sbd2 Sbd7 12.Sb3 Lf8?!** Sehr kontraintuitiv, Fritz offenbart seine

Schwächen. **13.a4! b4 14.Sfd2! Ld5 15.f3 Ld6 16.g3 e5 17.e4 Le6 18.Sc4 Lc7 19.Le3 a5 20.Sc5 Sxc5 21.Lxc5 Sd7 22.Sd6+ Kf8 23.Lf2?!** (23.Sb5+! Sxc5 24.Sxc7 Tc8 25.Tac1 Lh3+ 26.Kg1 Sb3 27.La6+- hätte noch kraftvoller eine strategische Gewinnstellung herbeigeführt.) **23...Lxd6 24.Txd6 Ke7 25.Tad1 Thc8 26.Lb5 Sc5!** (Diagramm)

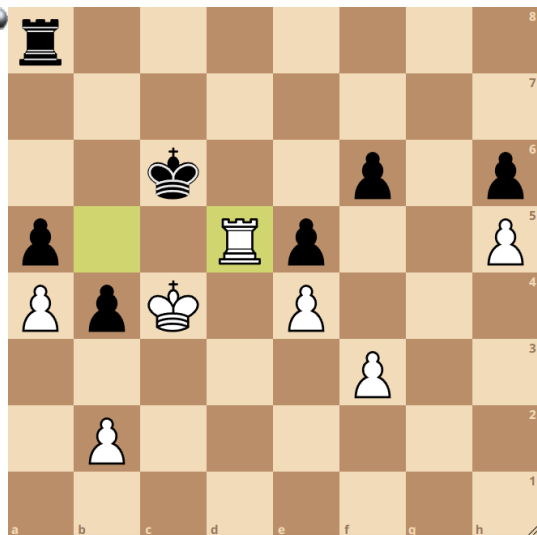


Funktioniert taktisch, wie bei der Engine zu erwarten. **27.Lc6 Lc4+ 28.Ke1 Sd3+ 29.T1xd3 Lxd3 30.Lc5! Lc4 31.Td4+ Kf6 32.Txc4 Txc6 33.Le7+ Kxe7 34.Txc6** Weiß erhält ein besseres Turmendspiel durch den aktiveren Turm. **34...Kd7 35.Tc5 f6?** (35...Tc8! Schwarz muss auf aktive Verteidigung setzen. 36.Txc8 [36.Txa5 Tc2=] 36...Kxc8 37.Kd2 Kd7 38.Kd3 Kd6 39.Kc4 Kc6 Weiß kann das Tempoduell nicht gewinnen, z.B.: 40.h4 g6 41.g4 h5 42.g5 Kd6 43.Kb5 f6 44.gxf6 g5 45.hxg5 Ke6 46.Kxa5 h4 47.f7 Kxf7 48.Kb6 h3 49.a5 h2 50.a6 h1D 51.a7=)

36.Kd2 Kd6 37.Td5+ Kc6 38.Kd3 g6?! 39.Kc4 g5 40.h3 h6 41.h4 gxh4 42.gxh4 Ta7 43.h5 Ta8 44.Tc5+ 44...Kb6 45.Tb5+ Kc6 46.Td5+- (Diagramm nächste Seite)

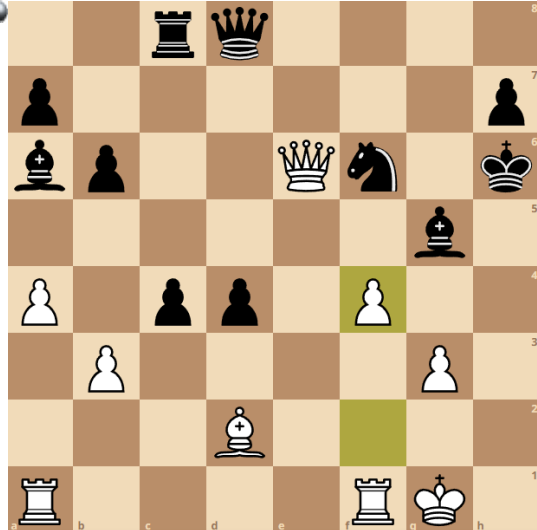
¹² Für das weitere Studium der einzelnen Wettkampfpartien empfehle ich Müller: Deep Blue.

Zugzwang! 46...Kc7 47.Kb5 b3 48.Td3
Ta7 49.Txb3 Tb7+ 50.Kc4 Ta7 51.Tb5
Ta8 52.Kd5 Ta6 53.Tc5+ Kd7 54.b3 Td6+
55.Kc4 Td4+ 56.Kc3 Td1 57.Td5+ 1:0



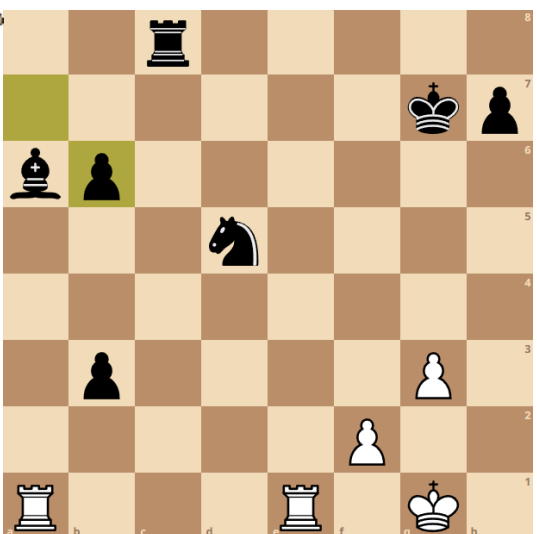
Wieder in einem strategischen Endspiel holte Kramnik gegen Fritz einen weiteren vollen Punkt in Partie 3, dieses Mal sogar mit Schwarz. Die folgende Partie ging Remis aus, in der fünften Runde erlebte Kramnik einen absoluten Blackout und verlor durch einen zweizügigen Figureneinsteller.

Partie 6 wurde wilder und Kramnik spielte ein Opfer im Stile Tals. Es war also inkorrekt und die Engine bestrafte es. In einer Eventualvariante hatte Kramnik einen wichtigen Schlüsselzug übersehen (Diagramm)



Lösung: Lh4!!-+ Weiß kann nicht nehmen, da nach Dg8+ Schwerfiguren getauscht werden und Schwarz das Endspiel gewinnt. Stattdessen wich Kramnik in eine Variante aus, die ein schlechteres Endspiel nach sich zog.

Die Maschine zeigte in dieser Partie jedoch eine Schwäche, die bereits aus den Beispielen des vorherigen Kapitels bekannt ist. (Diagramm)

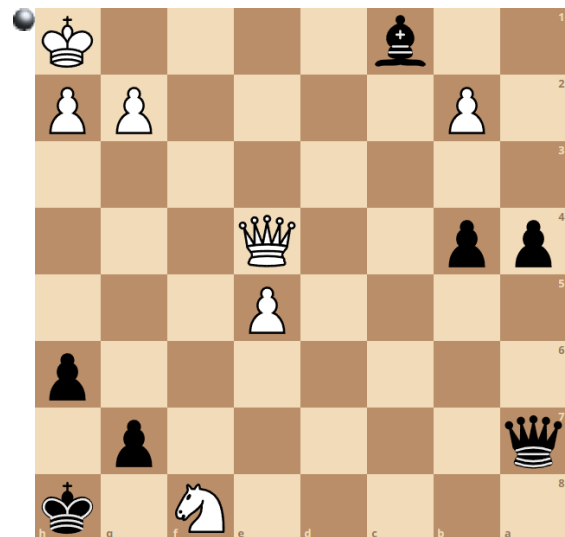


Lösung: Kramnik gab in dieser objektiv remisen Stellung auf. Auch Stockfish 14 tut sich schwer, die Bewertung anzuerkennen, da wir wieder auf eine langfristige Festung zusteuern. **1.Txa6 b2 2.Ta7+ Kg6 3.Td7 Tc1 4.Td6+ Sf6 5.Tdd1 b1D 6.Txc1 Df5 7.Tc6! b5 8.Tee6 b4 9.Tb6 h5 10.Txf6+ Dxf6 11.Txb4=** Der Turm pendelt zwischen f4 und h4 und Schwarz kommt nicht durch.

Trotz der noch komplizierteren psychologischen Situation erreichte Kramnik in den letzten beiden Partien solide Remisen und das Match endete 4:4. Die Revanche gab es 2006 in Bonn.

Die verbesserte Fritz-Version zeigte weiterhin Schwächen, die allerdings weniger schwerwiegend waren als die der verbesserten Kramnik-Version. Im Auftaktduell gelang es Kramnik, aus dem Katalanen (übrigens mindestens damals eine gute Eröffnungswahl gegen Engines, da häufig strategische Endspiele entstehen) ein strategisches Endspiel herbeizuführen. Ein wenig ungenaues Spiel und gute Verteidigung Fritzchens führten zum Remis.

In Runde 2 zeigte Kramnik seine menschliche Seite. Nach eigentlich gelungener Anti-Computer-Eröffnung gelangte er in ein Endspiel und zu dieser Stellung:

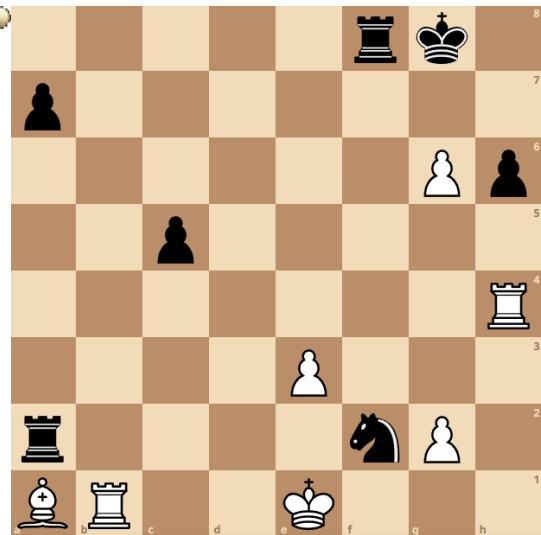


Lösung: Kramnik griff mit **34...De3??** fürchterlich daneben, wonach die Partie durch Matt entschieden wurde. Stattdessen hätte **34...Kg8 35.Sg6 De3 36.Dd5+ Kh7 37.Sf8+ Kh8** die Stellung im Remis enden lassen.

In der dritten Partie spielte Fritz stark gegen den Katalanen, Kramnik schaufelte sich durch eine Festung ins Remis. Dem Russen gelang auch im Russen der Folgerunde ein Remis auf beiderseitig hohem Niveau. Partie 5 brachte Kramnik mit Weiß in eine unangenehm aussehende Situation, die dieser aber weltmeisterlich behandelte (Diagramm nächste Seite)

Lösung: **Kf1!!** Der einzige Zug. Schwarz sah nichts Besseres, als mit **Sh3+ Ke1 Sf2** ein Remis zu forcieren.

Damit musste Kramnik die letzte Partie mit Schwarz gewinnen, um ein Unentschieden zu erzielen. Er wählte eine zweischneidige Eröffnung und in einer schönen Matchpartie holte Fritz schließlich den zweiten vollen Punkt und gewann somit das Duell.



Der Kramnik-Ansatz im Anti-Computer-Schach seiner Zeit war ausgefeilter als der von Kasparow. Eventuell, da der Gegner besser bekannt war. Der Stil des Exweltmeisters passte aber auch besser zu den Schwächen des Apparats, was beim Taktiker Kasparow nicht der Fall gewesen war. Trotz der Chancen zeichnete sich in den Kämpfen jedoch bereits ab, dass die Engines an den Menschen vorbeiziehen würden.

Eine sehr erfolgreiche Anti-Computer-Strategie zeigte sich in einer 3-Minuten-Blitzpartie (!) zwischen Hikaru Nakamura und Rybka im Jahre 2008. Der wesentliche Punkt der folgenden eindrucksvollen Partie kann verstanden werden, indem nur die Diagramme und Kommentare betrachtet werden, da viele Züge keine wesentliche Veränderung der Stellung bedeuten. Es sind dennoch alle Züge angegeben, damit die Langatmigkeit besser spürbar wird.

1.g4 d5 2.h3 h5 3.g5 g6 4.d4 Lg7 5.Sc3 c6 6.Sf3 Lf5 7.Sh4 e6 8.Sxf5 exf5 9.Lf4 Se7 10.e3 O-O 11.Dd2 Sd7 12.O-O-O Te8 13.Lg2 Sf8 14.Le5 Se6 15.f4 Sc8 16.Kb1 Sd6 17.Lf1 Dd7 18.a4 b6 19.La6 Sc7 20.Le2 a6 21.Ld3 b5 22.h4 Se6 23.Se2 Te7 24.Ka1 Tee8 25.Da5 Sb7 26.Dd2 Sd6 27.Th2 Lf8 28.Tf2 Sg7 29.a5 Sb7 30.Sg1 Dd8 31.b4 Dd7 32.Sf3 Sd6 33.Lxd6 Lxd6 34.Se5 Db7 35.Kb2 Tac8 36.c3=
(Diagramm)



Nakamura legte es nicht darauf an, Rybka auf taktische Weise in der Eröffnung zu schlagen. Vielmehr visierte er eine festgefahrene Struktur an, die wir jetzt mit acht gegen acht Bauern erreichen. Wenn eine Seite Gewinnambitionen hat, sollte normalerweise versucht werden, eine möglichst gute Aufstellung der eigenen Figuren zu finden, um die Struktur zum richtigen Zeitpunkt durch Opfer zu öffnen. Der Mensch Nakamura versucht nicht, eine solche Aufstellung zu finden bzw. sieht, dass es kein Durchkommen gibt. Stattdessen entscheidet er sich, abzuwarten. Objektiv ist das die richtige Entscheidung.

36...Te7 37.Le2 Tee8 38.Lf3 Te7 39.Dc2
Td8 40.Tfd2 Lb8 41.Le2 Te6 42.Ld3 Te7
43.Tf2 Te6 44.Dd2 Te7 45.Tg1 Te6 46.Tff1
Te7 47.Dd1 Te6 48.Tg2 Te7 49.Dc2 Te6
50.Tfg1 Te7 51.Tf2 Te6 52.De2 Te7 53.Df3
Te6 54.Td2 Te7 55.Tgd1 Te6 56.Dg3 Te7
57.Kb1 Te6 58.Lc2 Te7 59.Sd3 Te6 60.Sc5
Dc8 61.Sxe6 Dxe6= (Diagramm)



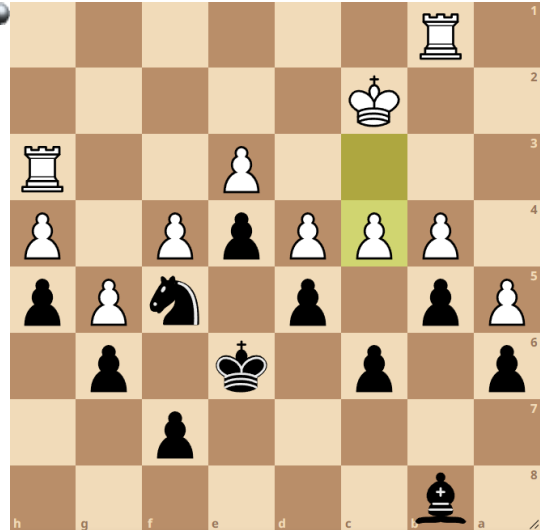
Selbst der Qualitätsverlust ändert nichts am
Remis bei bestmöglichem Spiel beider

Seiten. 62.Kb2 Te8 63.Te1 Dd7 64.Df3 Te7 65. Ld3 Te8 66.Tg2 Te7 67.Kb3 Te8
68.Ta2 Te7 69.Tc1 Te8 70.Tca1 Te7 71.Tg1 Te8 72.Th2 Te7 73.Dg3 Te8 74.Ka3 Te7
75.Tb1 Te8 76.Td2 Te7 77.Kb3 Te8 78.Ta1 Te7 79.Kb2 Te8 80.Te2 Te7 81.Tee1 Te6
82.Df3 De7 83.Df2 De8 84.Le2 De7 85.Ld1 De8 86.Lf3 De7 87.Dd2 De8 88.Kb3 De7
89.Kc2 De8 90.Kb2 De7 91.Tab1 Kf8 92.Df2 Ke8 93.Le2 Kd8 94.Ld3 De8 95.Df3
Ke7 96.Kb3 Kf8 97.Ta1 De7 98.Lc2 Kg8 99.Kb2 De8 100.Df2 De7 101.Dd2 De8
102.Ld3 De7 103.Le2 De8 104.Tad1 De7
105.Tc1 De8 106.Lf3 De7 107.Df2 De8
108.Tcd1 De7 109.Le2 Te4 110.Ld3 De6
111.Lxe4 fxe4= (Diagramm)



Nakamura opfert eine zweite Qualität.
Dadurch verbessern sich objektiv zwar
nicht die Gewinnaussichten, allerdings
bewertet die Engine das gewonnene
Material über. Auch Stockfish 14 urteilt bei
einer Tiefe von 33 mit +1,4.

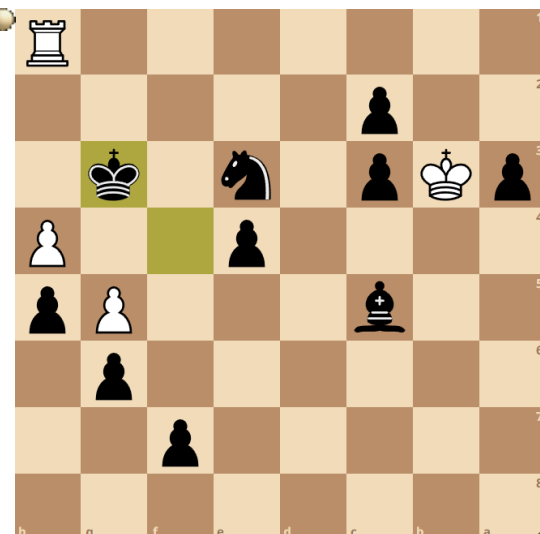
112.Ta1 Sf5 113.Kb3 Kf8 114.Th1 Ke8 115.Th2 Kd7 116.De1 Kd8 117.Td1 Kd7
 118.Tc2 Kd8 119.Df2 Sd6 120.Ka2 Df5 121.Th1 Kd7 122.Dg3 Ke6 123.Dh3 Ke7
 124.Dxf5 Sxf5 125.Th3 Ke6 126.Kb3 Lc7 127.Te2 Lb8 128.Te1 Lc7 129.Ta1 Lb8
 130.Td1 Lc7 131.Te1 Lb8 132.Te2 Lc7 133.Tg2 Lb8 134.Tg1 Lc7 135.Tb1 Lb8
 136.Tbh1 Lc7 137.Tg1 Lb8 138.Kb2 Lc7 139.Kc2 Lb8 140.Kd2 Lc7 141.Th2 Lb8
 142.Tc1 Lc7 143.Ta1 Lb8 144.Ke2 Lc7 145.Tg1 Lb8 146.Thh1 Lc7 147.Kd2 Lb8
 148.Tc1 Lc7 149.Th3 Lb8 150.Ta1 Lc7
 151.Thh1 Lb8 152.Tae1 Lc7 153.Th2 Lb8
 154.Th3 Lc7 155.Ta1 Lb8 156.Kc2 Lc7
 157.Tf1 Lb8 158.Kb3 Lc7 159.Tf2 Lb8
 160.Kc2 Lc7 161.Tg2 Lb8 162.Kd2 Lc7
 163.Th1 Lb8 164.Tf2 Lc7 165.Tfh2 Lb8
 166.Td1 Lc7 167.Tf1 Lb8 168.Te1 Lc7
 169.Ke2 Lb8 170.Th3 Lc7 171.Kd2 Lb8
 172.Kc2 Lc7 173.Tb1 Lb8 174.c4=



(Diagramm) Rybka versucht auf Biegen

und Brechen die scheinbare Gewinnstellung nicht durch die 50-Züge-Regel Remis
 ausgehen zu lassen. 174...dxc4 175.Kb2 Ld6 176.Ka3 Se7 177.Tb2 Sd5 178.Tg3 Kf5
 179.Tb1 Le7 180.Th3?? Rybka dreht komplett durch! 180.Tb2 hätte keinen Fortschritt
 bedeutet und somit trotz „besserer“ Stellung ein Remis forciert. Das Verhalten des
 Computers ähnelt dem menschlichen Frustverhalten, das manchmal ebenfalls eine
 (scheinbar) bessere Stellung schnell gewinnen möchte und diese bis zum Verlust
 überzieht. 180...Ld6 (180...Kg4 hätte hier bereits gewonnen. Z.B. 181.Th2 Ld6 182.Te2
 Kf3—+ Die weißen Bauern fallen und die schwarzen Freibauern werden übermächtig.)

181.Kb2?? (181.Rg3= Weiß müsste weiter
 abwarten und sich mit Remis
 zufriedengeben.) 181...Le7 182.Te1 Lxb4
 183.Te2 Lxa5 184.Kc1 Lb6 185.Kc2 a5
 186.Th1 a4 187.Kc1 a3 188.Tc2 La5
 189.Th3 Lb4 190.Te2 Ld6 191.Te1 b4
 192.Kb1 b3 193.Th2 c3 194.Tc2 Lb4
 195.Ka1 bxc2 196.Ka2 Sxe3 197.Kb3
 Kxf4 198.Tc1 c5 199.dxc5 Lxc5 200.Th1
 Kg3 (Diagramm)



Die Partie dauerte noch weitere 70 Züge, da Nakamura den hilflosen weißen König mit fünf Läufern durch die Manege jagte, um ihn dann in einer Ecke in Embryonalstellung kauern Matt zu setzen.¹³

Die drei ausschnittshaft betrachteten Matches und die Partie von Nakamura sollen veranschaulichen, wie stark sich die Engines im Laufe der Zeit steigerten, aber auch wie die Anti-Engine-Strategien sich entwickelten.

Abschließend sei hinsichtlich möglicher zukünftiger Matches zwischen Mensch und Maschine auf den Ausblick verwiesen, den Shakhriyar Mamedyarov während der abschließenden Pressekonferenz in Wijk aan Zee gab. Er hatte zuvor einen sehr starken zweiten Platz hinter Magnus Carlsen belegt und antwortete auf die Frage nach seiner Lieblingspartie im Turnier:

Seine favorisierte Partie sei seine einzige Verlustpartie. In dieser hätte Carlsen brillant gespielt und ihm keine Gegenmöglichkeiten gelassen. Er fuhr fort: „Vor dem Turnier dachte ich, die Menschen hätten keine Chance gegen eine Engine, aber jetzt schätze ich, es gibt eine gewisse Chance (...)“.¹⁴

Ob ein solches Match tatsächlich stattfinden wird oder ob Carlsen auch nur den Hauch einer Chance hätte, ist zum jetzigen Zeitpunkt natürlich reine Spekulation. Wenn jedoch ein Mensch Chancen gegen eine scheinbar übermächtige Maschine haben könnte, dann vermutlich ein Spieler wie Carlsen, der als Reflektor¹⁵ nicht einfach den pragmatischen „Stil“ einer Alpha-Beta-Engine in schlechterer Version widerspiegelt.¹⁶ Interessant wären meiner Meinung nach eher Konstellationen, in denen Carlsen mit Hilfe von Stockfish gegen AlphaZero spielt o.Ä.

13 Für die Kommentierung der Partie von Nakamura selbst vgl. www.youtube.com/watch?v=-UqCAIQEw2E.

14 Vgl. www.youtube.com/watch?v=S9_IqOpc-FY.

15 Grob vereinfacht zeichnet sich ein Reflektor durch intuitives und strategisch geprägtes Spiel aus. Für die Theorie verschiedener Spielertypen nach Lars Bo Hansen vgl. z.B. Müller/Engel.

16 Ob eine Engine mit neuronalem Netz auch als Pragmatiker oder doch eher als Reflektor gilt, kann an anderer Stelle diskutiert werden. Sicher ist, dass auch sie starke pragmatische Fähigkeiten aufzeigt.

3.2 Der sinnvolle Gebrauch

Die Engines haben Schach entmythologisiert. Alle Tal-Partien müssen schmerzhaft als fehlerhaft eingesehen und als unnütz abgestempelt werden. Das waren zwar keine Fragen, ich beantworte sie dennoch mit einem klaren Nein!

3.2.1 In der Forschung

Das Ziel des gemeinen Turnierspielers ist in der Regel der Gewinn von Partien. Ein wenig anders ist das bei vielen Fernschachspielern, denen neben dem Gewinn einer Partie vor allem die wissenschaftliche Wahrheit über eine Stellung am Herzen liegt. Entsprechend (und weil es anderweitig schwierig zu kontrollieren ist) wird in diversen Fernschachformaten Computerunterstützung erlaubt. Es geht also darum, die Engine gut zu bedienen und es werden selbstredend doch noch Partien entschieden – wenn auch deutlich weniger als auf dem vergleichbaren Durchschnittsniveau im Normalschach.

Schnittstellen von Fern- und Nahschach finden sich häufig in der Eröffnungsarbeit. Ein möglichst gutes Verständnis von Stellungstypen und davon ausgehenden Mittel- und Endspielen, hat im Schach immer Vorteile. Die Engine kann jedoch keine Pläne erklären oder Spielstile unterscheiden.

Wenn die Engine in einer Stellung Ausgleich auf vier verschiedene Weisen anzeigt, liegt es am Menschen zu interpretieren und die richtigen Varianten tiefer zu analysieren. Eine Variante führt vielleicht eher in taktische Stellungen mit beiderseitigem Angriff, eine andere in ein langweiliges Endspiel. Jeder Fleischkopf sollte sich hier auch an seinen eigenen Vorlieben und Fähigkeiten orientieren, allein 0,0 sagt nichts aus.

Bei der Bewertung gilt es auch zu beachten, was wir zu Beginn bzgl. der Tendenz zur Wahrheit erwähnt hatten. Im Schach gibt es nur drei Ergebnisse: Sieg, Remis und Sieg des Gegners. Ob eine Stellung mit +0,3 oder +0,4 bewertet wird, ist für den Menschen in der Regel nicht nachvollziehbar und sollte auf keinen Fall alleine die Wahl einer Variante beeinflussen. Auch ist eine gewonnene +4-Stellung nicht gewonnener als eine gewonnene +15,2-Stellung.

In einem Interview mit Fernschach-Großmeister Arno Nickel schlägt dieser folgendes Schema vor, um sinnvoll mit der Engine eine Stellung zu analysieren:

- „a) Ist meine „Strategie“ wirklich stellungsgemäss oder muss ich sie ändern? Muss ich sie grundlegend ändern oder nur modifizieren?
- b) Treffen die Stellungsbewertungen der Engine(s) zu? Welche Aussagekraft haben sie? Sind Bewertungsunterschiede zwischen einzelnen Varianten relevant oder nicht?
- c) Ist mein taktisches Vorgehen richtig?
- d) Wie gut „versteht“ das Schachprogramm die Stellung – wie gut verstehe ich selbst sie?“¹⁷

Der Mensch sollte die Engine nicht vergöttern, sondern hinterfragen. Wie bereits erwähnt versteht er manchmal Zusammenhänge sogar besser als die Engine oder – was bei Nichtgroßmeistern häufiger der Fall ist – er wird eines Besseren belehrt. Im Idealfall passt er sein Schachverständnis daraufhin ob seiner neuen Interpretation an und wird zu einem besseren Spieler. Dieser menschliche Lernvorgang ist verglichen mit dem AlphaZeros noch effizienter, da durch die komplexe menschliche Fähigkeit zu abstrahieren nicht erst Millionen Partien gespielt werden müssen.

Es kann bei einer professionellen Analyse nicht nur darum gehen, den besten Zug zu finden, sondern darum, die Stellung zu verstehen. In Studien offenbart sich im Verständnis der Stellung direkt die Lösung, aber in Partien stellen sich die meisten Positionen nicht als so klar lösbar dar. Es gilt, im Gespräch mit seinem digitalen Sekundanten die richtigen Fragen zu stellen.

Wie auch menschliche Großmeister haben Engines individuelle Unterschiede. Fritz kann auf den ersten Blick zu einem anderen Ergebnis kommen als Komodo, während Shredder, Leela und Houdini wieder andere Meinungen haben, obwohl sie teilweise dieselben technischen Ansätze verfolgen.

Außerdem ist die Leistungsfähigkeit immer von der Bedenkzeit und der Hardware abhängig. In einem Gespräch mit Ex-Weltmeister Rustam Kasimjanow erklärte mir dieser, dass Spieler der Weltspitze heutzutage fast nur noch auf Cloudengines zurückgreifen, also nicht mehr das eigene Endgerät für die aufwändige Rechenarbeit benutzen. Diese sind z.B. über Chessbase für wenige Dukaten mietbar.

¹⁷ Vgl. www.glarean-magazin.ch/2010/12/14/interview-mit-fernschach-grossmeister-arno-nickel/.

Das extrem hohe Niveau guter Fernschachpartien bietet natürlich auch einen reichhaltigen Schatz für das Nahschach. Ein Beispiel aus meiner favorisierten Variante gegen Slawisch, dem Boor-Angriff: **1.d4 d5 2.c4 c6 3.xd5 xd5 4.Sc3 Sf6 5.f3 Db6 7.e4 xe4 8.xe4 e5 9.Lb5+ Sc6 10.d5 Lb4 11.Dd3 a6 12.La4 Da5 13.Ld1 Sd4** (Diagramm) An dieser Stelle spielte ich lange Zeit mit Hilfe der wenigen Nahpartien und der Engine 14.Sf3 und musste mich häufig in unangenehmen Stellungen



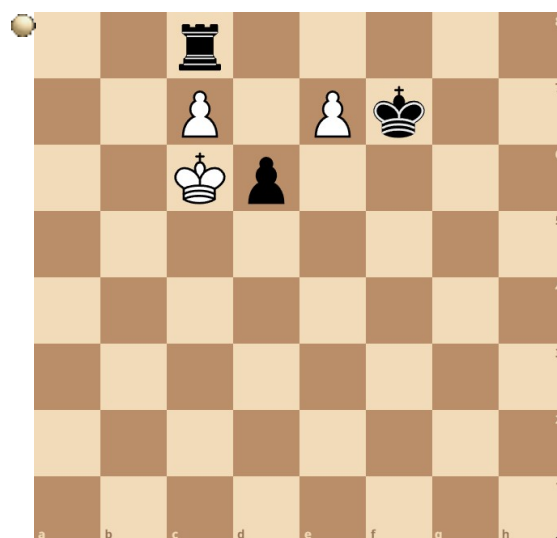
wiederfinden. Mein ehemaliger Vereinskollege Burkhard Treiber wies mich auf Fernschachpartien hin, in denen der Zug **13.Sge2!?** gespielt wurde. Meine damalige Engine (Stockfish 13) echaufferte sich einige Minuten und bespuckte mich dann mit brutalen Bewertungen, die allesamt den Anschein hatten, dass Schwarz gewinnen würde. Sie gab auf freundliche Nachfrage schnippisch die Variante **13...Sxe2 14.Lxe2 Sxe4 15.Dxe4 Lxc3+ 16.bxc3 Dxc3+ 17.Kf2 Dxa1** an. Noch während ich die Züge ausführte, merkte die Engine jedoch langsam, dass die Lage gar nicht so schlecht war wie anfangs vermutet. Sie half mir daraufhin sogar noch, mit **18.Le3!** das zweite Materialopfer zum Rollen zu bringen. So vertrugen wir uns wieder und erlebten gemeinsam unzählige Abenteuer in dieser netten Variante. Die Analysen sind zu verstreut und variantenreich, um sie gänzlich hier abzubilden, der Punkt sollte auch so klar sein. Stockfish 14 hat übrigens weniger Probleme mit der 13.Sge2-Variante.

Die Engine erleichtert es den Anwendern objektiv, einen Zugang zur Wahrheit zu finden. Das erleichtert jedoch nicht zwangsläufig die Arbeit generell oder verkürzt den Aufwand. In der Analyse einfach eine Engine anzuschalten und die Züge hinzunehmen oder in der Vorbereitung ihre Vorschläge auswendig zu lernen, bildet die falsche – wenn auch oft gesehene – Herangehensweise.

3.2.2 Im Training

Die Art, wie Engines für die Variantenberechnung programmiert werden, kann auf den Menschen umgeschrieben werden bzw. stammt von ihm ab. AlphaZeros Schlüsse, die es aus Simulationen zieht, ähneln menschlichem Lernen. Systeme und Schemata für die Variantenberechnung gehören als Orientierung auch in das menschliche Training. Als Beispiel Diagramm 9:

Diagramm 9



Lösung: A) Stellungsbewertung: Schwarz kann nach jedem möglichen weißen Zug Remis sichern, indem er $Txc7\ Kxc7\ Kxe7$ spielt. Da Weiß nicht mehr gewinnen wird, geht es also darum, Remis zu erreichen. B) Kandidatenzüge: $1.Kxd6$, $1.Kd7$, $1.Kb7$ und $1.e8D+$ fallen in den Blick. C) Zuerst berechnen wir die Züge, von denen wir denken, dass sie ausgeschlossen werden können, ohne viele Varianten zuzulassen. Häufig sind dies Schlag- und Schachzüge. ($1.Kxd6?$ $Ke8+$ Weiß gerät in Zugzwang und verliert beide Bauern.) ($1.Kd7?$ $Txc7+$ $2.Kxc7\ Kxe7+$ Schwarz gewinnt das Bauernendspiel.) ($1.Kb7\ Txc7+$ $2.Kxc7+$ Zugumstellung zu $1.Kd7$) Im Idealfall haben wir die anderen Kandidaten zuerst berechnet, sodass $1.e8D+$ ohne Berechnung und ohne zu wissen, ob dieser Zug Remis hält, D) gezogen werden kann. Entweder ist die Stellung ohnehin verloren oder wir ziehen den einzigen Zug. $1...Kxe8$ ($1...Txe8\ 2.Kxd6\ Te6+\ 3.Kd7=$ Schwarz muss Schachs geben oder den Bauern schlagen, damit dieser sich nicht umwandelt.) $2.Kxd6\ Ta8\ 3.Kc6\ Tc8\ 4.Kd6=$ Schwarz erzielt keine Fortschritte.

Darf ich keine Tal-Partien mehr zeigen? Doch, wenn sie wahrheitsgemäß wiedergegeben werden. In der Theorie ist der Gegner immer Carlsen und wir müssen mit den besten Zügen rechnen. Da wir aber eher selten gegen Carlsen spielen, dürfen wir uns auf der anderen Seite freuen, wenn der Gegner dies unter Beweis stellt. Wenn also erarbeitet wird, warum die Opfer inkorrekt sind, stellen auch objektiv nicht perfekte Partien kein Hindernis für einen guten Unterricht dar.¹⁸

¹⁸ Für viele Zuschauer war Carlens erste Gewinnpartie im WM-Kampf gegen Nepomniatchi die beste, weil der Weltmeister seine Klasse unter Beweis stellen konnte. Objektiv war es wohl bis dahin die Partie mit den meisten und gravierendsten Fehlern im Kampf.

Ja, die Opfer von Tal waren häufig inkorrekt, aber er ist trotz ihnen bzw. durch sie zum stärksten Spieler seiner Zeit ernannt worden. Die dahintersteckenden Ideen sind motivierend und lehrreich für andere Stellungen. Warum hat der Gegner die Widerlegungen nicht gesehen? Welche Motive kommen vor? Beispielsweise durch die Erstellung eines Spielerprofils von Tal kann ein anderer Blickwinkel aufgebaut werden. Der Schüler könnte sich in den chaotischen Stellungen fragen: „Was würde Tal jetzt tun?“ und kommt vielleicht zu genialen Ideen.

Sowieso sollten dem Schüler keine Varianten durch Trainer oder Engine vorgekaut werden, sondern er muss für einen Lernerfolg selbst überlegen. Studien (mit einigen Ausnahmen, siehe Kapitel 3.1.1) mit der Engine zu betrachten, gibt selbstredend wenig Sinn, aber auch bei anderen Stellungen ist dies der Fall.

Der Schüler sollte für sein Training ohne Trainer, in dem die Engine eine große Hilfe darstellen kann, angeleitet werden. Dafür kann der Trainer vorführen, wie er eine Partie mit Engine analysiert – wie er sie und sich kritisch hinterfragt.

Eine Engine ersetzt keine gute Vorbereitung. Wenn der Trainer seine Beispiele vorher selbst nicht verstanden oder schlecht ausgewählt hat, bringt in der Regel auch eine Analyse mit Engine im Hintergrund nichts.

Neben den „schach-internen“ Problemen kann die Engine dem Menschen natürlich auch weder bei den Bereichen der Psychologie, dem Zeitmanagement oder der Ästhetik helfen. Dafür wurde die Maschine selbstredend nicht geschrieben und es gilt diese wichtigen Bereiche gesondert zu betrachten.

Versuche, Erklärungen in menschengerechter Sprache durch die Schnellanalysefunktion von Lichess, DecodeChess o. Ä. zu liefern, bieten nette Ansätze, stecken aber meiner Empfindung nach noch in den Kinderschuhen bzw. sind noch sehr oberflächlich.

Komplett ohne Engine geht es heutzutage fast nicht mehr, weder in der Forschung, noch im Training. Warum nur fast? Weil es der junge indische GM D. Gukesh als Musterbeispiel geschafft hat, ein extrem starker Spieler zu werden, ohne großartig mit der Engine zu arbeiten.¹⁹

19 Vgl. www.chessbase.com/post/gukesh-wins-110th-hillerod-and-cannes-open-consecutively. Da er vermutlich andere Medien genutzt hat, die Engineanalysen voraussetzten, darf dieses „ohne Engine“ nicht zu wörtlich genommen werden. Dennoch wird ersichtlich, dass es auf die richtige Einstellung ankommt.

Zu guter Letzt möchte ich nun noch fünf Rollen vorschlagen und erklären, die meiner Meinung nach die Verwendung einer Engine im Training an entsprechenden Stellen begründen können. Die Engine kann verwendet werden als:

1. Problemlöser
2. Überprüfer
3. Inspirationsquelle
4. Negativbeispiel
5. Endgegner

1. Problemlöser: Die Lösung einer schwierigen Aufgabe sollte im Idealfall der Schüler ohne Hilfe erbringen. Da dies nicht immer funktioniert, gibt es Hilfestellungen, die der Trainer in Abstufungen geben kann, insofern er die Aufgabe verstanden hat. Manchmal gibt es jedoch praktische Probleme, die selbst starke Trainer nicht auf Anhieb lösen können, z.B. weil die Vorbereitung (ohne es zu wissen) unvollständig war.

Es kann hier mitunter fruchtbar sein, dass Schüler und Trainer zusammen rätseln. Der Trainer sollte in solchen Situationen denselben Ehrgeiz und dieselbe systematische Vorgehensweise an den Tag legen, die er auch von seinem Schüler erwartet. Der Schüler merkt dann, dass auch der Trainer manchmal Probleme hat und nicht unfehlbar ist. Das ist natürlich auf der einen Seite ein Eingeständnis und fordert vielleicht ein wenig Überwindung. Auf der anderen Seite macht es den Trainer aber auch nahbarer und zeigt die Komplexität von Schach.

Manchmal bringt aber auch das ambitionierteste Vorgehen des Trainers nichts, wenn er schlicht zu schwach bzw. die Aufgabe zu kompliziert ist. Man könnte das Problem als beiderseitige Hausaufgabe zum nächsten Trainingstermin aufschieben, jedoch ist dies unter praktischen Gesichtspunkten nicht immer möglich und verspricht nicht immer eine Lösung.

Wenn Schüler und Lehrer zu dem Schluss kommen: „Wir kommen hier nicht weiter, aber unter den Bedingungen einer Turnierpartie wäre xy unser Zug“, kann bei beiderseitigem Einverständnis die Engine befragt werden. Dies sollte nicht einfach aus Faulheit geschehen und der Lernfortschritt wird weniger stark vorangetrieben als bei einer aus eigener Kraft gefundenen Lösung. Jedoch hemmt eine ungelöste Aufgabe, in die man eine gewisse Zeit investiert hat, unter Umständen die Motivation, was sich negativ auf folgende Aufgaben auswirkt.

Entweder gibt die Engine bei ihrer Befragung eine klare Lösung an und diese muss nur nachvollzogen werden oder selbst mit Computerhilfe wird das Problem nicht klar lösbar. Z.B. anhand des Schemas von Arno Nickel (siehe Kap. 3.2.1) sollte der Trainer die Engine korrekt lesen und mit ihr gegen sie und sich selbst arbeiten.

2. Überprüfer: Diese Funktionsweise ist einfacher zu erklären und nicht immer notwendig. Wenn eine Problemlösung vorliegt, sollte der Trainer wissen, dass sie richtig ist. Z.B. bei nebenlösigen Aufgaben besteht aber unter Umständen eine Restunsicherheit, die durch die Engineprüfung im Nachhinein ausgeschlossen werden kann. Gerade in der Eröffnungsarbeit ist eine Überprüfung im Sinne des vorigen Kapitels häufig angebracht.

3. Inspirationsquelle:

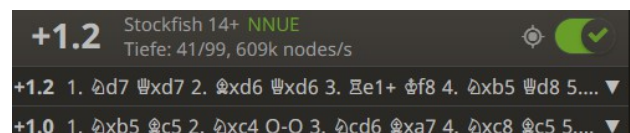
Auch wenn die Engine selbst keine Ästhetik kennt, kann sie doch Ansätze liefern, die für den Menschen kreativ wirken und den Horizont des eigenen Spiels erweitern können. Hier ein Beispiel aus meiner eigenen Eröffnungsarbeit im Katalanen (keine Aufgabe mit eindeutiger Lösung, es geht um die Suche nach Kandidatenzügen).

Diagramm 9



Lösung:

In dieser Stellung sind sehr viele Kandidatenzüge denkbar und nicht unbedingt schlecht. In all dem



Gewusel ist bisher jedoch noch keiner der von mir befragten fünf²⁰ Spieler (Spielstärke 2200–2500) auf den Computerzug gekommen. Erfreut waren dennoch alle nach der Auflösung und wir waren geneigt, die Stellung besser verstehen zu wollen. Engines können also durch ihre zauberhafte Art auch motivierend sein. **1.Sd7** funktioniert, da **1...Lxf4** (1...Dxd7 2.Lxd6 Dxd6 3.Te1+ Kf8 4.Dxb7±; 1...Kxd7?? 2.Lh3++-;

²⁰ Natürlich nicht repräsentativ.

1...Sf/bxd7?? 2.Lxd6+-) 2.Te1+ Se4 (2...Kd8 3.Sc5+-) 3.Sxe4 dxe4 4.Lxe4 Lxe4
 5.Txe4+ Kd8 6.Sxb8 Dxb8 7.Td4+ Ke8 8.Dd7+ Kf8 9.Txf4±

4. Negativbeispiel: Wie schon erwähnt, weiß die Engine nicht alles und kann auch nicht alles durch Variantenberechnung herleiten. Der Schüler weiß dies aber nicht immer und teilweise komplexe technische Erklärungen wie in Kapitel 2 könnten gerade jüngere Schüler überfordern.

Um den „Engine=Gott-Glauben“ zu widerlegen und Pragmatikern zu zeigen, dass Schach nicht nur aus reiner Variantenberechnung besteht, können Beispiele wie aus Kapitel 3.1.1 behandelt werden. Außerdem ist mit dieser Erkenntnis in der Regel ein kleiner Motivationsschub verbunden („Ich bin manchmal besser als die Engine!“).

5. Endgegner: „Meinst du, dass du das sogar gegen die Engine gewinnst/Remis hältst?“ ist eine Anfrage an das Selbstvertrauen des Schülers und in der Regel eine Steigerung von „Meinst du, dass du das sogar gegen mich gewinnst/Remis hältst?“. Gesundes Selbstvertrauen ist bekanntermaßen eine der wichtigsten Eigenschaften eines guten Schachspielers und kann hier gefördert werden.

Wenn der freundliche Komodowaran zum feindseligen Drachen und der Stockfisch zum Megalodon mutiert, zeigt sich häufig, ob eine Stellung wirklich verstanden wurde. Die Engine kann als Spielpartner verwendet werden, allerdings ist auch hier Vorsicht geboten. Stellungen, die technischen oder berechnenden Eigenschaften bedürfen, können gegen den Computer ausgespielt werden, wenn darauf geachtet wird, dass Engine- und Menschansatz in den Stellungen identisch ist.

Bei Standardendspielen wie dem Matt mit Springer und Läufer oder dem mit zwei Läufern kann die Engine eingesetzt werden, da sie, um das Matt möglichst lange herauszuzögern, auch menschlich gesehen zäh verteidigt. Beispielsweise bei Endspielen mit König und Bauer gegen König, in denen es darum geht, ein Schlüsselfeld zu erreichen, ist sie jedoch nicht immer geeignet. Die längste Variante verlangt dem Menschen nicht immer die am schwierigsten verständliche Technik ab und die Engine unterlässt auch gewisse Gewinnversuche, bei denen sie schon weiß, dass sie bei optimalem Spiel nichts bewirken.

Es gibt seit einiger Zeit Versuche, Engines „menschlicher“ (oder gar als „Bots“ im Stile bestimmter Personen) spielen zu lassen, allerdings ist der Programmieraufwand zu hoch, als dass dies für jedes einzelne Endspiel mit seinen jeweils eigenen Prinzipien

angewendet werden könnte. Ein menschlicher Gegner, der das Endspiel verstanden hat, ist also häufig der bessere Trainingspartner.

Natürlich ist bei allen Einsatzmöglichkeiten der Engine auf die äußeren Begebenheiten zu achten. Spielstärke und Alter der Schüler stellen wie immer entscheidende Faktoren bei der Themenwahl und Konzeption des Trainings dar. Wenn bspw. bei blutigen Anfängern die goldenen Eröffnungsregeln noch nicht verinnerlicht sind, bringt es nichts, mit der Engine geniale Feinheiten im 27. Zug der Bauernraubvariante im Najdorf zu entwickeln usw.

4. Fazit

Aus Sicht des Menschen ist erstaunlich, wie präzise ein Computerprogramm nur durch Variantenberechnung und/oder Statistik die Realität wahrnehmen kann. Aus Computersicht wäre wiederum erstaunlich, dass es scheinbar Wege gibt, durch weitere Methoden, die weder die Engine noch der Sterbliche selbst vollständig verstehen, effizienter die Wahrheit zu erkennen.²¹ Orientierungspunkt für den Computer sind die Schachregeln, der Mensch schafft es hingegen, in abgeschlossenen Systemen eigene Prinzipien mit Regelcharakter zu etablieren.

Wenn es um die reine Spielstärke geht, liegt die Maschine sicherlich uneinholbar vorne, sodass sie zum wesentlichen Bestandteil der Forschung an Schach, aber auch zum eigenen Training oder dem mit Schülern dazugehört. Wer versucht, das Spiel der Engine zu verstehen, kann häufig mehr Schlüsse für sein eigenes Spiel anstellen als über das der Engine.

Bei diesen Schlüssen können große Prinzipien verinnerlicht werden, es können noch nie dagewesene Ideen aufkommen, kurzum – es kann etwas gelernt werden. Vereinfachte Darstellungen oder die Verschließung gegenüber höheren Prinzipien „weil die Engine ja eh besser ist“, führen zu keinem Fortschritt. Es bedarf also der rechten Interpretation. Für das Training bieten die fünf erwähnten Rollen der Engine meiner Meinung nach eine Grundlage für eine gezielte Nutzung der Engine auch im Training. Wie von anderen Menschen verschieden Spielertyps, kann der Mensch viel von der Engine lernen, wenn er versucht, sie zu verstehen.

²¹ Als Theologe sei mir die Anmerkung verziehen, dass sich hier interessante Parallelen zu verschiedenen Weltanschauungsmodellen aufzwängen.

Quellenverzeichnis

Literatur:

ERTEL, WOLFGANG: Grundkurs Künstliche Intelligenz. Eine praxisorientierte Einführung, Wiesbaden 2016.

JUSSUPOV, ARTUR: Schachunterricht, 2006.

MÜLLER, KARSTEN/SCHAEFFER, JONATHAN: Man vs. Machine, 2018.

PROHASZKA, PETER zur Partie Petrosian-Hazai, Schilde 1970, in: Asik, Josip: American Chess Magazine. Issue No.21, 2021, S.67–73, zweiter Teil.

SADLER, MATTHEW/REGAN, NATASHA: Game Changer. AlphaZero's Groundbreaking Chess Strategies and the Promise of AI, Alkmaar 2019.

SULLIVAN, CHARLES zur Partie Spielmann-Rubinstein, St. Petersburg 1909, in: ChessBase Magazin 198, 2020.

DVDs:

MÜLLER, KARSTEN/ENGEL, LUIS: Spielertypen. Ihre Stärken und Schwächen, Hamburg 2020.

MÜLLER, KARSTEN: Deep Blue, in: Pelletier, Yannick et al.: Master Class Vol. 11. Vladimir Kramnik, 2019.

Online-Artikel:

Chessbase.com:

SHAH, SAGAR: Gukesh closes in on 2600, 03.06.2020, www.chessbase.com/post/gukesh-wins-110th-hillerod-and-cannes-open-consecutively.

SCHULZ, ANDRÉ: 25 Jahre Chessbase, 19.05.2011, <https://de.chessbase.com/post/25-jahre-chabase>.

Chess.com:

ERLY, ANDREW: Stockfish+NNUE, Strongest Chess Engine Ever, To Compete In CCCC, 20.07.2020, www.chess.com/news/view/stockfishnnue-strongest-chess-engine-ever-to-compete-in-cccc?ref_id=43524416.

KLEIN, MIKE: Google's AlphaZero besiegt Stockfish in einem 100 Partien Vergleich, 07.12.2017, www.chess.com/de/news/view/google-s-alphazero-besiegt-stockfish-in-einem-100-partien-vergleich-3971.

SERPER, GREGORY: Computers in chess... Good or Evil? Part Two, 17.02.2013, www.chess.com/article/view/computers-in-chess-good-or-evil-part-two.

Sonstige:

www.greelane.com/wissenschaft-technologie-mathematik/wissenschaft/how-many-atoms-are-in-human-body-603872/, 05.04.2019.

Längstes Tablebase Endspiel: <https://tb7.chessok.com/probe/3/61>.

Stockfish evaluation guide: <https://hxim.github.io/Stockfish-Evaluation-Guide/>.

EIGEMANN, WALTER: Interview mit Fernschachgroßmeister Arno Nickel, 2010, www.glarean-magazin.ch/2010/12/14/interview-mit-fernschach-grossmeister-arno-nickel/.

YouTube-Videos:

SOULEIDIS, GEORGIOS: Analysen der Matches DeepBlue gegen Kasparow, YouTube kanal „The Big Greek“ 2020, www.youtube.com/watch?v=E1qatUrxpHs&list=PLXoFpqP9YfB1ICChsu6EwMi3cfmrkviPo.

BUSSE, MICHAEL: Interview mit Matthias Wüllenweber, www.youtube.com/watch?v=8Kjkwj9DTk8, YouTube kanal „Schachgeflüster“ 04.02.2022.

BUSSE, MICHAEL: Interview mit Stefan Meyer-Kahlen, www.youtube.com/watch?v=FDrfWTTyQZ8, YouTubekanal „Schachgeflüster“ 27.11.2021.

BUSSE, MICHAEL: Interview mit Peter Heine Nielsen, www.youtube.com/watch?v=MRpZFo2kZAs, YouTubekanal „Schachgeflüster“ 11.06.2021.

BUSSE, MICHAEL: Interview mit Arno Nickel, www.youtube.com/watch?v=mFlAnL_70U4, YouTubekanal „Schachgeflüster“ 14.03.2020.

RISKO, NICK: Chess Puzzles that Computers Can't Solve!, www.youtube.com/watch?v=15nuJdAUW0s, YouTube-Kanal „Saint Louis Chess Club“, 25.01.2021.

Mamedyarovs Statement: „Magnus has a chance against engines“, www.youtube.com/watch?v=S9_IqQpc-FY, YouTubekanal „Daily Dose of Chess Clips“ 30.01.2022.

NAKAMURA, HIKARU: Chess Positions That Are Too Confusing For Computers, www.youtube.com/watch?v=-UqCAIQEw2E, YouTubekanal „GMHikaru“ 21.05.2021.

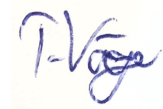
SADLER, MATTHEW: "Exactly How to Attack" | DeepMind's AlphaZero vs. Stockfish, YouTubekanal „chess24“, 06.12.2018, www.youtube.com/watch?v=bo5plUo86BU.

Eigenständigkeitserklärung

Hiermit bestätige ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken (dazu zählen auch Internetquellen) entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht.

Osnabrück, den 13.04.2022

Ort, Datum



Tobias Vöge