1994.05

Es ist entschieden: Das Mühlespiel ist unentschieden

R. Gasser und J. Nievergelt
Informatik, ETH Zürich

1. Das Mühlespiel als kombinatorisches Problem

Das ehrwürdige Mühlespiel haben wir in dieser Rubrik bereits einmal behandelt (Zur Kombinatorik von "n-in-a-row" und Blockadespielen, Informatik Spektrum, 13, No .4, 221-225, Aug. 1990). Die meisten Leser werden die Spielregeln als Kind erlernt habe, also sollte es zum Verständnis dieses Artikels genügen, eine Brettstellung abzubilden, die drei Phasen des Spielverlaufs in Erinnerung zu rufen, und vor allem einige Statistiken über den Zustandsraum des Mühlespiels festzuhalten.

Auf dem unten abgebildeten Brett mit 24 besetzbaren Punkten tummeln sich maximal 9 weiße und 9 schwarze Steine, die in der Eröffnung gesetzt werden, im Mittelspiel von Punkt zu benachbartem Punkt ziehen, und im Endspiel, wenn ein Spieler nur noch 3 Steine hat, springen dürfen. Mit dem Doppelziel, durch das Schließen einer Mühle gegnerische Steine zu schlagen, oder noch besser, sie so einzuschließen, dass der Gegner keinen Zug hat. Mühle-Experten würden es wohl kaum erwarten, aber der Computer versichert uns, dass Bild l eine gegenseitige Zugzwangstellung festhält: gegen optimales Spiel verliert Weiß am Zug in 74. Halbzügen, Schwarz in 60 Halbzügen. Ein Halbzug, oder "ply" auf englisch, ist ein Zug eines Spielers, im Gegensatz zum üblichen Zugbegriff im Schach, der genau genommen ein Zugpaar darstellt.

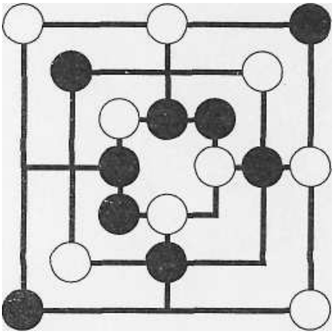


Bild 1: Gegenseitige Zugzwangstellung - wer zieht, verliert

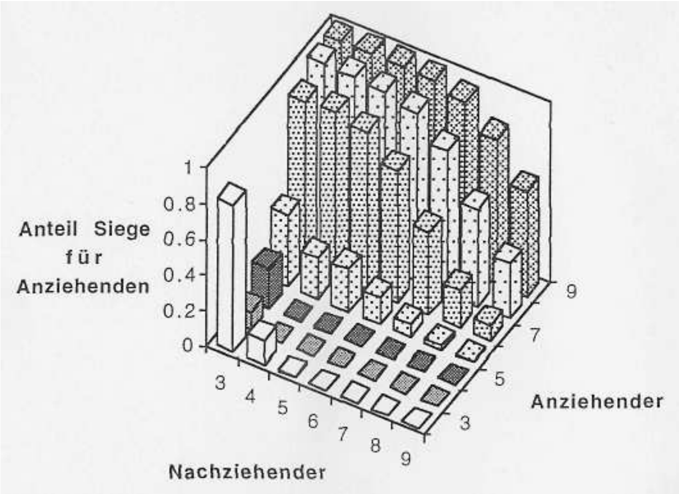
In der Rubrik Overflow, August 1990, hatten wir das 3-gegen-3 Mühle-Endspiel vollständig analysiert. Mit 24 spielbaren Punkten auf dem Mühlebrett gibt es $\binom{24}{3}\binom{21}{3}$ = 2'691'920 Konfigurationen oder Zustände, welche durch Berücksichtigung aller Symmetrien auf nur 56'922 wesentlich verschiedene Zustände reduziert werden können. Das ist für Computer eine so kleine Zahl, dass das erschöpfende Durchsuchen dieses Zustandsraums auf einem Macintosh in ca. 12 Minuten abläuft.

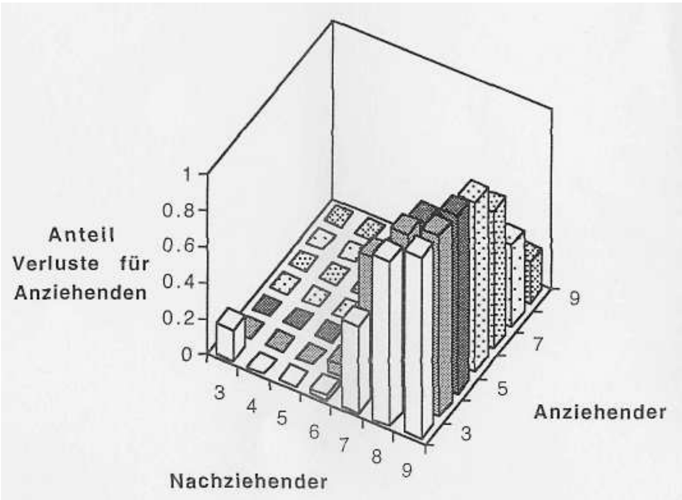
In ganz anderen Größenordnungen liegt das Problem beim vollen Mühlespiel. Nach Berücksichtigung aller Brettsymmetrien (Spiegelungen an 4 Achsen und am mittleren Ring, da der innere und der äußere Ring gleichwertig sind) ergeben sich interessante Statistiken, von denen wir einige zusammenfassen.

2. Ergebnisse

Was Mühlespieler schon lange vermutet haben, nämlich, dass eine beiderseitig korrekt gespielte Partie unentschieden endet, ist durch eine Koalition mehrerer Computer bestätigt worden - durch Berechnungen, die sich mit Unterbrechungen über drei Jahre erstreckten.

Als Nebenprodukt dieses Hauptresultat ergibt die Rechnung natürlich auch eine Fülle statistischer Ergebnisse über das Mühlespiel, deren Auswertung und Interpretation (Bedeutung für Mühlespieler: wie erkennt man Gewinn- oder Verlust-trächtige Stellungen?) keine leichte Aufgabe ist. Als Beispiel untersuchen wir die Bedeutung des Materials, gemessen über alle möglichen Stellungen mit i gegen j Steine, $3 \le i, j \le 9$, in Abhängigkeit des Anzugs, d.h. des Rechts, den nächsten Zug auszuführen. Die folgenden Bilder 2 und 3 illustrieren die Verteilung der gewonnenen und verlorenen Stellungen.





Bilder 2 & 3: Die Bedeutung des Materials und des Anzugs

Bilder 2 und 3 belegen die Aussage: "Ceteris paribus" (vorausgesetzt, alles andere sei gleich), wie Schachmeister Siegbert Tarrasch (1862-1934) immer lehrte, sind sowohl Materialvorteil wie auch das Anzugsrecht meistens vorteilhaft. Der Vorteil des Anzugs ist auf den ersten Blick ersichtlich am größeren Volumen des Gebirges im Bild 2 verglichen mit dem in Bild 3. Auch ganz extrem am Spezialfall des 3-3 Endspiels, in dem der Anziehende über 80% der Stellungen gewinnt, der Nachziehende knapp 20%.

Der Vorteil des Materialvorsprungs ist ersichtlich in Bild 2 an der Höhe > 50% des Gebirges hinter der Diagonale, die von der 3-3 Ecke links zur 9-9 Ecke rechts führt. Je größer der Materialvorteil, desto größer i.A. die Zahl der gewonnenen Stellungen, wie die monoton fallende Flanke des Berges mit Gipfel in der hinteren 9-3 Ecke zeigt.

Das Komplement der Summe der beiden Gebirge in Bilder 2 und 3 misst den Anteil der unentschiedenen Stellungen. Interessant, wie die verschiedenen i-j Teilspiele von "extrem friedlich" (fast alle 4-4, 5-4, 5-5 Stellungen sind unentschieden) bis extrem kampfbetont variieren (fast keine 3-3, und nur wenige Stellungen mit einer Differenz von 4 oder mehr Steinen sind unentschieden). Überraschend zeigen die Bilder, dass, mit Ausnahme des Spezialfalls 3-3, Siege fast nur auf dichtbevölkertem Brett zu finden sind, wenn einer der Spieler 7 oder mehr Steine hat.

Zur Interpretation obiger Darstellung ist zu beachten, dass alle vertikalen Balken auf l normiert sind - die Grosse (Anzahl Stellungen) des entsprechenden i-j Teilraumes ist im Bild nicht ersichtlich. Da jeder Balken nur eine Aussage über seinen Teilraum macht, stellt das Volumen der abgebildeten Gebirge also nicht das natürliche Mass für die Anteile Gewinn-Unentschieden-Verlust im gesamten Mühlespiel dar. Letztere sind aber ähnlich gelagert. Im Mittel über alle Stellungen des Mühlespiels in den Phasen des Ziehen-und-Springens sind 47.6% für den Anziehenden gewonnen, 21.6% unentschieden, 30.8% verloren.

Es ist allerdings zu bemerken, dass die abgebildeten Prozentzahlen sich nicht direkt in Siegeschancen für praktische Spieler übersetzen lassen - sogar dann nicht, wenn diese Spieler perfekt wären. Die abgebildeten Statistiken beziehen sich auf den Raum aller möglichen Stellungen, nicht auf den Raum der Stellungen, die im Spiel tatsächlich vorkommen. Diese 2 Räume besitzen verschiedene statistische Eigenschaften - im Schach, z.B., kann die Anzahl der "realistischen" Stellungen abgeschätzt werden *[Nievergelt 91]* und ist nur ein verschwindender Bruchteil aller möglichen Stellungen. Es ist zweitens zu beachten, dass die abgebildeten Statistiken sich auf die Phasen des Ziehen-und-Springens beziehen, nicht auf die Eröffnungsphase des Setzens. In der Eröffnung kommen andere Kriterien zur Geltung: der Nachziehende in der Ausgangsstellung, beim leeren Brett, hat den Vorteil, den letzten Stein auf ein beliebiges Feld setzen zu dürfen in Kenntnis der Lage aller gegnerischen Steine (siehe Abschnitt 3).

3. Die Methode: Zangenangriff vom Spieianfang und vom Spielende her

Wie sucht man einen Raum ab, der nicht nur aus 1010 Zuständen besteht, sondern in dem auch ein reichhaltiges Netz von Beziehungen zwischen diesen Zuständen aufgebaut werden muss? Die Informatik kennt zwei Hauptarten von Suchalgorithmen, vorwärts (ausgehend von der aktuellen Stellung hin zu einem gewünschten Ziel) und rückwärts (ausgehend von den bekannten Endstellungen zurück zur aktuellen Stellung). Beide wurden zur Lösung des Mühlespiels eingesetzt: Vorwärts-Suche für die 18-ply-Setzphase, Rückwärts-Suche über 28 Teilräume der Form "i gegen j Steine" für $3 \leq i, j \leq 9$.

Ausgehend vom leeren Brett zielt die Vorwärts-Suche auf günstige bekannte Werte hin. Da die einzigen a priori bekannten Werte, die gewonnenen und verlorenen Endstellungen, i.A. zu weit in der Zukunft liegen, wird zuerst die Rückwärts-Suche gerechnet, welche der darauf folgenden Vorwärts-Suche einen Horizont von gesicherten Werten als Ziel anzubieten. Bild 4 deutet an, dass die 3 Teilräume 9-9, 9-8, 8-8 genügen.

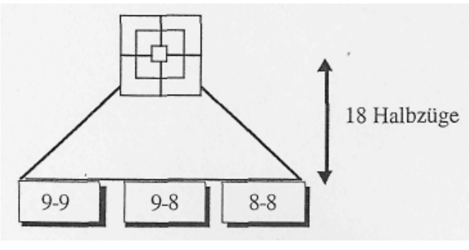


Bild 4: Drei Datenbanken genügen als Zielhorizont für die Vorwärtssuche der Setzphase

Bei der Rückwärts-Suche fließt Information von den durch die Spielregeln definierten Werte (Sieg oder Niederlage) vom 3-gegen-3 Teilraum hin zum 9-gegen-9 Raum gemäss Bild 5.

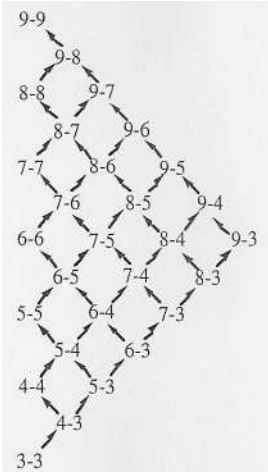


Bild 5: Information fließt vom Bekannten zurück zum noch Unbekannten

Um diese einleuchtende Idee des Zangenangriff im Detail zu begründen (warum von beiden Enden her suchen, warum sollen sich die beiden Suchvorgänge gerade an der Grenze zwischen der Setzphase und der Ziehphase treffen?), und vor allem, um sie dann durchzuführen, müssen Dutzende von schwierigen technischen Fragen beantwortet werden. Wir wollen aus dieser Vielfalt nur einige wenige erwähnen.

• Warum genügt es, in der Setzphase nur auf die 3 Teilräumen 9-9, 9-8, 8-8 hinzuzielen? Während des Setzens können viele Mühlen geschlossen werden, also kann die Eröffnung direkt in jeden Teilraum "i-gegen-j" führen. Antwort: Der Beweis, dass das Mühlespiel unentschieden ist, besteht aus zwei unabhängigen Teile. Erstens, die Existenz einer Spielstrategie für Weiß, die ihm ein unentschieden sichert, und zweitens, die analoge Aussage für Schwarz. Beide Spielstrategien mit der Garantie "ergibt mindestens unentschieden" werden bereits in den 3 erwähnten Teilräumen gefunden, also kann die Setzphase sich auf diesen Horizont beschränken.

• Dieses Vorwärts-Suchen ist dank des bekannten alpha-beta Suchverfahrens, das viele Stellungen als "minderwertig" entlarvt und eliminiert, sehr effizient. Beispiel: von den etwa 3.5 Millionen möglichen Stellungen nach 8 Halbzügen wurden beim Suchen einer Strategie, die mindestens unentschieden garantiert, nur ein sehr kleiner Brachteil erzeugt und betrachtet. Genau 15'513 Stellungen beim Beweis, dass Weiß mindestens unentschieden halten kann, und 4'393 beim Beweis, dass Schwarz mindestens unentschieden halten kann. Weiß, der Anziehende in der Ausgangsstellung, setzt den ersten Stein aufs leere Brett. Das erweist sich als ein Nachteil, denn der Vorteil liegt bei Schwarz, der den letzten Stein setzen darf in Kenntnis der Lage aller gegnerischen Steine. Es ist also schwieriger (zeitaufwendiger) zu beweisen, dass in der Ausgangsstellung Weiß mindestens unentschieden hat, als dass Schwarz mindestens unentschieden hat - der alpha-beta Algorithmus muss für Weiß fast viermal mehr Stellungen untersuchen!

• Die Darstellung des Zustandsraums mit circa 1010 Stellungen muss möglichst Speicher-effizient geschehen. Eine geeignet gewählte Funktion bildet jede Stellung S l-1-deutig auf eine ganze Zahl ("Gödel-Nummer") i(S) ab, die als Index in einem Array A verwendet wird, der die Spielwerte W(S) speichert, l Byte pro Stellung. Einerseits soll die Indexfunktion einfach und schnell berechenbar sein, anderseits sollen die erzeugten Indexwerte den Adressraum (Indexbereich) von A möglichst kompakt füllen - zwei widersprüchliche Anforderungen.

• Unter den 28 Teilräumen ist 3-3 der kleinste, 8-7 Mitetwas über l09 Stellungen der größte. Um möglichst große Teile im Zentralspeicher zu halten werden verschiedene Datenkompressionsverfahren eingesetzt, was wiederum im Widerspruch zu schnellem Zugriff steht.

• Die Organisation der Rechnung in Abhängigkeit der Grosse des aktuellen i-j Teilraumes, des Zentral-und-Plattenspeichers, ist ein schwieriges Programmoptimierungsproblem.

4. Warum?

Mühle ist nur das jüngste in einer Reihe von Spielen, die im vergangenen Jahrzehnt durch erschöpfendes Suchen gelöst wurden: Qubic *[Patashnik 80]*, Connect-4 *[Allen 89, Allis 89]*, Go-Moku *[Allis 94]*, und viele Schachendspiele mit wenigen Figuren (siehe die Übersicht *[Herik 86]* und viele neuere Ergebnisse wie z.B. *[Thompson91, 92]*, *[Stiller 91]*). Weitere werden folgen, wir haben wohl erst den Anfang einer ganzen Reihe "gelöster" Gesellschaftsspiele gesehen.

Warum lässt man Computer Tage- oder Jahrelang rechnen, um kombinatorische Spiele durch erschöpfendes Suchen zu lösen? Stiehlt eine exakte Lösung nicht den Reiz des Spieles, sodass es für Spieler uninteressant wird? Diese in der Frühzeit der Computerspiele oft geäußerte Befürchtung trifft kaum zu. Sogar Anfänger können ja Freude finden an Tätigkeiten, die andere besser, und irgendeine Maschine perfekt ausübt. Also: negative Folgen müssen wir nicht befürchten, aber gibt es irgend etwas Positives an der Computerspielerei?

'Brute-force' Lösungen sind mathematisch nicht 'elegant', sind ohne Computer gar nicht nachvollziehbar, und liefern nach der Erzeugung von Gigabytes von Daten kryptische Urteile wie "das Mühlespiel ist unentschieden". Lösungsmethode und Ergebnis sind gar nicht 'benutzerfreundlich' präsentiert, und das Wissen um die Existenz optimaler Spielstrategien macht Sie noch nicht zu einem guten Mühlespieler. Die Interpretation einer Spieldatenbank in eine für Menschen genießbare Form verlangt tiefe Fachkenntnisse, ein feines Fingerspitzengefühl für didaktische Darstellung eines komplexen Inhalts, und sehr viel Arbeit. Um die von Ken Thompson berechnete Datenbank für das oft auftretende Schachendspiel KTBKT, d.h. König, Turm und Bauer gegen König und Turm, für Schachspieler zu interpretieren brauchte Großmeister John Nunn ein Buch von 320 Seiten *[Nunn 92]*.

Also nochmals: Warum lässt man Computer Tage- oder Jahrelang rechnen, um Spiele auf eine Art zu "lösen", die dem Spieler selten etwas bringt? Die erste Antwort ist dieselbe wie auf die Frage "Warum erklettert man die Eiger-Nordwand?" Um zu zeigen, dass man es kann. Aber es gibt noch eine zweite Antwort: kombinatorisches Suchen in großen Räumen ist eine sehr allgemeine Lösungsmethode von zunehmender Wichtigkeit. Jahrzehntelang haben sich Informatiker auf die effizientesten Algorithmen gestürzt, die sie finden konnten, immer mit dem Ziel, den Rechenaufwand möglichst klein zu halten. Algorithmen mit Rechenzeiten, die höchstens polynominal in der Grosse der Eingabedaten wachsen, monopolisierten die Aufmerksamkeit der theoretischen Informatiker.

Aber solche effiziente Algorithmen kann man nur bei ausgewählten, relativ einfachen Problemen finden. Die Welt der Anwendungen ist aber nicht einfach, und die meisten Probleme von praktischer Bedeutung, sofern sie überhaupt für Computer zugänglich sind, verlangen rechenintensive Methoden. Der Informatiker weiß zwar so ziemlich alles über Sortieren und binäres Suchen, das es zu wissen gibt, aber in Bezug auf erschöpfendes Suchen in sehr großen Zustandsräumen sind wir immer noch im Stadium des Übens, des Schärfens unserer Werkzeuge. Und die wohldefinierten "Mikroweiten" der Spiele haben gerade die richtige Komplexität, um als Maßstäbe unserer Fähigkeit des Suchens in endlichen, aber nach heutiger Ansicht sehr großen, Räume zu dienen.

Noch vor einem Jahrzehnt wären solche Zeit- und vor allem speicherfressende Rechnungen nur auf den allergrößten Rechnern möglich gewesen, falls überhaupt. Der rasante Fortschritt der Hardware wird uns noch weitere Größenordnungen an roher Rechenkapazität bescheren. Mit verteilten Rechnungen auf Netzen von Tausenden von unterbenutzten Rechnern steht heute, oft kostenlos, eine rohe Rechenkapazität zur Verfügung, welche diejenige der Supercomputer bei weitem übersteigt. Es ist durchaus sinnvoll, durch spielerische Experimente herauszufinden, was damit alles berechnet werden kann.

5. Wie glaubwürdig ist eine jahrelange Rechnung?

Bei 10 Gigabytes als Ergebnis einer jahrelangen Rechnung drängt sich die Frage besonders lautstark auf, die man bei jeder Rechnung stellen sollte: wie glaubwürdig ist das Resultat? Natürlich haben sich im Laufe dieser Marathon-Rechnung etliche Fehler eingeschlichen: Programmierfehler in der selbstgeschriebenen Software; der verwendete Compiler ist nicht über alle Zweifel erhaben, besonders bei der Allokation riesiger Datenmengen; durch Hardware verursachte Bit-Fehler auf dem Plattenspeicher traten auch auf.

Etliche Verifikationsrechnungen mit Hilfe neuer Programme überprüften die Konsistenz der erzeugten Mühle-Datenbank und der Setzphase. Die ersten Verifikationen hatten Fehler aufgedeckt, nach deren Verbesserung lief eine Verifikationsrechnung auf einem Netzwerk von Sun-Workstations während ca. 3 Monaten fehlerfrei zu Ende. Weitere noch laufende Verifikationen haben keine Inkonsistenten aufgedeckt. Wie man so schön sagt: "Wir sind uns keiner unentdeckter Fehler bewusst".

Es ist vielleicht nie ganz auszuschließen, dass sich irgendwo in einer Datenmenge von 10 Gigabytes ein falsches Bit verbirgt. Das Problem der "Verifikation des Verifikationsalgorithmus" war schon den alten Römern bekannt: "wer bewacht die Wächter?" Unser Vertrauen in das angekündigte Ergebnis basiert auf der Beobachtung, dass das "Unentschieden" sehr breit abgestützt ist - es gibt für beide Spieler viele Möglichkeiten, das Unentschieden zu erreichen. Hinweise dafür sind:

- Der Wert der Ausgangsstellung war sowohl vor wie nach der Fehlerbehebung unentschieden.

- Nur wenige Eröffnungspositionen konnten nicht als mindestens Unentschieden bewiesen werden

- Vorsichtige Mühlespieler lernen bald, das Unentschieden zu halten.

Aber natürlich hoffen wir auf eine unabhängigen Verifikation der Mühle-Datenbank und deren kompakten Zusammenfassung "unentschieden". In den Natur- und technischen Wissenschaften sind wiederholbare Experimente der Prüfstein, der Behauptungen von Resultaten unterscheidet. Auch die Informatik kommt nicht darum herum, experimentelle Ergebnisse durch unabhängige Experimente nachzuvollziehen. Unabhängiges Überprüfen wird in dem Masse wichtiger, wie sich zunehmend viele Resultate auf zeitraubende Rechnungen stützen.

Hat das Mühlespiel durch seine "Lösung" an Interesse verloren? Wohl kaum, nicht einmal für Gasser's Mühleprogramm Bushy. Dieses hatte 1990 an der zweiten Computer Olympiade in London, rein mit Heuristiken ausgerüstet, den bauschen Nine Men’s Morris Meister, Michael Sunley, in einem Wettkampf mit 4 Siegen und 2 Unentschieden geschlagen *[Levy 91]*. Gestärkt mit Datenbanken des gesamten Mühlespiels, und darauf erpicht, unter den optimalen Zügen die trickreichsten zu wählen, wartet Bushy jetzt auf neue Herausforderungen zum Wettkampf!

Bibliographie

[Allen89]	Allen, J.D., "A Note on the Computer Solution of Connect-Four", 134-135 in [Levy 89].
[Allis 89]	Allis, L.V., van der Meulen, M. and van den Herik, HJ., "A Knowledge-Based Approach to Connect-Four: The Game is Over: White To Move Wins", 113-133 in [Levy 89].
[Allis 94]	Allis, L.V, "Searching for Solutions in Games and Artificial Intelligence", Doctoral Thesis, University of Limburg, Maastricht, The Netherlands, to appear.
[Henk 86]	van den Herik, H.J. and Herschberg I.S.,"A DataBase on Data Bases", ICCA Journal 9(1), p. 29 (March 1986).
[Lew 89]	Levy, D. and-Beal, D. (Eds.), "Heuristic Programming in Artificial Intelligence 1: The First Computer Olympiad", Ellis Horwood, London, 1989.
[Levy 91]	Levy, D., "Nine Men's Morris", 55-57 in D.N.L. Levy, D.F. Beal (eds): Heuristic Programming in Artificial Intelligence 2: The 2nd Computer Olympiad, Ellis Horwood, Chichester, 1991.
[Nievergelt 91]	J. Nievergelt, "Information content of chess positions: Implications for game-specific knowledge of chess players", 283-289 in Machine Intelligence 12, (eds. J. E. Hayes, D. Michie, E. Tyugu) , Clarendon Press, Oxford, 1991.
[Nunn 92]	Nunn, John, "Secrets of Rook Endings", Batsford Ltd. UK, 1992.
[Patashnik 80]	Patashnik, O., "Qubic: 4x4x4 Tic-Tac-Toe", Mathematics Magazine Vol.53 No.4, pp.202-216, 1980.
[Stiller 91]	Stiller, L., "Group Graphs and Computational Symmetry on Massively Parallel Architecture", The Journal of Supercomputing, 5, 99-117 (1991)
[Thompson 91]	Thompson, K., "Chess Endgames Vol. l", ICCA Journal 14(1), p. 22, 1991.
[Thompson 92]	Thompson, K., "Chess Endgames Vol. 2", ICCA Journal 15(3), p. 149, 1992.