

Die Kunst des Nichtstuns

Dr. Christian Donninger über den Null-Move-Algorithmus

„Wir würden gar vieles besser kennen, wenn wir es nicht zu genau erkennen wollten“ (J. W. Goethe, „Erfahrung und Wissenschaft“). Schachprogramme haben noch immer den Ruf, mit brutaler Gewalt (brute force) ihre Siege zu erringen. Es heißt, sie rechnen bis zu einer bestimmten Tiefe stur alle möglichen Varianten durch und nützen taktische Fehler des menschlichen Gegners aus. Doch allein damit sind heutzutage keine Lorbeeren mehr zu gewinnen. Programme bzw. ihre Autoren müssen sich schon ein „klügeres“ Vorgehen einfallen lassen.

Kein einziges kommerzielles Spitzenprogramm rechnet heute mehr nach dem reinen brute-force-Schema. Auf Suchtiefe 7 werden in meinem Programm Nimzo-Guernica manche Varianten bereits nach 3 Halbzügen abgebrochen, andere aber bis zu 20 Halbzüge lang untersucht. Im schlimmsten Fall kann das Programm auf dieser Tiefe ein Matt in 2 übersehen, bei günstiger Konstellation der Sterne und der Figuren ein Matt in 10 ankündigen.

Andere Programme sind da noch viel extremer. So erschwindelte sich The King auf der Mikro-WM '93 im Spiel gegen Nimzo durch ein Patt in 16 noch einen halben Punkt. Auch The King schneidet auf Tiefe sieben manche Varianten bereits nach drei Halbzügen ab. Im ersten Aufeinandertreffen King-Nimzo, Maastricht 1991, hat er auch tatsächlich eine simple zweizügige Drohung übersehen.

Die Grenzen des Wachstums

Laut H. Qualtinger ist im realen Leben Simmering gegen Kapfenberg der Inbegriff von Brutalität. Im Computerschach besteht sie darin, auf jeden der typischerweise 40 möglichen Züge alle 40 möglichen Antworten, die Antworten auf die Antwort usw. zu betrachten. Bereits bei einer Suchtiefe von drei müßte man mit dieser Methode 64000 (40x40x40) Stellungen untersuchen. Da man aber mitten in einer Abtauschsequenz oder bei Schachgeboten nicht einfach zu rechnen aufhören kann, ergeben sich nach dieser Methode schon auf Suchtiefe drei ein paar hunderttausend Stellungen. Selbst auf einem Pentium würde ein derartiges Programm nie über Suchtiefe vier hinauskommen. Ein nach diesem Schema rechnendes „primitiv-brutales“ Programm ist daher auch nie geschrieben worden.

Der Alpha-Beta-Algorithmus

Mit Hilfe des Alpha-Beta-Algorithmus läßt sich aber die Methode verfeinern und damit wesentlich effektiver machen. Die Grundidee von Alpha-Beta ist sehr simpel: Angenommen, wir führen in der Ausgangsstellung eine Suche der Tiefe zwei durch. Nach 1.e2-e4 werden wieder alle möglichen Antworten durchsucht und ...e7-e5 als beste Antwort mit der Bewertung 0.0 gefunden. Untersucht man nun etwa den schlechten Zug 1.h2-h4, wird man die zuvor gefundene Antwort e7-e5 zuerst ausprobieren und die Stellung z.B. mit -0.3 aus der Sicht von Weiß bewerten. Obwohl noch bessere Antworten auf 1.h4 gefunden werden könnten, ist es nicht mehr notwendig, diese tatsächlich zu berechnen. 1.e4 ist (innerhalb des Suchhorizontes) auf alle Fälle besser als 1.h4. Man sagt: 1...e5 widerlegt h4. Eine Reihe von anderen Anfangszügen können ebenfalls sofort mit 1...e5 widerlegt werden, es ist auch in diesen Fällen nicht notwendig, die restlichen 19 möglichen Antworten durchzuackern.

Auf Tiefe drei schaut die Sache schon etwas anders aus. Angenommen, 1.e4 e5 2.Sf3 wird mit 0.1 für Weiß bewertet. Untersucht man nun 1.h4 e5, so muß Weiß versuchen, e5 zu widerlegen und die bisher beste Bewertung von 0.1 zu verbessern. Weiß wird daher alle möglichen Züge ausprobieren und anschließend feststellen, daß sich 1...e5 nicht widerlegen läßt und h4 somit ein schlechter Zug ist. Auf Tiefe vier wird wiederum Schwarz in der Regel ein einziger guter Zug (z.B. d5) genügen, um jede mögliche Antwort von Weiß auf 1...e5 zu widerlegen.

Eine typische Alpha-Beta-Suchfolge verläuft nach folgendem Muster: Schlechter Zug – eine gute Widerlegung – alle Züge, die die Widerlegung vergeblich zu widerlegen versuchen – eine gute Widerlegung ...

Man braucht daher nur auf ungeradzahligem Suchtiefe alle möglichen Züge zu untersuchen, während auf geradzahligem Suchtiefe ein einziger guter Zug als Widerlegung genügt. Alpha-Beta kommt also bei gleicher Anzahl von Positionen doppelt so tief, oder (was dasselbe ist): Die Positionen nehmen nicht mehr von einer Ebene zur nächsten mit dem Faktor 40, sondern nur noch mit der Wurzel aus 40 zu.

Allerdings läßt sich diese erhebliche Verbesserung nur erreichen, wenn auf den geraden Zugtiefen im-

mer eine gute bzw. die beste Widerlegung als erster Zug ausprobiert wird. Z.B. wird auf 1.h4 der Zug 1...h5 oder 1...g5 nicht als Widerlegung genügen. Würde aber ein Programm immer sofort die beste Antwort finden, könnte man sich die Suche überhaupt ersparen und bereits in der Ausgangsstellung diesen besten Zug spielen. Durch zahlreiche Tricks erreichen die meisten Programme eine Trefferquote von ca. 90%, und der Zuwachs von einer Suchtiefe zur nächsten würde ungefähr den Wert 7 betragen, also sehr nahe beim theoretisch erreichbaren Optimum sein.

Der Unterschied zwischen gerader und ungerader Suchtiefe erklärt auch, warum Programme etwa vom fünften in den sechsten Halbzug schneller kommen als vom sechsten in den siebten. Allerdings wird der Effekt durch Erweiterungen und die im folgenden beschriebenen selektiven Techniken etwas verwischt.

Trotz dieser erheblichen Einsparung übersieht ein mit Alpha-Beta rechnendes Programm innerhalb des Suchhorizontes tatsächlich nichts. Bis zum heutigen Tag versuchen auf Supercomputern und Spezial-

hardware laufende Programme mit dieser Methode den Gegner niederzuwalzen.

Bekanntlich gewann Ed Schroeder jedoch mit seinem Mikro-Computer-Programm Gideon bei der Weltmeisterschaft '92 in Madrid den Titel noch vor den schweren Jungs auf ihren Großrechnern. Abgesehen von der Notwendigkeit einer besseren positionellen Bewertung läßt sich dies nur bewerkstelligen, wenn man den Suchprozeß noch effektiver als Alpha-Beta gestaltet. Tatsächlich wächst die Anzahl der Positionen von einer Suchtiefe auf die nächste bei den meisten guten kommerziellen Programmen um den Faktor 4 bis 5; sie sind damit also wesentlich besser als das theoretische Optimum der intelligenten Brutalität. Der großmeisterliche Verzweigungsfaktor liegt nach den Studien des holländischen Psychologen de Groot übrigens bei 1,7.

Das Zauberwort heißt Selektivität. Die Berechnung wird für viele Varianten bereits vor der vollen Suchtiefe beendet. Im Prinzip kann man dabei zwischen einer auf Wissen basierenden statischen Selektivität und einer sich auf den Suchprozess stützenden dynamischen Selektivität unterscheiden.

... immer die neueste Schach-Software ...

NIMZO-Guernica

CHESBASE total

CHES GENIUS 2.0

Mephisto GIDEON

CHES ASSISTANT SOCRATES 3.0 HIARCS 2.1

CHES FRIEND ZARKOV 3.0 L-CHES WIN

CHESMASTER 4000 TURBO

M-CHES PRO 3.5

GAMBIT-SOFT

Feckenhauser Str.27
D-78628 Rottweil



Tel: 0741-21573
Fax: 0741-15217

Positionelle Selektivität

Im obigen Beispiel wurde der Zug 1.h4 durch e5 widerlegt. Angenommen, die Stellung wird nach 1.h4 von der statischen Bewertungsfunktion etwa mit -0.1 aus der Sicht von Weiß eingeschätzt. Normalerweise ist das Zugrecht ein Vorteil, wir können also mit ziemlicher Sicherheit erwarten, daß Schwarz irgendeinen Zug hat, der diese Bewertung für Weiß noch weiter verschlechtert. Da wir nur beweisen wollen, daß h4 schlechter ist als der bisher beste Zug e4 mit der Bewertung 0.0 , ist es damit nicht mehr notwendig, einen schwarzen Gegenzug auf h4 auszuprobieren.

In diesem Fall haben wir nicht viel gewonnen und nur eine Position eingespart. In der Version 1 von Nimzo war dieses Prinzip aber noch weiter ausgebaut, es wurden Varianten nach folgender Regel frühzeitig beendet: Wenn die noch verbleibende Suchtiefe bis zum Horizont kleiner gleich drei ist und außerdem die Seite mit Zugrecht klar im Vorteil ist und außerdem keine hängenden Figuren hat, so beende die Suche. D.h., auch auf der Suchtiefe vier würde der Suchprozeß bereits nach 1.h4 abgebrochen werden. Natürlich darf man die Suche nicht beenden, wenn die Seite mit Zugrecht zwar momentan im Vorteil ist, aber hängende, möglicherweise nicht mehr zu rettende Figuren besitzt. (Ein bedrohter König wird hier auch als hängende Figur betrachtet. Ebenso ist ein Freibauer des Gegners quasi eine bedrohte eigene Dame).

Diese Regel besitzt zwei Schwachpunkte. Erstens muß die Bewertungsfunktion ziemlich umfangreich und damit langsam sein, um die wichtigsten Bedrohungen auch zu erkennen. Ein schwieriges Problem stellen z.B. überladene oder eingesperrte, aber im Moment noch nicht direkt bedrohte Figuren dar.

Zweitens versagt die Regel genau in jenen Situationen, in denen man sie am meisten braucht. In scharfen taktischen Stellungen hängt fast immer eine Figur, und die Wahrscheinlichkeit, die Suche frühzeitig beenden zu können, sinkt daher erheblich. Man könnte das Kriterium „es darf keine Figur hängen“ abschwächen. Z.B.: Wenn die Seite mit Zugrecht bereits eine Figur im Vorteil ist, darf maximal ein Bauer hängen. Versuche mit dieser Regel endeten aber im Fiasko. Ein Beispiel: Der weiße unbewegliche Turm auf a1 kann vom Läufer auf g7 gefressen werden. Weiß wird nun, wenn möglich, nach obiger Regel einen ungeschützten Bauern auf d4 stellen, die Bedrohung des Turmes damit scheinbar abgewehrt haben und die Suche beenden. Es wird sich über die von Schwarz zur Erzielung dieser Stellung „geopferte“ Figur freuen. In Wirklichkeit kann sich aber auch der Turm bereits seinen Holzpyjama anmessen lassen.

Wahrscheinlich kann man eine Bewertungsfunktion schreiben, die auch diesen Fall erkennt, aber das Ganze wird damit immer komplexer und damit auch langsam.

Wie man zu einer Idee kommt

Auf der holländischen Meisterschaft '91 spielte ich mit Nimzo Version 1 gegen den Fritz-Zwillingsbruder Quest von Frans Morsch. Obwohl Quest am Anfang irgendwie komisch spielte, wurde Nimzo schlußendlich ganz furchtbar zerlegt. Ich war ziemlich beeindruckt und Frans Morsch erzählte mir, daß das Programm, obwohl nur mit einer einfachen Bewertungsfunktion ausgestattet, eine hoch-selektive Suchtechnik verwende. Diese Suchtechnik basiere auf einer rekursiven Verwendung des Null-Zuges.

Trotz der imposanten Demonstration von Quest verfolgte ich die Idee nicht weiter, sondern verstaute sie irgendwo in einem Abteil meines Oberstübchens. Schließlich wollte ich ein „intelligentes“ Programm schreiben und nicht mit irgendeiner Technik arbeiten, die kein Schachwissen benötigt und daher auch auf dem Controller einer Waschmaschine implementiert werden könnte (zumindest, wenn man Frans Morsch heißt). Nach einem Jahr vergeblichen Bemühens, die selektive Suche weiterzuentwickeln, ließ ich jedoch von meinem eitlen Wahn und begann, mit dem Nichtstun zu experimentieren.

Die Null-Zug-Technik

Effektiv suchen heißt: schlechte, irrelevante Züge so schnell wie möglich zu widerlegen. Angenommen, Schwarz führt nach 1.h4 statt e5 irgendeinen Pseudo- oder Null-Zug aus (z.B. Th8-h8) und übergibt wie gewöhnlich das Zugrecht an Weiß. Weiß kann damit de facto zwei Züge hintereinander ausführen. Gelingt es nun Weiß nicht, eine positive Bewertung zu erreichen, so hat der Null-Zug 1.h4 bereits widerlegt, und wir können die Berechnung beenden.

Auf den ersten Blick ist nur der Zug e5 durch den Null-Zug Th8-h8 ersetzt worden, und somit keine Beschleunigung des Suchprozesses erkennbar. Die Bewertungsfunktion könnte die Stellung nach 1.h4 Th8-h8 2.e4 auch besser als die bisher beste Variante 1.e4 e5 2.Sf3 beurteilen. In diesem Fall müßten auch alle Varianten mit 1.h4 e5 berechnet werden, um h4 zu widerlegen. Die Suche würde dadurch sogar länger dauern. Man muß den Null-Zug daher mit einer Reduzierung der weiteren Suchtiefe verbinden.

Angenommen, wir untersuchen wieder das leidige 1.h4 auf Suchtiefe fünf. In einer normalen Suche würden auf 1.h4 e5 noch drei weitere Halbzüge plus eine anschließende Ruhesuche durchgerechnet. Bei

optimalem Alpha-Beta sind das noch ca. 500 Positionen, plus für jede dieser Positionen eine entsprechende Ruhesuche, bei der beispielsweise unklare Abtauschsituationen aufgelöst werden. In der Summe sind dafür mindestens 1000 Positionen notwendig. Nach dem Null-Zug 1...Th8-h8 wird der Suchbaum um zwei Halbzüge verkürzt, es werden also nur noch die unmittelbaren Antworten von Weiß betrachtet, und anschließend beginnt direkt die Ruhesuche. In diesem Fall werden insgesamt ca. 50 Positionen verwendet. Widerlegt der Null-Zug 1.h4, so haben wir die Suche 20 mal so schnell durchgeführt wie mit normalem Alpha-Beta. Wird 1.h4 von Th8-h8 nicht widerlegt, so beträgt der zusätzliche Aufwand des Null-Zuges nur fünf Prozent.

Wann sollte man nichts tun?

Prinzipiell wird der Null-Zug in jeder Position des Suchbaumes als erster Zug ausprobiert. In einer Variante können damit mehrere Null-Züge vorkommen (aus diesem Grund wird die Suchtechnik auch als „rekursiv“ bezeichnet). Wie immer im Leben gibt es dazu ein paar wichtige Ausnahmen:

- Der vorhergehende Zug war bereits ein Null-Zug. Ohne diese Regel würde die Suche etwa zu 1.h4 Th8-h8 2.Th1-h1 Th8-h8 3.Ta1-a1 ... degenerieren. Möglich ist aber 1.h4 Th8-h8 2.e4 Th8-h8 3.d4 ... Diese Zugfolge entspricht wegen der zweifachen Tiefenreduzierung durch den schwarzen Null-Zug einer Variante von neun Halbzügen, obwohl wir nur drei tatsächliche Halbzüge ausgeführt haben.
- Die Seite mit Zugrecht steht im Schach. Wenn man im Schach steht, nützt das Nichts. Th8-h8 wäre außerdem ein illegaler Zug.
- Die Position ist die Ausgangsstellung der Suche. Das Wissen über die Bewertung des Null-Zuges hilft bei der Auswahl der anderen Züge in der Regel nicht weiter.
- Die Seite mit Zugrecht steht sehr schlecht. Es ist daher unwahrscheinlich, daß der Null-Zug den letzten Zug der Gegenseite widerlegt. Die Nullsuche wird daher mit großer Wahrscheinlichkeit umsonst getan. Wenn man in der Scheiße sitzt, ist auch allgemein im Leben Nichtstun wahrscheinlich nicht die beste Strategie.

Vor- und Nachteile

Ein klarer und eindeutiger Vorteil der Technik ist eine deutliche Effektivierung der Suche. Ein mit dieser Technik arbeitendes Programm wird um ein bis zwei Halbzüge tiefer suchen als dasselbe Programm mit gewöhnlicher Alpha-Beta-Technik. Mit anderen Worten, es spielt – falls keine Nachteile vorhanden sind – um 200 bis 300 Elo-Punkte besser!

Die Technik wirkt sich umso günstiger aus, je tiefer das Programm rechnet. Im Zuge einer tiefen Berechnung entfernt sich die Stellung immer mehr von der plausiblen, ausgeglichenen Ausgangsstellung und nimmt in den meisten Varianten bizarre, eine Seite klar bevorzugende Formen an. Diese absolut unnötigen Varianten (ca. 99% der Varianten sind von dieser Art) werden mit der Null-Zug-Technik sehr effektiv verkürzt und widerlegt. Die Schnelligkeit von Fransens Fritz ergänzt sich daher sehr gut mit der verwendeten Suchtechnik.

Die Nachteile heißen Horizont-Effekt und Zugzwang. Durch die Verkürzung der Variante wird natürlich das Risiko größer, etwas zu übersehen. In taktisch kritischen Situationen ist allerdings die Möglichkeit, zwei Züge hintereinander ausführen zu können, ein derartiger Vorteil, daß dieser Effekt sehr selten auftritt.

Bei Nimzo wirkt sich diese Horizontverengung „nur“ bei Mattdrohungen aus. Wird kurz vor dem Horizont ein Null-Zug ausgeführt, so geht der Suchprozeß wegen der verkleinerten Suchtiefe direkt in die Ruhesuche über. In der Ruhesuche werden momentan aber nur die Schlagzüge des Gegners, nicht jedoch „stille“ schachgebende Züge untersucht. Die Partie Nimzo-Kallisto von der Mikro-WM '93 liefert eine anschauliche Demonstration dieser Kurzsichtigkeit. Daß das Problem nicht an der Null-Zug-Technik, sondern an der sehr sparsamen Ruhesuche liegt, wird schon durch den Gegner bewiesen. Bart Weststraate hat ebenfalls von Frans Morsch die Null-Zug-Technik übernommen.

Problem Zugzwang

Die stillschweigende und prinzipielle Annahme der Null-Move-Technik lautet: Es gibt einen Zug, der zumindest so gut wie der Null-Zug ist. Durch die Ausführung des Null-Zuges überlasse ich der Gegenseite einen Vorteil. Reicht selbst dieser Vorteil nicht zum Ausgleich, so müssen die vorhergehenden Züge der Gegenseite schlecht gewesen sein. In Zugzwang-Stellungen trifft diese fundamentale Annahme per Definition nicht mehr zu. In der Eröffnung und im Mittelspiel sind Zugzwang-Stellungen nicht sehr wahrscheinlich, speziell in Bauernendspielen wird Zugzwang aber ein dominierendes strategisches Thema. Aus diesem Grund wurden in Nimzo die folgenden Gegenstrategien eingebaut:

- Falls in der Ausgangsstellung eine Seite keine Figuren mehr besitzt, wird die Null-Zug-Heuristik für beide Seiten ausgeschaltet. In diesen einfachen Endspielen nimmt die Effizienz der Hashtabellen stark zu und der Ausfall der selektiven Suchkomponente wird dadurch etwas kompensiert.

- Falls im Inneren des Suchbaumes das Figurenmaterial der Seite mit Zugrecht kleiner gleich einem Läufer ist, wird der Null-Zug in dieser Stellung ebenfalls nicht ausgeführt.
- In komplexeren Endspielen (z.B. jeweils 2 Türme und 1 Läufer) wird als letzter Zug in der Ausgangsstellung der Null-Zug untersucht. Stellt sich heraus, daß der Null-Zug besser als der beste reguläre Zug ist, so wurde eine Zugzwang-Situation entdeckt. Es werden in diesem Fall die Hashtabellen gelöscht und die Suche nach einem kräftigen Schluck aus der Zeitflasche von vorne begonnen.

Nach seinem Verhalten zu schließen, dürfte Fritz ebenfalls nach Regel I) und II) vorgehen. Regel III) ist 'Nimzo only'. Obwohl auf den ersten Blick elegant, wird sie von einem im Computerschach typischen Phänomen geplagt. Wirklich effektiv erkennt sie nur Zugzwang-Stellungen in der Ausgangsposition. Dann ist es aber meistens schon zu spät, das Programm hat sich bereits vorher durch Blind- und Blödheit in die Nesseln gesetzt.

Daß man mit diesen Regeln bei weitem nicht alle Probleme beseitigt, zeigt ein nettes Beispiel von R. Kofler im PC-Schach 1/93.



Weiß am Zug

Weiß setzt mit der Folge **1.Td1+ Tg1 2.Tf1 Txf1 3.Kxf1 a5 4.bxa5** usw. matt. Auch nach Stunden haben Nimzo und Fritz nicht die geringste Ahnung vom Matt. Der Zug **3.Kxf1** wird mit dem Nullzug **3...Kh1-h1** elegant pariert. Schaltet man in Nimzo die Null-Zug-Heuristik ab, ist die Mattanzeige eine Frage von ein paar Sekunden.

Die Null-Zug-Mode

Die technischen Details der Technik wurden von mir im ICCA-Journal September 1993 veröffentlicht. Seither ist ein richtiges Null-Zug Coming Out unter den Programmierern ausgebrochen. Neben dem Ahnvater Fritz/Quest und dem mir damals bereits bekannten Kallisto bestätigte in der Zwischenzeit

auch Dap Hartmann meine damalige Vermutung, daß der erstaunliche Aufschwung des Veteranen Dappet unter anderem auf richtiges Nichtstun zurückzuführen ist. Einige andere Programmierer haben inzwischen die Technik in ihre Programme eingebaut und schreiben mir auch ihre Verbesserungen.

Nach dem nervenaufreibenden Spiel The King-Nimzo auf der Mikro-WM '93 meinte Johan de Koning, daß The King auch mit der Null-Zug-Technik arbeite. Leider war seine Aufregung nicht groß genug; weitere Details kamen nicht mehr über seine Lippen. Nach The Kings Verhalten zu schließen, wird die Technik etwas anders eingesetzt. Ich hätte ihm jedenfalls auch noch den zweiten halben Punkt gegeben, wenn er mir verraten hätte, wie The King das Patt in 16 gefunden hat. Hätte er ihn genommen, wäre er jetzt Weltmeister und ich ein Stückchen gescheiter.

Aus zwei in Testspielen gegen Nimzo vorkommenden Endspiel-Patzern vermute ich, daß auch Gideon in der einen oder anderen Form die Null-Zug-Technik verwendet. Nur Richard Langs Genius dürfte Null-Zug-frei sein. Trotz mehreren hundert Testspielen ist mir dieses Programm aber bis zum heutigen Tag eine Sphinx. Das einzige, was ich wirklich weiß: Es ist kein Genie – aber es ist verdammt schwierig, besser zu sein.

Die perfekte Schachstation

Computer für höchste Ansprüche ab DM 2998,-

Dipl.-Inform. Dieter Steinwender
EDV-Beratung
Entenweg 34
22549 Hamburg

Nutzen Sie unsere Erfahrung! Wir bieten:

- Komplette-Systeme
- günstige Preise
- Betreuung und Beratung

Tel. (040) 831 43 43