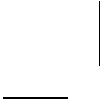


Memory versus Search in Games



Memory versus Search in Games

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit Maastricht,
op gezag van de Rector Magnificus,
Prof. dr. A.C. Nieuwenhuijzen Kruseman,
volgens het besluit van het College van Decanen,
in het openbaar te verdedigen
op vrijdag 16 oktober 1998 om 16.00 uur

door

Dennis Michel Breuker

Promotor: Prof. dr. H.J. van den Herik

Leden van de beoordelingscommissie:

Prof. dr. P.T.W. Hudson (voorzitter)

Prof. dr. A. de Bruin (Erasmus Universiteit Rotterdam)

Prof. dr. J. Schaeffer (University of Alberta, Edmonton)

Prof. dr. S.H. Tijs

Prof. dr. H. Visser



Dissertation Series No. 98-4

ISBN 90-9012006-8

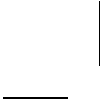
NUGI 855

Subject headings: artificial intelligence / games / search

©1998 Dennis Breuker

Cover design and photography: Hans Hoornstra

*Voor mijn moeder en
ter nagedachtenis aan mijn vader*



Contents

List of Tables	xi
List of Figures	xiii
Preface	xv
1 Introduction	1
1.1 Games	1
1.2 Knowledge versus search	3
1.3 Memory versus search	4
1.4 Problem statements	5
1.5 Outline of the thesis	6
2 The transposition table	9
2.1 Notions and concepts	10
2.2 Transpositions	13
2.3 A transposition table	14
2.3.1 Hashing	14
2.3.2 The traditional components	16
2.4 Implementing a transposition table	19
2.4.1 Data structures	19
2.4.2 Probability of errors	20
2.5 Experimental set-up	22
2.5.1 The game of chess	22
2.5.2 The game of domineering	26
2.6 The test domains	27
2.6.1 Chess test sets in the literature	27
2.6.2 Our chess test set	29
2.6.3 The domineering test set	30
2.7 Experiments and results	30
2.7.1 Comparing replacement schemes	31
2.7.2 Quantifying the merits of move and score	42
2.7.3 Using additional memory	44
2.8 Chapter conclusions	47

3	The proof-number search algorithm	51
3.1	An informal description	51
3.2	The pseudo-code of the algorithm	54
3.3	Experimental set-up	55
3.3.1	The search engine	57
3.3.2	The move ordering	58
3.4	The test set	58
3.5	Experiments	59
3.6	Results	59
3.6.1	Strengths of pn search	61
3.6.2	Weaknesses of pn search	64
3.7	Chapter conclusions	66
4	The pn^2-search algorithm	69
4.1	Pn search with small memory: pn^2 search	69
4.2	The size of the second-level pn search	70
4.3	Experiments	74
4.4	Results	75
4.5	Chapter conclusions	78
5	The graph-history-interaction problem	81
5.1	The history of a position	81
5.2	An example of the GHI problem	83
5.3	A review of previous work	85
5.4	BTA: an enhanced DCG algorithm	88
5.4.1	Phase 1: select the best node	90
5.4.2	Phase 2: evaluate the best node	93
5.4.3	Phase 3: back up the new information	95
5.5	The pseudo-code of the BTA algorithm	95
5.5.1	Phase 1: select the most-proving node	95
5.5.2	Phase 2: evaluate the most-proving node	99
5.5.3	Phase 3: back up the new information	100
5.6	Experimental set-up	102
5.7	Results	103
5.8	Chapter conclusions	104
6	Evaluations and conclusions	107
6.1	More memory and less search	107
6.2	Less memory and more search	108
6.3	Less memory and less search	109
6.4	Future research	109
6.4.1	More memory and less search	110
6.4.2	Less memory and more search	110
6.4.3	Proof-number search	111

Appendices

A	The chess middle-game test set	113
B	The chess endgame test set	117
C	The transposition-table results	121
D	The pn-search and pn²-search test set	129
E	The pn-search versus $\alpha\beta$-search results	137
F	The pn²-search results	139
G	The BTA results for pn search	143
	References	147
	Index	159
	Summary	163
	Samenvatting	167
	Curriculum Vitae	171

List of Tables

2.1	Game-theoretic results of domineering.	41
3.1	Comparing pn search and $\alpha\beta$ search.	61
5.1	Comparing four pn-search variants.	103
C.1	Results for 3-ply middle-game searches without time stamping. . . .	122
C.2	Results for 3-ply middle-game searches with time stamping.	122
C.3	Results for 4-ply middle-game searches without time stamping. . . .	122
C.4	Results for 4-ply middle-game searches with time stamping.	123
C.5	Results for 5-ply middle-game searches without time stamping. . . .	123
C.6	Results for 5-ply middle-game searches with time stamping.	123
C.7	Results for 6-ply middle-game searches without time stamping. . . .	124
C.8	Results for 6-ply middle-game searches with time stamping.	124
C.9	Results for 7-ply middle-game searches without time stamping. . . .	124
C.10	Results for 7-ply middle-game searches with time stamping.	125
C.11	Results for 8-ply middle-game searches without time stamping. . . .	125
C.12	Results for 8-ply middle-game searches with time stamping.	125
C.13	Results for 10-ply endgame searches with time stamping.	126
C.14	Replacement-scheme results for domineering.	127
C.15	Comparing move and score in the middle game.	127
C.16	Comparing move and score in the endgame.	128
C.17	PV results in the middle game.	128
C.18	PV results in the endgame.	128
E.1	Comparing pn search and $\alpha\beta$ search.	137
F.1	Two extremes of the fraction function.	139
F.2	The pn ² results for varying parameters a and b	140
G.1	The results for four pn-search variants.	143

List of Figures

2.1	A depth-first traversal of a search tree.	12
2.2	A breadth-first traversal of the search tree of Figure 2.1.	12
2.3	A best-first traversal of the search tree of Figure 2.1.	13
2.4	A BTM position that can be reached by distinct move orders.	14
2.5	The hash value.	16
2.6	A WTM position with blocked Pawns.	18
2.7	The $\alpha\beta$ -search function with a transposition table.	25
2.8	Schemes in the middle game (without time stamping).	35
2.9	Schemes in the middle game (with time stamping).	37
2.10	Using a transposition table in the middle game.	38
2.11	Comparing replacement schemes in the endgame.	39
2.12	Comparing replacement schemes in domineering.	40
2.13	Comparing move and score in the middle game.	43
2.14	Comparing move and score in the endgame.	45
2.15	Storing an n -ply PV in the middle game.	48
2.16	Storing an n -ply PV in the endgame.	49
3.1	An AND/OR tree with proof and disproof numbers.	53
3.2	The pn-search algorithm for trees.	55
3.3	The proof-and-disproof-numbers-calculation algorithm.	56
3.4	The most-proving-node-selection algorithm.	57
3.5	The node-expansion algorithm.	57
3.6	The ancestor-updating algorithm.	58
3.7	Mate in 38 (WTM); (L. Ugren).	62
3.8	Mate in 25 (WTM); (J.-L. Seret).	63
3.9	Problem 14 of <i>Win at Chess</i> (WTM).	65
3.10	Problem 150 of <i>The Enjoyment of Chess Problems</i> (WTM).	66
3.11	Problem 213 of <i>Win at Chess</i> (WTM).	67
3.12	Six-fold transposition in problem 213 of <i>Win at Chess</i> (WTM).	67
4.1	The fraction function $f(x)$	73
4.2	The theoretical size of the second-level search.	74
4.3	The practical size of the second-level search.	75

4.4	The pn^2 results with fixed parameter a .	76
4.5	The pn^2 results with fixed parameter b .	77
4.6	The pn^2 results with fixed ratio $\frac{b}{a}$.	78
5.1	A pawn endgame (WTM).	83
5.2	The GHI problem in the pawn endgame.	84
5.3	A search tree with repetitions.	85
5.4	The DCG corresponding with the tree of Figure 5.3.	86
5.5	Our DCG corresponding with the DCG of Figure 5.4.	90
5.6	Encountering the first repetition c .	91
5.7	Encountering the second repetition c .	92
5.8	Encountering the repetition e .	93
5.9	Marking node F as a <i>possible-draw</i> .	94
5.10	Marking node C as a <i>possible-draw</i> .	94
5.11	The BTA pn -search algorithm for DCGs.	96
5.12	The function <code>SelectMostProvingNode</code> .	97
5.13	The function <code>SelectBestChild</code> .	98
5.14	The procedure <code>ExpandNode</code> .	99
5.15	The procedure <code>UpdateAncestors</code> .	100
5.16	The procedure <code>UpdateOrNode</code> .	101
5.17	The procedure <code>UpdateAndNode</code> .	102
5.18	Mate in 14 (WTM); (J. Kriheli).	105