Swisstest: ...die schlechten ins Kröpfchen!

Großmeister können es, Weltmeister tun es, Amateure möchten es, und Patzer sollten es – nämlich das, was (virtuos angewendet) im Schach die große Klasse von der großen Masse unterscheidet: das Vermeiden schlechter Züge. Genau dieses für jede Schach-Performance Fundamentale liegt auch einem neuen, 64-teiligen Computerschach-Test zugrunde, welcher ungewohnte Wege geht, indem seine Stellungen mal nicht nach dem einzig richtigen Zug fragen, sondern mit einem verführerisch falschen locken. Der sog. *SwissTest* wurde vom Autor unlängst der breiten Computerschach-Community vorgestellt.

Ein praktizierender Schachspieler, gleich welcher Stärke, erfährt bei jedem Turnier (oft schmerzlich) von neuem, dass der Erfolg keineswegs von seinen gefundenen *guten*, sondern vielmehr von den vermiedenen *schlechten* Zügen bestimmt wird. Kaum je die *Tops*, sondern fast immer die *Flops* entscheiden über den Ausgang von Turnier-Partien – von der Welt des Blitz- und Rapid-Schachs noch gar nicht geredet.

Selbstverständlich, es gibt sie durchaus, jene Genie-Blitze der Aljechins und Kasparows, die aus heiterem Himmel mit einem Donnerblitz die feindliche Stellung in Schutt und Asche legen. Aber man schaue sich in der Schachliteratur der großen Kommentatoren von Keres bis Karpow mal genau an, was diesen ach so fetten Doppel-Ausrufezeichen in vielen Fällen vorausgeht! Man ahnt es: ein dickes Fragezeichen für den Verlierer. (Apropos und allerdings: Man lasse mal zum Spaß die modernen Spitzen-Schachprogramme die Kommentatoren kommentieren ... Schlicht unglaublich, welch haarsträubender Varianten-Unsinn einem jahrzehntelang, großmeisterlich und unwidersprochen zwischen auch *renommiertesten* Buchdeckeln aufgetischt wurde. Es ist sehr zu hoffen, dass Computerschach sich dereinst bis zu den Schach-Verlegern rumspreche ... Oder auch nicht. Denn das Wehklagen über so viel zu zer*shreddernde* Makulatur wäre riesig – wenngleich ja das beeindruckende schachkultur-historische Erbe der sog. "Theorie" davon unberührt bliebe.)

Swisstest: ...die schlechten ins Kröpfchen!

64 Partiestellungen aus 100 Jahren Schachgeschichte

Kurzum, vom Remis-Tod des Schachs dürfen allenfalls Roboter, aber keinesfalls Menschen reden! Womit wir beim Gegenstand dieses Reports wären, bei der Computerschach-Aufgabensammlung *SwissTest2*, welche der Schreibende Ende Dezember 2004 (als grundlegend verbesserte und erweiterte Neuauflage einer vor zweieinhalb Jahren entstandenen Fassung) präsentierte. (Der Name dieser mit 64 Mittel- und Endspielstellungen aus den letzten 100 Jahren Schachgeschichte bestückten Test-Suite hat rein private Gründe und leitet sich übrigens aus der chronischen Verärgerung des Autors über manche Spitzenpolitiker seines Schweizer Heimatlandes ab, welche in ihrer Angst vor Fehlentscheiden lieber gleich gar nichts mehr machen ...)

Ganz bewusst enthält der *SwissTest* nicht nur menschliche Fehler von Tschigorin und Tarrasch über Fischer und Spasski bis hin zu Schirow und Kasparow, sondern zahlreiche Positionen auch aus der allerjüngsten internationalen Computerschach-Szene, wo die "Silikanten" entscheidend danebenhauten. (Inwiefern sich übrigens Mensch und Maschine hinsichtlich der Qualität ihrer Fehler unterscheiden, wäre ein speziell interessantes Untersuchungsfeld!). Allen "Lösungszügen" ist jedenfalls eines gemein: Sie sollen *nicht* gefunden werden. Denn diese zu vermeidenden Züge (im CS-Jargon: *avoid moves* bzw. *am*) führen entweder forciert zum Verlust oder sind zumindest die nachweislich schlechtesten unter mindestens drei, oft auch vier oder mehr *vernünftig spielbaren* Varianten. Gleichwohl ist es die eigentliche Pointe dieses Schach-Testes, dass die falschen Züge äußerst verführerisch sind: Unter allen 64 Aufgaben ist wohl keine, die nicht von mindestens einem Dutzend gegenwärtig benutzter Engines trotz schneller Hardware falsch gelöst würde...

Swisstest: ...die schlechten ins Kröpfchen!

Bad moves für Anwender und Programmierer

Von Anfang an schwebten mir zwei Zielsetzungen des *SwissTest* vor:
1. Dem Anwender ein schnelles Instrument an die Hand zu geben, mit welchem die ungefähre Leistungsgegend eines neuen Programms in dem mittlerweile über 300 Engines umfassenden (bzw. fast wöchentlich wachsenden) Feld eruiert werden könne;
2. Dem Programmierer behilflich zu sein beim Verbessern seiner Engine, indem der Test relativ verlässlich den schachlichen Fortschritt indiziere.

Nun kann bekanntlich ein Stellungstest, zumal mit "nur" 64 Aufgaben, grundsätzlich keine Engine-Rangliste erzeugen, welche zu 100 Prozent korrelierte mit einer seriösen, von der internationalen Anwenderschaft in abertausenden von Computer-Matches generierten Turnier-Rangliste. Eröffnungsbücher, Zeiteinteilung, Lern- oder Pondering-Techniken, Single- oder Dual-Verwendung etc. – das alles entscheidet bekanntlich über die Turnier-Performance eines Programms ebenso wie seine reine Spielstärke. (Meine ganz persönliche Referenz-Rangliste findet der Leser übrigens in dieser Ausgabe der CSS Online, wobei der Schreibende ziemlich zufällig der Urheber der betreffenden Datenbank ist: Die wichtigste Arbeit wurde von zahlreichen fleißigen Engine-Testern aus der ganzen Welt geleistet).

Was jedoch eine nicht zu kleine, mit *korrekten* und *eindeutigen*, dabei schachlich *ausgewogenen* und ein *breites Spektrum* aufweisenden Stellungen bestückte Test-Suite durchaus zu leisten vermag, ist eine *grobe relative* Rangierung. Dabei sind Ausreißer praktisch unvermeidlich; mir ist jedenfalls kein Stellungstest (egal welcher Ziel- und Zusammensetzung) bekannt, der nicht (teils sogar krasse!) Fehl-Platzierungen produzierte. (Ich habe verschiedentlich Berechnungs-Software angeregt, welche diese immanente Krankheit aller Test-Suiten eventuell eindämmen könnte, und vereinzelte Programmierer haben versprochen, sich dieser Sache mal anzunehmen; auf die Entwicklung darf man gespannt sein!)

Swisstest: ...die schlechten ins Kröpfchen!

Zwei mögliche Verfahren im *SwissTest*

Der *SwissTest*-Download enthält gegenwärtig drei Dateien:

- Eine mit der weltweit sehr verbreiteten ChessBase-GUI erstellte CBH-Datei, welche neben dem Avoid-Move auch alle vernünftigerweise spielbaren Alternativen enthält. Wird also Fritz & Co. fürs *automatisierte Abarbeiten* verwendet, haben die Engines neben dem Avoid-Move auch *zwingend die besseren Züge* zu berücksichtigen;

- Eine PGN-Datei, welche von jeder modernen Schach-Oberfläche verarbeitet werden kann, ebenfalls die Varianten & Analysen enthält und v.a. fürs *manuelle Testen* verwendbar ist;

- Eine EPD-Datei, die *nur die Avoid-Moves* (alle Züge mit Fragezeichen) auflistet und die ebenfalls in jede gängige GUI importiert werden kann.

Der Test-Autor selbst bevorzugt zwar die CBH-Datei, er konnte sich aber nicht endgültig festlegen, welches der beiden grundsätzlichen Test-Verfahren sinnvollerweise angewendet werden soll ... Lässt man nämlich die Suite unter Fritz & Co. verarbeiten, hat man zwar die Gewissheit, dass der schlechte Zug nicht etwa gar "zugunsten" eines noch schlechteren vermieden wird. Andererseits wird de facto aus dem angestrebten am- ein "ganz gewöhnlicher" Test, welcher nach Best-Moves fahndet. Ein entscheidender struktureller Unterschied bleibt natürlich trotzdem: Es sind mehrere Lösungen möglich, während traditionelle Stellungstests weitaus überwiegend je *die eine richtige* abfragen.

Die zweite Test-Option ist der (ggf. automatisierte) Einsatz der EPD-Datei, wobei eine Aufgabe bereits als gelöst gilt, wenn der Avoid-Move vermieden wird – aus welchen Gründen auch immer ... In dieser Variante wird die grundlegende Konzeption dieses *SwissTest2* natürlich am reinsten realisiert – auch wenn hier die Zuverlässigkeit von Rankings beeinträchtigt werden könnte. Auf alle Fälle wäre ein Resultat-Vergleich beider Verfahren nicht uninteressant.

Swisstest: ...die schlechten ins Kröpfchen!

60 Sekunden lang Leichtes bis Mittelschweres

Der *SwissTest2* will als pures *Pragmatikum* daherkommen: Er sollte weder zu leicht noch zu schwer sein, er will "jedem Programm etwas bieten", und er kann in rund einer Stunde/Engine absolviert werden. Auf schneller aktueller Hardware (ab ca. Pentium 4) empfehle ich also eine Bedenkzeit von 60 Sekunden/Stellung, was heutzutage +/– auch einer weitverbreiteten Bedenkzeit-Praxis in der Tester-Community (z.B. 40/40-Turniere u.Ä.) entspricht. Auf älteren PCs mögen 90 oder gar 120 Sekunden angemessen sein. Die Bedenkzeit ist dabei so lange zweitrangig, als für alle Engines *identische Bedingungen* gewährleistet sind, denn dieser Test funktioniert nicht nach einer bestimmten "Formel".

Hingegen lege ich sehr die Verwendung des Ranking-Tools "EloStatTS" des Statistikers Frank Schubert nahe, zumal beim Testen unter der Fritz-GUI. Schuberts Utility verwendet ein mathematisches Verfahren, das sehr differenzierte Ranglisten generiert. Für ein zuverlässiges Resultat ist allerdings eine möglichst hohe Anzahl beteiligter Engines Voraussetzung.

Swisstest: ...die schlechten ins Kröpfchen!

64 Programme kontra 64 Verführungen

Das nachstehende SwissTest2-Ranking wurde mit insgesamt 64 Engines bei einer BZ von 60 Sekunden/Position auf einem Pentium 4-3 GHz/128 MB Hash/4-Steiner-TBs/Fritz8-GUI erspielt, ausgewertet wurde mit EloStatTS (Startwert: 2600). Die Auswahl der beteiligten Engines war dabei nicht zufällig, weil gleichzeitig noch etwas "vergleichende Verhaltensforschung" betrieben werden sollte: Ich zog 32 Programme in je zwei Versionen (nämlich der jeweils aktuellen sowie einer älteren) hinzu, und zwar so, dass das neuere Release auch das jeweils erwiesenermaßen stärkere ist. (Denn es ergibt keinen Sinn, Versionen mit minimalem Stärke-Unterschied – berühmt-berüchtigstes Beispiel: ChessTiger 15 & ChessTiger 2004 – zu vergleichen. Solche Differenzierung kann ein Test mit "nur" 64 Stellungen vorläufig noch nicht leisten. Als Turnier-Anhaltspunkt konsultierte ich die bereits erwähnte Comp2004 sowie allerneuste private Match-Ergebnisse verschiedener Einzeltester).

	Program	Elo	+/-	Matches	Score
1	Shredder8-128MB/P4-3Ghz	2693	7	3252	63.6%
2	Fritz8-128MB/P4-3Ghz	2672	8	2961	60.6%
3	Shredder6.02-128MB/P4-3Ghz	2670	7	3039	60.4%
4	Junior8-128MB/P4-3Ghz	2665	8	2857	59.6%
5	Junior9-128MB/P4-3Ghz	2655	8	2836	58.1%
6	Gandalf6.0-128MB/P4-3Ghz	2650	7	3014	57.6%
7	Hiarcs9-128MB/P4-3Ghz	2643	8	2801	56.3%
8	Ruffian2.0.2-128MB/P4-3Ghz	2637	7	2971	55.6%
9	Aristarch4.50-128MB/P4-3Ghz	2637	7	2863	55.5%
10	Fritz6-128MB/P4-3Ghz	2635	7	2879	55.2%
11	SOS4forArena-128MB/P4-3Ghz	2634	7	2902	55.2%
12	Gandalf4.32h-128MB/P4-3Ghz	2633	8	2777	55.0%
13	Ktulu4.2-128MB/P4-3Ghz	2633	8	2602	54.9%
14	List512-128MB/P4-3Ghz	2633	8	2828	54.8%
15	Aristarch4.40-128MB/P4-3Ghz	2629	8	2862	54.4%
16	TheBaron1.5.0-100MB/P4-3Ghz	2629	8	2775	54.3%
17	Fruit2.0-128MB/P4-3Ghz	2625	8	2776	53.8%
18	Gothmog1.0-128MB/P4-3Ghz	2622	8	2617	53.1%
19	Tao5.6-128MB/P4-3Ghz	2619	8	2667	52.9%
20	Crafty19.15-108MB/P4-3Ghz	2617	8	2613	52.5%
21	Pharaon3.1-128MB/P4-3Ghz	2613	8	2628	51.9%
22	Ruffian1.0.1-128MB/P4-3Ghz	2613	8	2649	52.0%
23	SmarThink0.17a-128MB/P4-3Ghz	2612	8	2866	51.8%
24	Delfi4.5-128MB/P4-3Ghz	2611	8	2630	51.7%
25	Hiarcs8-128MB/P4-3Ghz	2610	8	2759	51.5%
26	Chessmaster10000-128MB/P4-3Ghz	2610	8	2659	51.6%
27	Pharaon2.62-128MB/P4-3Ghz	2608	8	2586	51.2%
28	Crafty18.12-116MB/P4-3Ghz	2607	8	2554	51.0%
29	Arasan8.3-128MB/P4-3Ghz	2604	8	2594	50.6%
30	Tao5.4-128MB/P4-3Ghz	2603	8	2590	50.5%
31	Ufim5.01-128MB/P4-3Ghz	2602	8	2515	50.2%
32	Yace0.99.87-128MB/P4-3Ghz	2599	8	2512	49.7%
33	Chessmaster9000-128MB/P4-3Ghz	2598	8	2635	49.8%
34	SOS.2forArena-128MB/P4-3Ghz	2598	8	2607	49.6%
35	List504-128MB/P4-3Ghz	2597	9	2392	49.4%
36	Ktulu3.8-128MB/P4-3Ghz	2594	8	2449	48.9%
37	LittleGoliathRevival-128MB/P4-3Ghz	2591	8	2640	48.8%
38	Arasan7.4-128MB/P4-3Ghz	2590	9	2376	48.3%
39	Fruit1.5-128MB/P4-3Ghz	2583	8	2583	47.4%
40	WildCat4.0-128MB/P4-3Ghz	2578	8	2458	46.7%
41	Dragon4.5-128MB/P4-3Ghz	2578	8	2459	46.6%
42	AnMon5.51-128MB/P4-3Ghz	2576	8	2470	46.5%
43	SmarThink0.16b-128MB/P4-3Ghz	2575	8	2557	46.3%
44	WildCat3.0-128MB/P4-3Ghz	2574	8	2374	46.1%
45	LittleGoliathNemesis-128MB/P4-3Ghz	2574	9	2429	46.1%
46	Gothmog0.3.0-128MB/P4-3Ghz	2573	8	2441	45.9%
47	GLC3.00-128MB/P4-3Ghz	2571	8	2466	45.6%
48	Yace0.99.50-128MB/P4-3Ghz	2569	8	2370	45.4%
49	Quark2.35-128MB/P4-3Ghz	2569	8	2429	45.4%
50	Chispa4.0.2-128MB/P4-3Ghz	2567	9	2424	44.9%
51	Delfi4.2-128MB/P4-3Ghz	2566	9	2400	45.0%
52	AnMon5.21-128MB/P4-3Ghz	2566	8	2323	44.8%
53	TheBaron1.2.1-100MB/P4-3Ghz	2565	8	2456	44.8%
54	Dragon4.2-128MB/P4-3Ghz	2563	9	2318	44.4%
55	KingOfKings2.56-128MB/P4-3Ghz	2561	9	2345	44.1%
56	KingOfKings2.02-128MB/P4-3Ghz	2560	8	2329	44.0%

```
 57 Ufim4.04-128MB/P4-3Ghz                : 2557    9   2301    43.6%
 58 Quark1.76-128MB/P4-3Ghz               : 2557    8   2504    43.6%
 59 GLC2.18-128MB/P4-3Ghz                 : 2549    9   2337    42.4%
 60 Eagle0.5.1-128MB/P4-3Ghz              : 2536    9   2335    40.5%
 61 KnightX1.91-128MB/P4-3Ghz             : 2531    8   2323    40.0%
 62 Eagle0.3-128MB/P4-3Ghz                : 2517   10   2222    37.8%
 63 KnightX1.77-128MB/P4-3Ghz             : 2513    9   2242    37.3%
 64 Chispa3.01-128MB/P4-3Ghz              : 2472    9   2144    31.8%
```

Bei dieser Rangliste ist ein Aspekt uninteressant, nämlich die *absoluten* ("Elo"-)Zahlen bzw. Abstände; in den Fokus geholt seien vielmehr die beiden bereits oben angesprochenen Momente:

a) die ungefähre Rang-"Gegend" und
b) der Versionen-Vergleich.

Punkt a) zeigt auf den ersten Blick eine durchaus plausible Reihung; die meisten Programme rangieren +/– dort, wo sie auch z.B. die Comp2004 oder andere einschlägige Ranglisten sehen.
Allerdings sind leider zugleich empfindliche Fehl-Rangierungen ersichtlich: Die beiden Juniors sind sicher nicht gleich stark, und bestimmt gehört das Chessmaster-Duo ins oberste Feld-Viertel. Noch die eine und andere Ungereimtheit mehr wäre zu entdecken, zöge man als testkritischer Erbsenzähler zu Felde.

Sehr viel zufriedener bin ich jedoch mit dem Ergebnis betreffs Punkt b), und hier arbeitet offenbar der *SwissTest2* zuverlässiger als jede andere mir geläufige Suite: nämlich bezüglich des Versionen-Vergleichs. Denn bis auf zwei Ausnahmen (Junior und Goliath) bildet der Test die Hierarchie der jeweiligen Paare perfekt ab. Das ist bei so wenigen Stellungen doch verblüffend – auch wenn man berücksichtigt, dass eben wirklich deutlich unterschiedliche Programm-Derivate zum Einsatz kamen.

Quo vadis, *SwissTest2*?

Zwar war das bei weitem zu viel der Ehre, als unlängst im CSS-Forum der List-Programmierer Fritz Reul – Schöpfer einer der weltweit stärksten Schach-Engines – urteilte, der *SwissTest2* sei "ein wirklich produktiver Beitrag für die Schachentwicklung". Aber es ist wohl kein Zufall, dass der Test insbesondere bei den Programmierern auf Interesse stößt, und selbstverständlich freute es den Autor, wenn diese Schachrätsel-Suite dazu beitragen könnte, der Programmierer-Gilde bei dem einen oder anderen kleinen Fortschritt ihrer Engines zu helfen.
Trotzdem, die vorliegende Zweitauflage des *SwissTest2* soll noch nicht das endgültige Wort sein: Eine dritte, finale Fassung wird (höchstwahrscheinlich) folgen, welche die aus dem Anwenderkreis stammenden Anregungen verarbeiten (und ggf. gar Aufgaben entfernen) soll, sodass die Aussagekraft bzw. die Zuverlässigkeit auch bezüglich der Rangierung neuer Programme nochmals erhöht werden kann.

Solange ist der *SwissTest2* dezidiert "work in progress" – für kreative Kritik sowohl an den Testverfahren wie auch an einzelnen Stellungen bzw. deren Analysen ist der Autor also stets offen, und betreffs etwaiger Test-News statte man seinen "SCHACH-NOTIZEN" gelegentliche Besuche ab. Auf alle Fälle hofft er, dass der SwissTest2 mindestens so viel Spaß bereite, wie er seinem Urheber Arbeit machte ... *(Walter Eigenmann)*

Informationen zum Autor:
Walter Eigenmann