

Elektronik in Hobby und Beruf

Oktober 1980 Nr. 10 DM 4,50

CHIP

Zeitschrift für Mikrocomputer-Technik

Jetzt mit
acht Seiten
Programm-
service

Osterreich öS 40, Schweiz sfr 5,00, Luxemburg lfr 81, Niederlande hfl 5,50

Weltsensation

**Basic-Computer
für die Tasche**

CHIP TEST

**Personal-Computer
CBM 3032**

Selbstbau

**Interface für
Minikassetten**

Der Programmier-Trick

**Taschenrechner-
Doping**

HiFi für Freaks

**Computergesteuerte
Receiver**

Explodierende Intelligenz

Schachcomputer
Das Geheimnis ihrer Erfolge

CHIP-Wissen
**Wie Computer
sprechen**

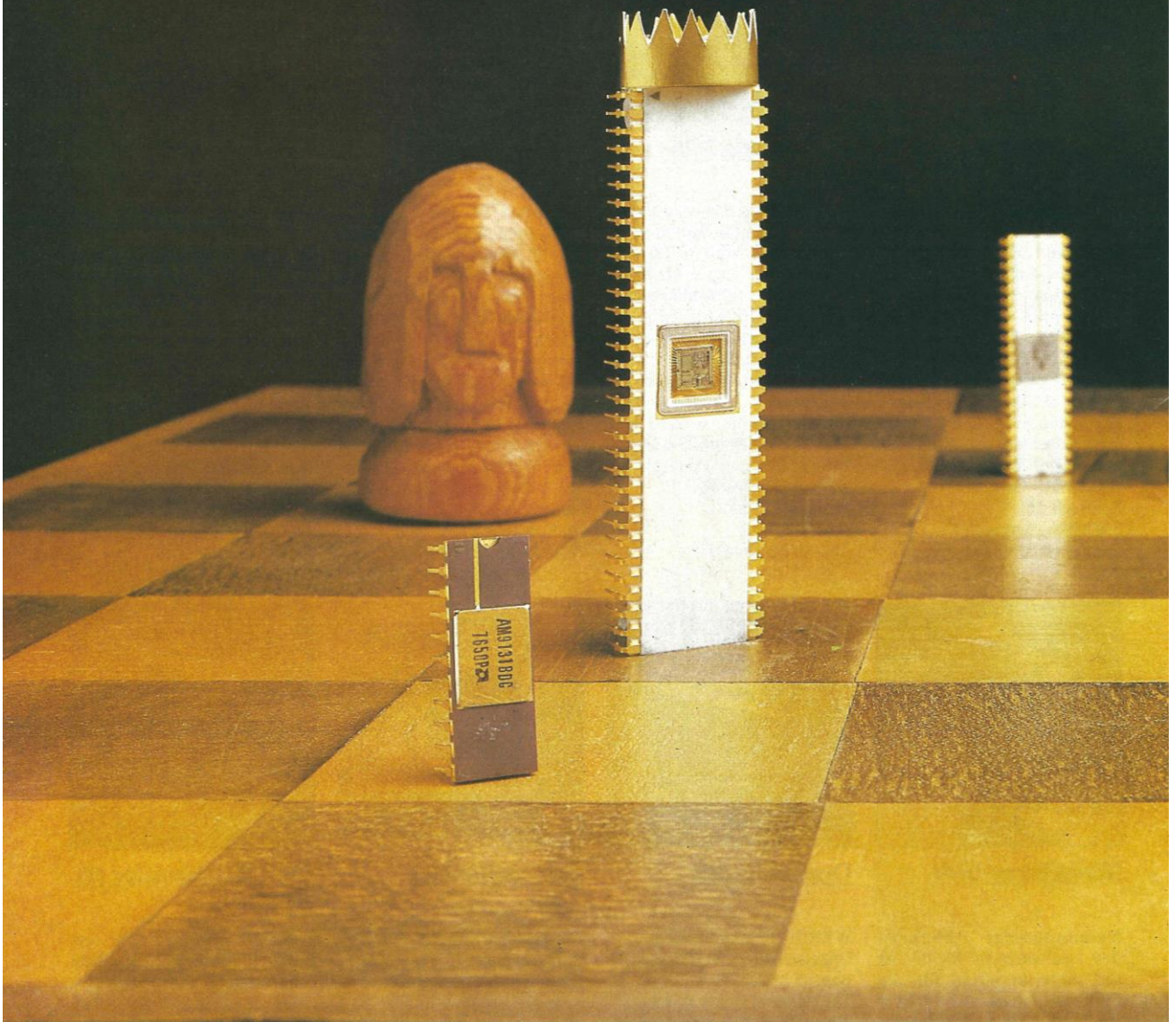
Wie Schachcomputer funktionieren

(Quelle: www.chip.de/ - Computermagazin Chip Nr. 10 - Oktober 1980) (photo copyright © by www.schaakcomputers.nl/) (600 dpi)

Titelstory

Wie Schachcomputer funktionieren

Kampf



Nirgendwo gibt es sie handgreiflicher zu sehen, die künstliche Intelligenz, als beim Spiel gegen den Schachcomputer. Zwei Preisträger aus dem Wettbewerb „Jugend forscht“ zeigen – wie die Schachcomputer siegen oder verlieren. Der Kampf der Könige wird durch Menschen gewonnen, denn ein Schachcomputer ist so stark, wie sein Programm – und das hat ein Programmierer gemacht.

Wie Schachcomputer funktionieren

(Quelle: www.chip.de/ - Computermagazin Chip Nr. 10 - Oktober 1980) (photo copyright © by www.schaakcomputers.nl/) (600 dpi)

der Könige



Schach, das königliche Spiel, war immer schon ein Testfeld für Intelligenz. Gute Schachspieler, so meint man, denken auch im Leben präzise und erfolgreich. Manchem mag dieser naheliegende Schluß gewagt erscheinen, fest steht, daß das Schachspielen die kleinen grauen Zellen sehr beansprucht. Das Schachbrett und die Spielregeln zeigen auch, weshalb das so ist. Das Spielfeld besitzt 64 Felder, zunächst nicht viel.

Wie schnell man aber damit in unübersichtbare Situationen gerät, schildert die Sage vom indischen Weisen, der sich, als Belohnung für eine gute Tat, von seinem König nichts anderes erbat als Reiskörner. Und zwar auf das erste Feld des Schachspieles eins, auf das zweite zwei, auf das dritte vier, auf das vierte acht... Jeweils die doppelte Menge als auf dem vorhergehenden Feld wollte er haben, das erschien ihm angemessen für sich.

Wie Schachcomputer funktionieren

(Quelle: www.chip.de/ - Computermagazin Chip Nr. 10 - Oktober 1980) (photo copyright © by www.schaakcomputers.nl/) (600 dpi)

lungstiefe und der Güte der Bewertungsfunktion.

Das Minimaxprinzip

Um den besten Zug zu finden, wird das sogenannte Minimaxprinzip angewendet. Dieses Prinzip besagt, daß die am Zug befindliche Seite bestrebt ist, den größtmöglichen (maximalen) Gewinn des Gegners möglichst herabzusetzen (zu minimieren). Das Programm geht davon aus, daß die Seite, die am Zug ist, an jedem Knoten den für sie besten Zug wählt. Ein Beispiel:

Den Endstellungen im Spielbaum werden die Bewertungen zugeordnet, die zu jeder Endstellung gehören (Bild Beispielbaum oben):

Der Computer mustert die Zugmöglichkeiten durch und stellt fest, daß er von der zweiten Ebene aus, wenn er selbst am Zuge ist, je nach Verlauf der Partie eine Endstellung mit der Bewertung 5, 9, 12 oder 15 immer erreichen kann. Denn auf einen der vier Punkte der zweiten Ebene muß ihn der Gegner ja bringen (Bildmitte):

Der Computer kann den Gegner von der nullten Ebene aus in jede der beiden Lagen der ersten Ebene bringen. Wählt der Computer den rechten Zweig, dann kann der Gegner in der ersten Ebene ebenfalls den rechten Zug wählen und hat damit verhindert, daß der Computer neun „Gewinnpunkte“ einstreicht. Wählt der Computer den linken Zweig in der nullten Ebene, dann kann der Gegner in der ersten Ebene immerhin noch verhindern, daß der Computer in der Endstellung 15 Gewinnpunkte einstreicht. Der Gegner sollte also auch hier den rechten Zug wählen.

In der ersten Ebene kann der Gegner des Computers also den Gewinn des Computers jeweils minimieren. In der nullten Ebene sollte der Computer also das Maximum der vom Gegner „minimierten“ Gewinnpunkte durch seine Zugwahl zu erreichen versuchen. Der Computer sollte also links ziehen, dann kann er auf 12 Punkte kommen. Der Gegner wird dann, wenn er den Schaden für den Computer wirklich groß machen will, rechts ziehen, damit der Computer nicht 15 Punkte einstreicht. Der Computer wählt ebenfalls rechts, um auf die zwölf Punkte zu kommen.

Der Alpha-Beta-Algorithmus

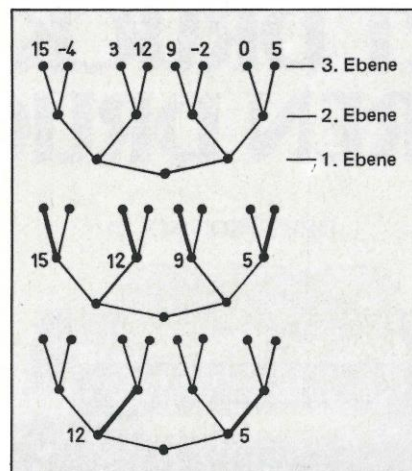
Im Jahre 1958 entwickelten einige Wissenschaftler der Carnegie-Mellon-Universität in Pittsburgh den sogenannten Alpha-Beta-Algorithmus. Mit diesem Verfahren reduziert sich die Anzahl der

zu analysierenden Endstellungen im Spielbaum bei gleichbleibendem Endergebnis um ein Vielfaches.

Im Durchschnitt gehen bei einem Schachspiel von einem Punkt des Spielbaumes 38 Zugmöglichkeiten aus und nicht zwei, wie in dem Beispiel vereinfachend angenommen.

Bei einer Verschachtelungstiefe von zwei Halbzügen, also Zug und Gegenzug, untersucht das Programm seine erste Zugmöglichkeit und die darauf folgenden durchschnittlich 38 Gegenzüge und bewertet die so entstehenden Endstellungen. Das Programm erwartet dabei immer den besten Gegenzug des Gegners. Dann prüft der Computer seine zweite Zugvariante und stellt beispielsweise fest, daß der vierte Gegenzug auf diese zweite Möglichkeit dem Computer einen größeren Nachteil bringen würde, als der beste Gegenzug zur ersten Variante.

Es ist daher nicht mehr nötig, weitere Antworten auf den zweiten Zug des



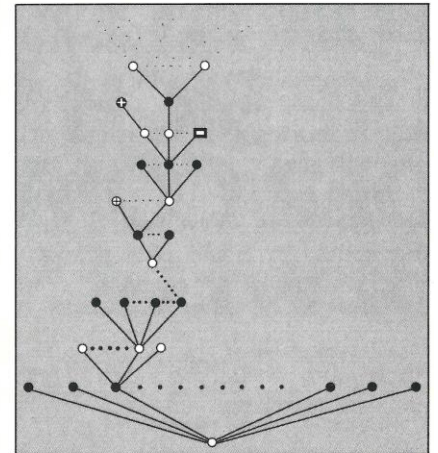
Ein Beispielbaum, vereinfacht mit nur je zwei Zugmöglichkeiten auf jeder Ebene. Die Zahlen geben die vom Programmierer eingespeicherte Bewertung der an diesem Punkt bestehenden Stellung wieder

Computers zu untersuchen, da nach dem vierten Gegenzug aufgrund des Minimaxprinzips die zweite Variante des Computers schon ausgeschlossen ist. Das Programm kann also sofort zur Analyse der dritten Variante übergehen.

Unter optimalen Bedingungen reduziert sich die Zahl der zu bewertenden Endstellungen bei einer Tiefe T und Z Zugmöglichkeiten für jede Seite von Z^T bis auf maximal $2\sqrt{Z^T}$. Der Wert für den besten Zug des Computers wurde als Alpha, der Wert für den besten Zug des

Gegners als Beta bezeichnet, wodurch das Verfahren seinen Namen erhielt.

Der Alpha-Beta-Algorithmus findet sowohl bei der „A-Strategie“ als auch bei der „B-Strategie“ Verwendung. Die Wirksamkeit des Verfahrens läßt sich erhöhen, wenn Weiß und Schwarz die für sie besten Züge zuerst ausführen. Bei



Der Spielbaum des Schachspiels. Die meisten Zweige sind weggelassen. Ein Spiel ist ein Weg durch den Baum, wobei die weißen Kreise symbolisieren, daß Weiß am Zuge ist, was bedeutet, daß Weiß einen Zweig wählen kann. Für Schwarz gilt Entsprechendes

Programmen, die nach dem Prinzip der „A-Strategie“ arbeiten, werden die Züge an jedem Knoten des Spielbaums mit Hilfe einer Vorbewertungsfunktion, die im wesentlichen nur materielle Gesichtspunkte berücksichtigt, vorsortiert. Bei Programmen der „B-Strategie“ werden die plausiblen Fortsetzungen ebenfalls sortiert und dann erst probiert.

Die Killerheuristik

Gesucht sind also beim Programmieren von Schachprogrammen Verfahren, die möglichst schnell zeigen, ob ein Probezug sinnvoll ist. Ein Gegenzug auf einen Probezug, der zeigt, daß damit nichts zu gewinnen ist, heißt Widerlegung. Oft kann es sein, daß die verschiedenen Probezüge durch ein und denselben Gegenzug widerlegt werden. Ein krasser Fall: Eine bedrohte Dame bleibt bedroht, wenn man sie nicht schützt, sondern nur marginale Bauern irgendwo zieht. All diese Bauernzüge werden durch den Killerzug gegen die Dame widerlegt. Während also die Shannonsche A-Strategie ganz stur alle Varianten bis zum bitteren Ende durchprüft und die B-Strategie wenigstens die nach erkennbaren

Wie Schachcomputer funktionieren

(Quelle: www.chip.de/ - Computermagazin Chip Nr. 10 - Oktober 1980) (photo copyright © by www.schaakcomputers.nl/) (600 dpi)

Gesichtspunkten sinnlosen Züge beim Prüfen wegläßt, führt der Alpha-Beta-Algorithmus ein dynamisches Prinzip in die Schachprogrammierung ein, mit dem die A- und B-Strategie verbessert werden können: Beim Probeziehen werden die Ergebnisse früherer Proben mit berücksichtigt, was ab der zweiten Probe Einsparungen bringen kann. Die Killerheuristik nun bringt, wenn Sie eingebaut ist in ein Schachprogramm, noch mehr Dynamik hinein. Ein Killerzug kann eine ganze Reihe von Probezügen sinnlos machen und sollte immer zuerst getestet werden.

Schachprogramme sind intelligent

Da den Programmen wegen ihrer begrenzten Verschachtelungstiefe ein langfristiger Plan fehlt, ist ihre strategische Spielstärke, verglichen mit ihrer taktischen, relativ gering. Dies wirkt sich insbesondere nachteilig im Endspiel aus. Um diese Tatsache während der Eröffnung zu vermeiden, sind die meisten Schachprogramme mit einer sogenannten Eröffnungsbibliothek ausgestattet, in der alle gängigen Eröffnungen gespeichert sind.

Trotzdem läßt sich nicht bestreiten, daß die Programme in bezug auf ihre Fähigkeit, Schach zu spielen, intelligent sind. Dies kann man belegen, indem man den sogenannten Turing-Test auf Schachprogramme anwendet. Dieser, vom englischen Mathematiker Alan Turing entwickelte „Intelligenztest für Computer“ besagt, daß einer Maschine Intelligenz zugesprochen werden muß, wenn bei der Lösung einer Aufgabe nicht mehr feststellbar ist, ob diese Leistung von einem Menschen oder von einem Computer erbracht worden ist.

Das Schachprogramm für „Jugend forscht“

Entwicklungsbeginn: 1979
 Sprache: Fortran IV
 Länge: 1000 Zeilen
 Rechner: TR 440
 Strategie: Shannon B und Shannon A

Verschachtelungstiefe: 3 Halbzüge
 Datenstruktur: Größtenteils Felder mit 8 Zeilen und 8 Spalten
 Feld FIG: entspricht dem Spielfeld und enthält in einer Position 1 für Bauer, 2 für Turm, 3 für Springer, 4 für Läufer, 5 für Dame, 6 für Kö-

nig; Computerfiguren sind positiv, Spielerfiguren negativ, leere Felder enthalten 0. Nicht rochierte Türme und Könige erhalten ± 8 aufaddiert
 Feld BEDVC: enthält die Anzahl der Bedrohungen auf ein Spielfeld durch Computerfiguren
 Feld BEDVG: Bedrohungen durch Spielerfiguren
 Feld BEDGEG: enthält die Anzahl von gegnerischen Figuren, die von diesem Feld aus bedroht werden
 Feld ZUGM: enthält die Anzahl der Zugmöglichkeiten, die die Figur dort hat

Baumsuche

Nachdem der Spieler einen Zug eingegeben hat und dieser ausgeführt wurde, hat das Programm die Aufgabe, selbst einen Zug zu berechnen. Das Hauptprogramm ruft dazu das Unterprogramm COMZUG auf. Zuerst werden durch einen Aufruf des Unterprogramms ZUGFC alle legalen Zugmöglichkeiten des Computers erzeugt und in einem Feld gespeichert.

Anschließend werden alle Züge der Reihe nach ausgeführt, vorberechnet (hierzu wird die gleiche Bewertungsfunktion wie für die Bewertung der Endstellungen verwendet) und zurückgesetzt. Zu jedem Zug wird die Vorbewertung gespeichert. Die so vorsortierten Züge werden danach ein zweites Mal ausgeführt (die besten Züge zuerst).

Vor dem Zurücksetzen wird das Unterprogramm TIEFE2 aufgerufen. Dort werden zuerst wieder alle möglichen Gegenzüge erstellt und vorberechnet. Dann werden auch hier die Züge ein zweites Mal ausgeführt, und zwar die besten Gegenzüge zuerst. Vor dem Zurücksetzen wird von hier das Unterprogramm TIEFE3 aufgerufen. Diese Verschachtelung der Unterprogramme ließe sich theoretisch immer weiter fortsetzen. Bei unserem Programm ist jedoch TIEFE3 die Endebene bei der Verschachtelung.

Im Unterprogramm TIEFE3 werden zunächst alle Züge erstellt. Nun wird, falls möglich, der beste Zug vom vorhergehenden Aufruf von TIEFE3 ausgeführt (Killerheuristik). Anschließend werden alle Schlagzüge und Bauernumwandlungen und danach alle übrigen Züge ausgeführt. Das Programm merkt sich die beste Bewertung und gibt diese an TIEFE2 zurück. Das Programm prüft jedoch nach jeder Bewertung, ob es den be-

schriebenen Alpha-Beta-Algorithmus anwenden und so die Anzahl der zu bewertenden Stellungen stark reduzieren kann.



Schachgroßmeister Petrossjan im Kampf gegen ein Computerprogramm namens Black Knight. Die Partie endete Remis

In TIEFE2 merkt sich das Programm die kleinste Bewertung, die es aus TIEFE3 zurückgemeldet bekam und gibt sie nach COMZUG zurück. Auch hier wird, wenn möglich, der Alpha-Beta-Algorithmus angewendet. Im Unterprogramm COMZUG merkt sich das Programm wieder wie in TIEFE3 die beste Bewertung und zusätzlich noch den dazugehörigen Zug. Diesen Zug führt das Programm dann als seinen endgültigen Zug aus und gibt ihn zusammen mit dem aktuellen Spielfeld auf der Konsole aus.

Bewertung

Die Bewertungsfunktionen umfassen taktische und strategische Aspekte. Die Bewertungskriterien (Heuristiken) bauen auf allgemeinen Erfahrungsregeln im Schachspiel auf. Das wichtigste Kriterium ist die Bewertung des Materials und der Schlagwechsel und Bedrohungen. Für das Material gilt: Bauer = 300, Springer = 975, Läufer = 1050, Turm = 1500 und Dame = 2700 Punkte. Vom Material her identische Stellungen werden durch die Positionsbewertung der übrigen Heuristiken (z.B. Entwicklung der leichten Figuren, Beherrschung der Brettmitte, Gesamtzahl der Zugmöglichkeiten usw.) unterschieden. Die Positionsbewertung macht maximal etwa den Wert eines Bauern aus.

Die Bewertung von Stellungen im Spielbaum übernehmen zwei Unterprogramme: BEWFC und BEWFS. Das erste bewertet Stellungen, die nach Zügen der Computerfiguren entstanden sind. Das andere Unterprogramm bewertet Stellungen, die nach Zügen des Gegners (der Spielerfiguren) im Spielbaum auftreten. Die Bewertungsfunktionen werden sowohl für die Vorbewertungen als auch für die Bewertung von Endstellungen verwendet. Vor jedem Aufruf eines der

Wie Schachcomputer funktionieren

(Quelle: www.chip.de/ - Computermagazin Chip Nr. 10 - Oktober 1980) (photo copyright © by www.schaakcomputers.nl/) (600 dpi)

Name	Land	Verfasser	Computer
Mychess	USA	D. Kittinger	Cromeco
Duchess	USA	T. Truscott B. Wright E. Jensen	?
Master	GBR	J. Birmingham P. Kent	IBM 3033
Parvell	BRD	T. Nitsche E. Heum W. Wolff	Siemens SMS 2
Sargon 3.0	USA	D. u. C. Spracklen	Apple II
Chess 5.0	USA	D. Slate W. Blanchard	Großrechner
Bibi Belle	USA	T. Scherzer K. Thomson J. Condon	Eigenbaurechner PDP 11/70
Schach 2.3	BRD	M. Engelbach	Burroughs 7800
Kaissa	UdSSR	M. Ponskoy	IBM
Chess 4.9	USA	L. Atkin D. Cahlander	CDC Cyber 176
Chaos	USA	F. Swartz	Amdahl
BCP	GBR	D. Beal	PDP 11/70
Lexcentrique	Kanada	C. Jarry	Amdahl 470
Ostrich	Kanada	M. Newborn	Data General Nova 3
Avit	Kanada	T. Marsland	Amdahl 470

beiden Unterprogramme (also bei der Bewertung einer neuen Stellung) erfolgt ein Aufruf des Unterprogramms AENDER. Dadurch werden die aufgeführten Felder und Variablen, die die Grundlage für die Bewertung bilden, überarbeitet und auf den aktuellen Stand gebracht. Schlagwechsel in den Bewertungsfunktionen werden bei uns teils durchgerechnet, teils „durchgeschätzt“. Das Programm verkalkuliert sich dabei äußerst selten. Wie schon erwähnt, verwendet unser Programm zwei getrennte Bewertungsfunktionen. Dies ist aus dem einfachen Grund notwendig, da es bei der Bewertung von Bedrohungen und Schlagwechseln ein großer Unterschied ist, welche Seite am Zug ist. Ein Beispiel: Bei einer zu bewertenden Stellung im Spielbaum wäre der Gegner des Computers am Zug. Der Gegner bedroht einen nicht gedeckten Bauern des Computers. Der Computer muß damit rechnen, daß der Bauer beim nächsten Zug geschlagen wird. Aus diesem Grund bekommt er für diesen bedrohten Bauern 300 Minuspunkte. Würde der Computer in der gleichen Stellung einen Bauern des Gegners bedrohen, so würde er hierfür nur sehr wenig oder gar keine Punkte (abhängig von den Zugmöglichkeiten dieses Bauern) erhalten, weil der Gegner am Zug wäre, und der Computer damit rechnen muß, daß sich der Gegner mit seinem nächsten Zug der Bedrohung entzieht oder den Bauern deckt. Ähnliches gilt für mehrfach bedrohte und gedeckte Figuren (Schlagwechsel). Ein weiterer Vorteil bei dieser Methode ist, daß man Matt und Patt in den Bewer-

tungsunterprogrammen abfragen kann, da die Felder für die Bedrohungen und Zugmöglichkeiten der einzelnen Figuren zur Verfügung stehen. Im Gegensatz dazu „sieht“ CHESS 4.8 ein Matt dadurch, daß es beim Versuch weiterzuverschachteln, für die eigentlich am Zug befindliche Seite keine legalen Zugmöglichkeiten findet. Dies bedeutet beispielsweise, daß CHESS 4.8 vier Halbzüge tief verschachteln muß, um ein zweizügiges Matt mit Sicherheit zu erkennen, während dieses Programm mit drei Halbzügen auskommt.

Auf der Rechenanlage, auf der das Programm zur Zeit läuft, bewertet es 10 bis 30 Stellungen pro Sekunde. Bei einer Verschachtelungstiefe von zwei Halbzügen nach der A-Strategie bewertet das Programm inklusive der Vorbewertungen ca. 150 Stellungen, um seinen endgültigen Zug zu finden. Bei einer Verschachtelungstiefe von drei Halbzügen sind es insgesamt 2500 bis 3000 Stellungen. Während der ersten Züge versucht das Programm, die jeweilige Ausgangsstellung in der Eröffnungsbibliothek zu finden. Ist die Stellung gespeichert, so wird der dazu vermerkte Zug sofort ausgeführt. Die Eröffnungsbibliothek enthält derzeit 215 verschiedene Stellungen aus 20 verschiedenen Schacheröffnungen.

Menschen gegen Maschinen

Am 11. 12. 1979 spielte der Schotte David Levy im Hamburger ZDF-Studio in einem Schaukampf für Millionen gegen einen Riesenrechner namens Cyber 176. Levy, internationaler Schachmeister,

wettete 1968 gegen eine Gruppe Wissenschaftler aus dem Gebiet „Artificial Intelligence“, daß er zehn Jahre lang unter Turnierbedingungen nicht von einem Computer geschlagen würde. 1250 englische Pfund waren eingesetzt. Levy gewann diese Wette 1978. Er spielte seine letzte Partie in dieser Wette gegen das Programm Chess 4.7, gegen das er im Turnier drei Partien gewann, eine unentschieden spielte und eine verlor. In Hamburg gelang es Levy nicht, einen Sieg zu erringen. Nach vier Stunden Kampf mit allen Raffinessen mußte Levy ein Remis akzeptieren. Das Programm, das ihm dieses Unentschieden abtrotzte, hieß Chess 4.8, eine Weiterentwicklung des 4.7. Der Computer stand übrigens in den USA, in Minneapolis, von wo aus über Satellit die Kommandos an einen Roboterarm in Hamburg abgesandt wurden. Dies ist die derzeitige Spitze des Wettkampfes Mensch gegen Maschine. Angefangen hatte es 1952, als der Mathematiker und Computerpionier A. M. Turing das Schachprogramm Turochamp zum ersten Mal gegen einen Menschen antreten ließ. Der Mensch, obwohl nur mittelmäßig spielend, gewann.

Computer gegen Computer

Das erste Turnier der Denkmassen war ein 1966/67 durchgeführter Vergleichskampf IBM 7090 (USA) gegen M-20 (UdSSR). Der Sowjetrusse gewann mit 3 : 1 Punkten. Dann aber ging es in den USA richtig los mit der Entwicklung. 1970 wurden die ersten US-Schachcomputermeisterschaften ausgetragen, die jetzt regelmäßig durchgeführt werden. Chess 3.0 hieß der Champion der ersten Jahre, ein von den Studenten Davis Slate, Larry Atkin und Keith Gorlen geschriebenes Programm. Es wurde bis heute laufend verbessert von 3.0, 3.5, 3.6, 4.0... bis zum heute vorhandenen Chess 5.0. Dieses Programm beherrschte die Szene, zumindest die westliche, denn 1974, auf der ersten Schachweltmeisterschaft, wurde es von dem Russischen Programm Kaissa geschlagen. Kaissa läuft übrigens auf einer IBM 370. Kaissa wurde 1974 Weltmeister. Auf der zweiten Weltmeisterschaft der Schachprogramme in Toronto in Kanada 1977 konnte Chess sich endlich auch auf Weltenebene durchsetzen. Die Version 4.6 gewann – allerdings ohne mit Kaissa direkt kämpfen zu müssen, denn Kaissa wurde von einem US-Programm namens Duchess aus der Duke University aus dem Feld geschlagen. ☐

Wolfgang Stahn
Kai Hunstedt

Wie Schachcomputer funktionieren

(Quelle: www.chip.de/ - Computermagazin Chip Nr. 10 - Oktober 1980) (photo copyright © by www.schaakcomputers.nl/) (600 dpi)