



Wer einer typischen Computer-Anzeige wie „plus 0,8 Bauern“ blind vertraut, kennt die Rechner nicht. Natürlich muss man in die Varianten hineingehen ... aber wann ist es genug? Wenn Sie selbst die Stellung klar bewerten können, wenn der Computer plus zwei anzeigt, oder plus drei? Es hängt von der Position ab: Computer sind tolle Analysehilfen, aber es gibt noch etliche Stellungstypen, in denen sie ziemlich neben der Spur laufen und die fachliche Unterstützung des Menschen brauchen.

London, Dezember 2000: Wladimir Kramnik zieht Kasparows Eröffnungsvorbereitung mit der Berliner Verteidigung die Zähne. Nach seinem Matchsieg sagt er, er habe absichtlich eine Variante gespielt, die Computer nicht gut verstehen, um seinem zu computergläubigen Gegner maximale Probleme zu bereiten.

Brissago, Oktober 2004: Derselbe Wladimir Kramnik opfert gegen Peter Leko die Dame, um einen unaufhaltbaren Freibauern zu bilden:

Kramnik - Leko, Brissago 2004



6k1/5p1p/p1pb1nq1/1P4p1/3P4/1BPb1PP1/1P1NrQ1P/R1B3K1 w - - 0 24

Schachprogramme finden eine gewisse Zeit Dxe2 ungeheuer attraktiv, weil sie den a-Bauern durchlaufen sehen. Weiß kann tatsächlich umwandeln und zieht sogar mit Schach ein, wird aber dennoch mattgesetzt, bevor er von seinem Materialvorteil profitieren kann: **24.Dxe2 Lxe2 25.bxa6 Dd3 26.a7** (Kramnik roch hier den Braten und zog **26.Kf2**, verlor aber trotzdem) **26...De3+ 27.Kg2 Lxf3+ 28.Sxf3 De2+ 29.Kg1 Sg4+** Die Vermutung, er habe in der Vorbereitung einfach nur den Computer nicht lange genug laufen lassen oder zu blind dessen Einschätzung vertraut, wies Kramnik übrigens zurück.

## Metzel-Schwäche

Ein Einzelfall? Keineswegs! Dynamische Stellungen, in denen beide Seiten starke Drohungen haben, verstehen Schachprogramme öfter nicht, oder erst nach längerer Rechenzeit.

Anand – Morosewitsch, San Luis 2005



1rb2rk1/6p1/p1q1p3/1pbpPp1Q/2nB1P2/P1N4R/1PP3PP/2K2R2 w - - 0 23

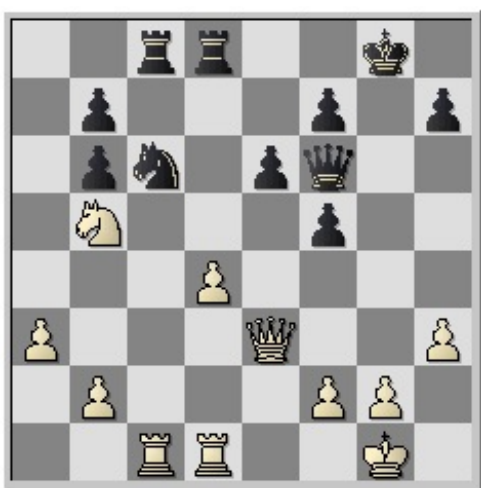
Anand hat den Läufer auf h7 geopfert und muss schnell über den gegnerischen König herfallen. Schwarz steht bereit, Weiß am Damenflügel zu vernichten. Anand spielte das natürliche Tf3 und wickelte in ein Dauerschach ab. Programme eiern zwischen den verschiedensten Zügen herum, die sie alle mit einer Bewertung im Remis-Bereich versehen. Analysiert man ein bisschen herum, um herauszufinden, dass es keinen Gewinn für Weiß gibt, fällt sehr schnell auf, dass die Schachprogramme große Schwierigkeiten mit dem schwarzen Gegenspiel haben, weil sie die weißen Angriffschancen zu gut bewerten. Auch wenn Weiß mal ein bisschen opfert, zeigen die Programme noch Bewertungen nahe dem Remis an.

Und da liegt das zweite Problem von Schachprogrammen in dynamischen Stellungen: Remis-Bewertung kann heißen, es sind nur noch die Könige auf dem Brett, es gibt ein technisches Remis, das Schachprogramm hält eine Zugwiederholung für die beste Möglichkeit, ob zu Recht oder zu Unrecht – oder Remis-Bewertung bedeutet „ich habe keine Ahnung, was hier los ist, aber innerhalb meiner Suchtiefe sehe ich kein Unheil“. Wobei die angezeigte Suchtiefe keineswegs bedeutet, dass alle Varianten so tief untersucht worden wären; im Gegenteil beschneiden alle modernen Programme drastisch.

Die Schlussfolgerung für Eröffnungs-Analytiker muss daher lauten: die Bewertung von Schachprogrammen in Stellungen aus dem Königsgambit, der Traxler-Variante oder ähnlich komplexen Systemen mit beiderseitigen taktischen Chancen ist mit sehr sehr großer Vorsicht zu genießen. Natürlich wird niemand solche Eröffnungen ohne Programm untersuchen wollen, aber wenn eine Engine mal eine Minute lang plus zwei anzeigt, dann muss das noch gar nichts heißen.

In ruhigen Stellungen finden sich die Programme dagegen ziemlich gut zurecht, wie auch das Match zwischen Kramnik und Fritz zeigte. Vor allem die Standardpläne bereiten keine Probleme, jedes moderne Programm weiß den Minoritätsangriff zu spielen, soviel Wissen haben sie von ihren Schöpfern implantiert bekommen. Die Hauptstärke liegt aber nach wie vor in der tiefen Suche, aus der viele Lösungen von Stellungsproblemen herausfallen, die Menschen als „positionell“ bezeichnen würden.

DeepBlue – Kasparow, Philadelphia 1996

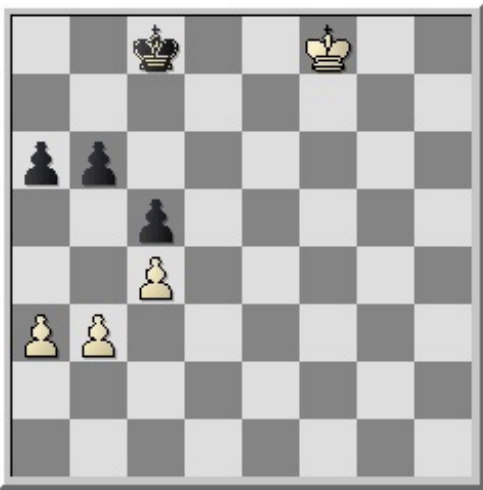


2rr2k1/1p3p1p/1pn1pq2/1N3p2/3P4/P3Q2P/1P3PP1/2RR2K1 w - - 0 23

DeepBlue, nicht gerade als Musterbeispiel ausgefeilten Schachwissens bekannt, spielte hier **23.d5**. Die Kommentatoren waren aus dem Häuschen: der Computer opfert einen Bauern, um seine Bauernschwäche gegen aktives Spiel einzutauschen. Der große Garri wunderte sich, denn damals machten Computer so etwas nicht. DeepBlue war aber ein dummes Programm und hatte keine Ahnung, was es machen darf und was nicht; die Maschine hatte vor d5 bereits erkannt, dass sie den Bauern zwangsläufig zurückgewinnt. Über die Schwäche von isolierten Bauern hatten die Programmierer das blaue Wunder offenbar aufgeklärt, und so ergriff es freudig die günstige Gelegenheit, sich des Schwächlings zu entledigen – reine Taktik.

## Zieh, Cowboy!

Das Vorurteil über taktisch unfehlbare, aber positionell anfällige Schachprogramme trifft also nicht mehr zu – es handelt sich um ein Relikt aus der Steinzeit. Ein Fossil, das überlebt hat, ist dagegen die Endspielschwäche. Es hat sich zwar einiges getan, dennoch bleibt die Bewertung von Endspiel-Stellungen höchst zweifelhaft. Der Hauptgrund dafür liegt im Zugzwang-Problem. Jedes moderne Schachprogramm nutzt die Nullmove-Technik. Zugrunde liegt die Idee, dass es meist eine schlechte Idee ist, gar nichts zu tun und dem Gegner einen Zug zu schenken. In taktischen Stellungen hilft der Nullmove, sehr schnell minderwertige Varianten zu erkennen, ein Nullmove-Programm sucht wesentlich tiefer und in wichtigeren Varianten. Die Annahme, es sei nachteilig, einmal auszusetzen, trifft aber im Endspiel nicht immer zu – jedes simple Dreiecksmanöver, jedes Gegenfeld-Manöver führt die Idee ad absurdum. Schachprogramme schalten daher im Endspiel gelegentlich den Nullmove aus, die Kriterien, nach denen sie vorgehen, sind aber ziemlich primitiv und überaus lückenhaft.



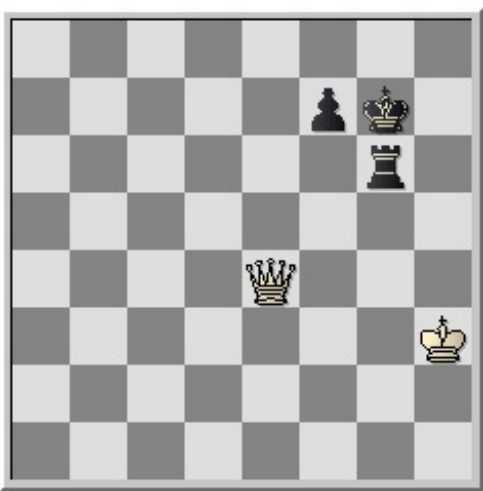
2k2K2/8/pp6/2p5/2P5/PP6/8/8 w - - 0 1

Diese hübsche, aber für Menschen nicht zu schwierige Stellung verwirrt die meisten Programme nachhaltig.

**1.Ke8?** halten sie fast alle für interessant, aber nach **1...b5 2.a4 Kc7 3.Ke7 Kb6 4.Kd6 b4** ist die Stellung remis, weil Weiß wegen Patt nicht den c5 schlagen darf, wenn Schwarz Ka5 zieht. Das ist nicht sehr tief, und die Programme entdecken es schnell, weshalb sie 1.Ke8? auch mit Remis-Bewertung spielen.

Weiß gewinnt mit **1.a4!**, z.B. **1...b5 2.Ke7 Kc7 3.a5**; die Schwierigkeit für die Programme besteht in der Variante **2....Kd7**, wonach Weiß mit **2.a5!** gewinnt. Weil sie diese Variante für uninteressant halten, entscheiden sich die meisten Programme in der Diagrammstellung dafür, den König näher ans Getümmel zu bringen – und verschenken so den Sieg.

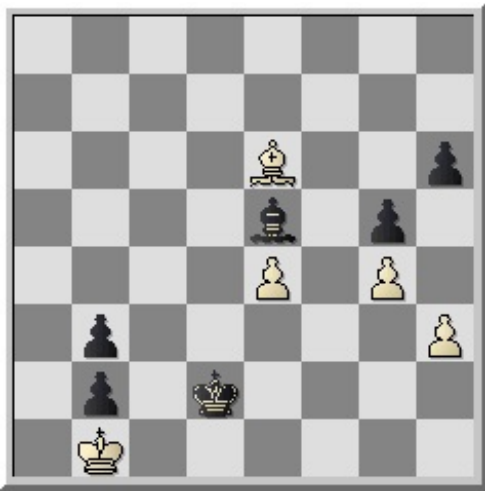
Ein grundsätzliches Problem haben Schachprogramme mit Festungen. Programme haben keine Ahnung von Tempi, sie verfolgen keine Pläne, sondern rechnen bloß, und sind daher auch nicht in der Lage zu erkennen, ob sie in einer Stellung Fortschritte machen oder nicht.



8/5pk1/6r1/8/4Q3/7K/8/8 w - -

Jeder Spieler mit Schachverstand für fünf Pfennig erkennt, dass Weiß die gegnerische Blockade nie durchbrechen kann. Programme haben aber nur Schachverstand für ein oder zwei Pfennig, und hier hilft tiefe Suche gar nichts – man kann 50 Züge lang den Materialvorteil behaupten, aber das war schon alles.

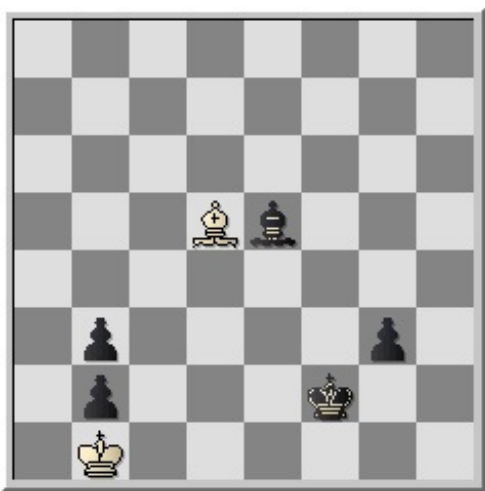
Der dunkelste Punkt am Himmel der computerisierten Schach-Herrlichkeit ist das leidige Endspiel mit ungleichen Läufern. Ein Freibauer stellt Schachprogramme ja ohnehin schon vor grundsätzliche Probleme, weil sein Materialwert undefiniert ist – kann er geschlagen werden, war er nix wert, sonst ist er erstmal ein Bauer, aber wenn er umwandelt, steht da plötzlich eine Dame. So früh wie möglich zu wissen, zu welcher Art Heldentum ein Freibauer neigt, ob zu Selbstaufopferung oder zum erfolgreichen Karriereleiter-Klettern, stellt eine große Herausforderung für Computerprogramme dar. Und dann kommen die Leute mit den ungleichen Läufern, und alles stimmt nicht mehr, alle Regeln für Freibauern funktionieren auf einmal ganz anders! Vor allem kann Gier ebenso schädlich sein wie in jeder anderen Stellung. Beispiel?



8/8/4B2p/4b1p1/4P1P1/1p5P/1p1k4/1K6

Dem Menschen scheint die Stellung ziemlich klar. Ein logischer Plan für Schwarz besteht darin, sich mit dem König über g2 an den h-Bauern heranzuschleichen und ihn zu verhaften. Dann kann er h5 ziehen und so auf der g-Linie einen weiteren Freibauern bilden, für den Weiß den Läufer geben muss. Die weißen Freibauern werden vom zentral postierten schwarzen Läufer mühelos gebändigt. Die Programme sehen das bestimmt, wenn sie hier einen schwarzen Vorteil von etwa zwei Bauern anzeigen, oder? Mit Weiß wollen sie jedenfalls alle auf b3 den Bauern schlagen, vermutlich aus Verzweiflung.

Die ist aber gar nicht angebracht, zumindest dann nicht, wenn man den Bauern b3 stehen lässt, denn es gibt noch ein kleines, aber wesentliches Detail in dieser Stellung: der weiße König steht auf patt. Das ist sehr unangenehm für Schwarz, weil es den skizzierten Gewinnplan vereitelt, denn Weiß könnte einfach seine Bauern opfern:



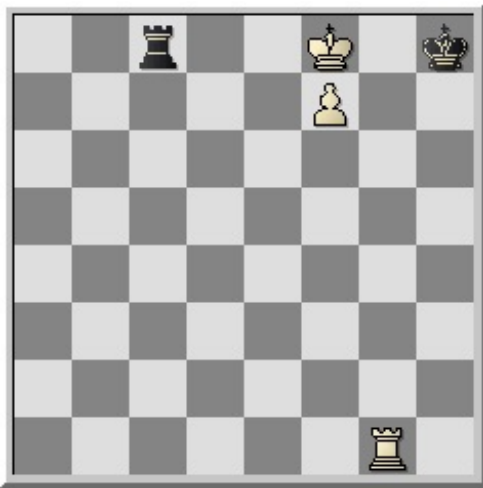
8/8/8/3Bb3/8/1p4p1/1p3k2/1K6

Diese Stellung dürfte wirklich für jeden Menschen oberhalb 800 Elo als offensichtlich unentschieden erkennbar sein. Schwarz kann nie den Bauern vorrücken, weil Weiß schlägt und pattgesetzt würde, falls Schwarz den Läufer nähme, beide Seiten können also nur noch hin- und herziehen. Die Programme werfen hier aber bis zu sechs Bauern Vorteil für Schwarz aus, und die Sechsstener-Tablebases helfen kein bisschen, weil ja nichts geschlagen wird. Auch hier wird erst die 50-Züge-Regel für ein Einsehen bei denkünstlichen "Intelligenzen" sorgen.



### Mehr Licht!

Schachprogramme sind für Analytiker praktische Werkzeuge, wenn sie um die Schwächen wissen und „drumherum analysieren“ können. Manche Programmierer haben ihren Geisteskindern jedoch spezielle Funktionen implantiert, um Analytikern unter die Arme zu greifen. Für Endspiel-Analysen unverzichtbar sind die Tablebases, die alle Stellungen mit bis zu sechs Steinen nebst perfekter Bewertung enthält. Ein kompliziertes Endspiel mit Dame gegen Dame und Bauer mutiert dadurch zur in Millisekunden erledigten Fingerübung, und für eine Stellung, die den Autor mehrere Nächte nicht schlafen ließ und über die in der Zeitschrift Kaissiber stand: „Schirow versuchte, blind zu lösen, scheiterte jedoch, und Swjaginsew fand es zuerst nicht, aber über Nacht hatte er Erfolg. Dizdar brauchte 20 Minuten, Vescovi 40.“, werfen die Programme ohne zu rechnen „Gewinn in 23“ aus:



2r2K1k/5P2/8/8/8/8/6R1 w - -

Der erste Zug ist Ke7 ...

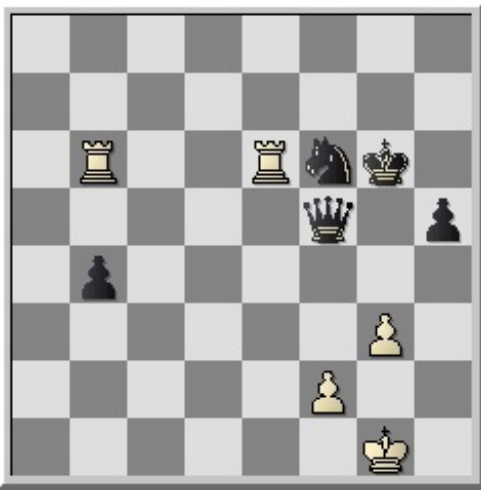
Die Endspieldatenbanken nutzen auch, wenn noch mehr als sechst Steine auf dem Brett sind, weil die Programme auch im Suchbaum auf sie zugreifen und während der Überprüfung von Abtausch-Aktionen für die resultierenden Stellungen perfekte Bewertungen bekommen. Fast alle Programme können mit den Endspieldatenbanken umgehen; wer sie nicht auf seinem Rechner installiert hat, kann sie auf der Seite [www.shredderchess.de](http://www.shredderchess.de) per Webbrowser abfragen.

Nicht nur für Endspiele wichtig ist die Fähigkeit eines Schachprogramms dazuzulernen. Wenn eine Stellung falsch bewertet wird, spielt man die korrekte Variante vor, bis der Blechtrottler ein Einsehen hat und korrekt bewertet. Geht man dann zur Ausgangsstellung zurück, sollte auch diese korrekt bewertet werden. Mit der „komm-ich-zeigs-dir“-Methode kann man sehr komplizierte Stellungen bis in die kleinsten Verästelungen analysieren – sofern das Schachprogramm diese Funktion unterstützt. Ein bisschen können es die meisten, wirklich gut sind aber nur Shredder und Rybka in dieser Disziplin, während beispielsweise Fritz recht schnell unter Amnesie zu leiden beginnt, wenn man viele Verästelungen untersucht.

Den Programmen Varianten erklären zu können ist zweifellos ein Schritt in die richtige Richtung, nämlich ein möglichst gut funktionierendes Analyse-Team aus Mensch und Computer zu bilden, aber es wird schnell anstrengend, beispielsweise in einem schwierigen Endspiel, welches das Programm der Wahl nicht durchschaut, immer wieder Varianten bis zur maschinenverständlichen Entscheidung vorzuspielen. Bei Festungen versagt die Methode völlig, weil erst die 50-Züge-Regel die Gewinnversuche der Programme bremsen kann. Die Lösung des Problems stammt von Großmeister Karsten Müller, der während der Arbeit an seinem Endspielbuch vor denselben Schwierigkeiten nicht kapitulierte, sondern dem Autor des Schachprogramms Shredder den Vorschlag machte, dem Programm eine Stellungsbewertung vorgeben zu können.

## Ins Hirn gegriffen

Seit Shredder 10 (aber nicht in der von Chessbase vertriebenen Version!) kann man per Rechtsklick auf das Engine-Fenster die Gott-Funktion „bestimme Stellungswert“ aufrufen und eine Bewertung vorgeben: Gewinn, Verlust, plus oder minus drei, ein Bauer Vorteil, Remis ... Die Festung im Beispiel oben würde, weil nur fünf Steine auf dem Brett stehen, schon von den Endspieldatenbanken erschlagen, aber man kann die Sache ja etwas verkomplizieren:



8/8/1R2Rnk1/5q1p/1p6/6P1/5P2/6K1 b - -

Diese Stellung ist remis. Weiß spielt **1.Txf6 Dxf6 2.Txb4** und hat eine Festung, weil er nur noch mit dem Turm zwischen f4 und h4 pendeln muss; Schwarz kann keine Fortschritte machen. Shredder, wie jedes andere Programm, hat davon natürlich keinen Schimmer. Per Gott-Funktion erklärt man nun die Endstellung der Variante (nach Txb4) für unentschieden, und schon zaubert das derart gedopte Programm den Zug 1.Txf6 mit Remis-Bewertung aus dem Hut.

Es ist interessant, dass diese so einfache und nützliche Funktion erst jetzt in einem Schachprogramm auftaucht, denn technisch wäre das vor 20 Jahren schon kein Problem gewesen.



### Fazit

Natürlich sind Schachprogramme Taktik-Monster, keine Frage. Aber sie behandeln eine Stellung nicht automatisch richtig, nur weil sie taktisch ist – in dynamischen Stellungen mit gegenseitigen Chancen hauen sie gern mal daneben, und im Endspiel gurken sie oft noch immer kräftig herum. Es eignen sich auch nicht alle Programme gleich gut zur Analyse – wie gut, hängt gar nicht von der reinen Spielstärke ab, sondern davon, wie gut Programm und Bediener zusammenarbeiten. Manche Programme unterstützen das gar nicht, manche wenigstens ein bisschen. Karsten Müllers Gott-Funktion dürfte der bisher größte Schritt in Richtung Analyse-Cyborg sein. Die Entwickler der anderen Programme wären gut beraten, diese Funktion ebenfalls zu implementieren und auf diesem Weg noch ein Stück weiterzugehen. Denn viel wichtiger als ein Sieg gegen einen Weltmeister ist es, welchen Nutzen Otto-Normalanwender aus seinem Programm ziehen kann.

*(Lars Bremer)*

---