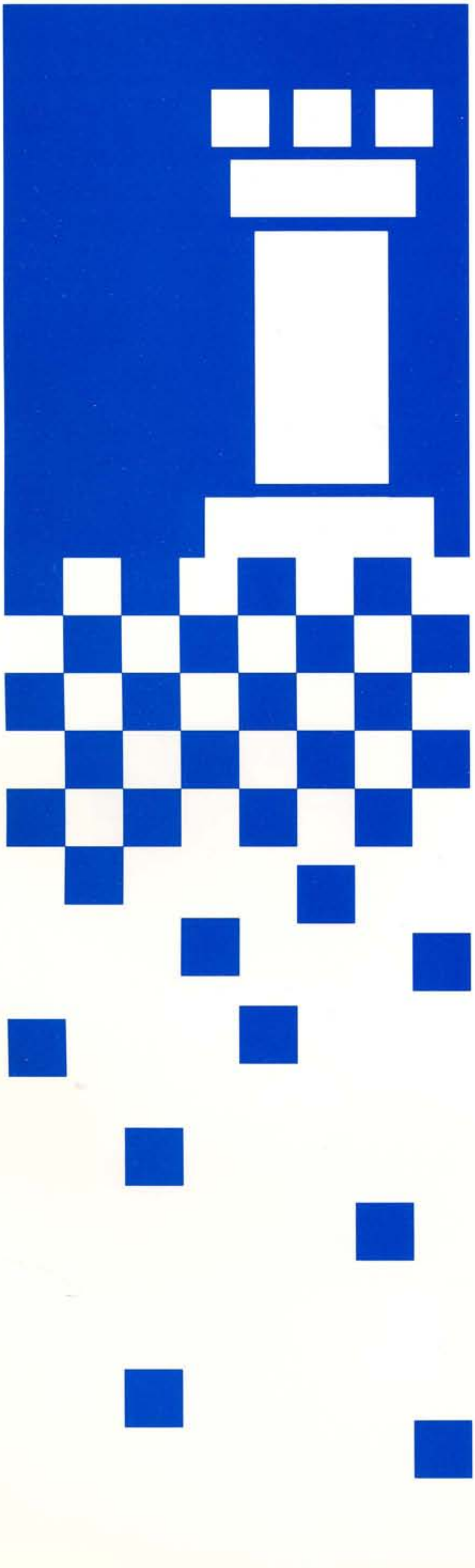

WORLD COMPUTER CHESS CHAMPIONSHIP

AGT

Edmonton, Canada
May 28, 1989

Kings move

Welcome to the 1989 AGT World Computer Chess Championship.

to Edmonton

AGT is proud to sponsor the World Computer Chess Championship, and provide the world-class data transfer technology required to stage the event. Each of the competing computers – located here and in various countries around the world – is linked to the competition site by AGT networks. Watch as artificial intelligences – from small PCs to huge supercomputers – match wits for the title of World Champion!

Join us as we welcome the world's most prominent electronic kings and queens to Edmonton!

WORLD
COMPUTER
CHESS
CHAMPIONSHIP

AGT

Important Times

Competition Schedule (Takes place in the Convention Centre, Salons 11 & 12)

Round 1	Sunday, May 28th	1 pm
Round 2	Sunday, May 28th	7 pm
Round 3	Monday, May 29th	7 pm
Round 4	Tuesday, May 30th	7 pm
Round 5	Wednesday, May 31th	7 pm

Awards Presentation

Luncheon	Thursday, June 1st	12 noon

Meetings

ICCA Triennial Meeting of the ICCA	Tuesday, May 30th	4:30 pm
ICCA Board Meeting	Wednesday, May 31th	11 am

Workshop: New Directions in Game-Tree Searching

Session 1	Monday, May 29th	1:30 pm
Session 2	Tuesday, May 30th	1:30 pm

Invited Guest Presentations

Donald Michie	Wednesday, May 31	1:30 pm
John McCarthy	Wednesday, May 31	2:15 pm

With Conference grant support from the Natural Sciences and Engineering Research Council of Canada

With secretarial assistance from Sandra Fliegel and Karen Kwiatkowski provided by the University of Alberta

Hosted by the Edmonton branch of the Canadian Information Processing Society (CIPS)

GAME COMMENTATORS

We are pleased to have the following people offering their expert commentary:

David Levy	President of the International Computer Chess Association and an International Chess Master.
David Slate	Co-author of Chess 4.6, winner of the 2nd World Computer Chess Championship, Toronto, 1977.
Kevin Spraggett	Canadian Champion and an International Chess Grandmaster. In 1988, he made the final 8 in the hunt for the Human World Chess Championship.
Mike Valvo	International Chess Master, tournament director for the 4th and 5th World Computer Chess Championships.

HONOURED GUESTS

Each World Championship, two people who have distinguished themselves for their pioneering work in computer chess are honoured. This year, we are pleased to acknowledge the contributions of:

John McCarthy	A pioneer in artificial intelligence research. He was involved in the first international computer chess match USA vs USSR in 1966.
Donald Michie	Chief Scientist for the Turing Institute, noted expert in AI technique applications and proponent of computer chess research for more than two decades.

INVITED GUESTS

A number of well-known names in the computer/chess world are part of this year's championship. They include:

Georgii Adelson-Velsky	
Vladimir Arlazarov	
Mikhail Donskoy	Authors of the program Kaissa, winner of the 1st World Computer Chess Championship, Stockholm, 1974.
Hans Berliner	Well known for his innovative work in computer chess, he is also a fromer World Correspondance Chess Champion.
Monty Newborn	He has organized computer chess events since 1970 and has been a frequent competitor with his Ostrich program.
Claude Shannon	Well-known for his work in information theory, he also wrote a pioneering paper on computer chess in 1948.
Ken Thompson	Inventor of the UNIX operating system and co-builder of the chess machine Belle, winner of the 3rd World Computer Chess Championship, Linz, 1980.

A SHORT HISTORY OF COMPUTER CHESS†

T.A. Marsland

Computing Science Department
University of Alberta
EDMONTON
Canada T6G 2H1

Review

Of the early chess-playing machines the best known was exhibited by Baron von Kempelen of Vienna in 1769. As might be expected, these were conjurer's tricks and grand hoaxes [1, 2]. In contrast, about 1890 a Spanish engineer, Torres y Quevedo, designed a true mechanical player for king-and-rook against king endgames. A later version of that machine was displayed at the Paris Exhibition of 1914 and now resides in a museum at Madrid's Polytechnic University [2]. Despite the success of this electro-mechanical device, further advances on chess automata did not come until the 1940's. During that decade there was a sudden spurt of activity as several leading engineers and mathematicians, intrigued by the power of computers and fascinated by chess, began to express their ideas on computer chess. Some, like Tihamer Nemes of Budapest [3] and Konrad Zuse [4], tried a hardware approach but their computer chess works did not find wide acceptance. Others, like noted computer scientist Alan Turing, found success with a more philosophical tone, stressing the importance of the stored program concept [5]. Today, best recognized are the 1965 translation of Adriaan de Groot's 1946 doctoral dissertation [6] and the much referenced paper on algorithms for playing chess by Claude Shannon [7]. Shannon's inspirational work was read and reread by computer chess enthusiasts, and provided a basis for most early chess programs. Despite the passage of time, that paper is still worthy of study.

Landmarks in Chess Program Development

The first computer model in the 1950's was a hand simulation [5]; programs for subsets of chess followed [8] and the first full working program was reported in 1958 [9]. By the mid 1960's there was an international computer-computer match [10] between a program backed by John McCarthy of Stanford (developed by a group of students from MIT) and one from the Institute for Theoretical and Experimental Physics (ITEP) in Moscow [11]. The ITEP group's program (under the guidance of Georg Adelson-Velskii) won the match, and the scientists involved went on to develop *Kaissa**, which became the first world computer chess champion in 1974 [12]. Meanwhile there emerged from MIT another program, *Mac Hack Six* [14], which boosted interest in Artificial Intelligence. First, *Mac Hack* was demonstrably superior not only to all previous chess programs, but also to most casual chess players. Secondly, it contained more sophisticated move ordering and position evaluation methods. Finally, the program incorporated a memory table to keep track of the values of chess positions that were seen more than once. In the late 1960's, spurred by the early promise of *Mac Hack*, several people began developing chess programs and writing proposals. Most substantial of the proposals was the twenty-nine point plan by Jack Good [15]. By and large experimenters did not make effective use of these works, at least nobody claimed a program based on those designs, partly because it was not clear how some of the ideas could be addressed and partly because some points were too naive. Even so, by 1970 there was enough progress that Monroe Newborn was able to convert a suggestion for a public demonstration of chess playing computers into a

† This article is a condensed and revised extract from the chapter "Computer Chess Methods", *Encyclopedia of Artificial Intelligence*, S. Shapiro (editor), Wiley 1987.

* The names of programs mentioned here will be written in italics. Descriptions of these programs can be found in various books [12, 13].

competition that attracted eight participants [16]. Due mainly to Newborn's careful planning and organization this event continues today under the title "The ACM North American Computer Chess Championship," and is sponsored by the Association for Computing Machinery (ACM).

In a similar vein, under the auspices of the International Computer Chess Association, a worldwide computer chess competition has evolved. Initial sponsors were the International Federation for Information Processing (IFIP) triennial conference in Stockholm (1974) and Toronto (1977), and later independent backers such as the Linz (Austria) Chamber of Commerce (1980), ACM New York (1983), the city of Cologne (1986), West Germany, and for 1989 AGT/CIPS, Edmonton. In the first world championship for computers *Kaissa* won all its games, including a defeat of *Chaos* after it had won against the favourite. An exhibition match against the the second place finisher the 1973 North American Champion, *Chess 4.0*, was drawn [10]. *Kaissa* was at its peak, backed by a team of outstanding experts on tree searching methods. In the second Championship (Toronto, 1977), *Chess 4.6* finished first with *Duchess* and *Kaissa* tied for second place. Meanwhile both *Chess 4.6* and *Kaissa* had acquired faster computers, a Cyber 176 and an IBM 370/165 respectively. The traditional exhibition match was won by *Chess 4.6*, indicating that in the interim it had undergone far more development and testing [17]. The 3rd World Championship (Linz, 1980) finished in a tie between *Belle* and *Chaos*. In the playoff *Belle* won convincingly, providing perhaps the best evidence yet that a deeper search more than compensates for an apparent lack of knowledge. In the past, this counter-intuitive idea had not found ready acceptance in the Artificial Intelligence community.

At the 4th world championship (New York 1983) yet another new winner emerged, *Cray Blitz* [18]. More than any other, that program drew on the power of a fast computer, here a Cray X-MP. Originally *Blitz* was a selective search program, in the sense that it could discard some moves from every position, based on a local evaluation. Often the time saved was not worth the attendant risks. The availability of a faster computer made it possible to use a purely algorithmic approach and yet retain much of the expensive chess knowledge. Although a mainframe won that event, small machines made their mark and seem to have a great future [19]. For instance, *Bebe* with special purpose hardware finished second, and even experimental versions of commercial products did well. The most recent 1986 event was also exciting. There *Hitech* seemed to dominate, but faltered in a better position against *Cray Blitz* allowing a four-way tie for first place. As a consequence an unknown micro-processor system, *Rebel*, nearly took it all, but in the end failed to clinch its final round game.

For the past two decades Canadian participation has been active and successful. Two programs, *Ostrich* and *Wita*, were at the inauguration of computer chess tournaments (New York 1970), and their authors went on to produce and instigate fundamental research in practical aspects of game-tree search [20-27]. Before its retirement, *Ostrich* (McGill) participated in more championships than any other program. Its contemporary, renamed *Awit* (Alberta), had a chequered career as a Shannon type-B (selective search) program, finally achieving its best result with a second place tie (New York 1983). Other successful programs were *Ribbit* (Waterloo), which tied for second in Stockholm (1974), *L'Excentrique* (McGill) and *Brute Force* (Manitoba). Currently the strongest Canadian program is *Sun Phoenix* (Alberta), a multiprocessor based system using work stations. In Cologne (1986) *Sun Phoenix* tied for first place with three others and hopes to do even better in its home town.

Implications

All this leads to the common question: When will a computer be the unassailed expert on chess? This issue was discussed at length during a panel discussion at the ACM 1984 National Conference in San Francisco. It is too early to give a definitive answer, even the experts cannot agree; their responses covered the whole range of possible answers from "in five years" (Newborn), "about the end of the century" (Scherzer and Hyatt), "eventually. - it is inevitable" (Thompson) and "never, or not until the limits on human skill are known" (Marsland). Even so there was a sense that production of an artificial Grand

Master was possible, and that a realistic challenge would occur during the first quarter of the 21st century. As added motivation, Edward Fredkin (MIT professor and well-known inventor) has created a special incentive prize for computer chess. The trustee for the Fredkin Prize is Carnegie-Mellon University and the fund is administered by Hans Berliner. Much like the Kremer prize for man-powered flight, awards are offered in three categories. The smallest prize of $5000 was presented to Ken Thompson and Joe Condon, when their *Belle* program earned a US Master rating in 1983. The second prize of $10,000 for the first program to achieve the equivalent of a Grand Master norm is to be awarded to *Deep Thought* later this summer, but the $100,000 for attaining world champion status remains unclaimed. To sustain interest in this activity, each year a $1500 prize match is played between the currently best computer and a comparably rated human.

One might well ask whether such a problem is worth all this effort, but when one considers some of the emerging uses of computers in important decision-making processes the answer must be positive. If computers cannot even solve a decision making problem in an area of perfect knowledge (like chess), then how can we be sure that computers make better decisions than humans in other complex domains -- especially in domains where the rules are ill-defined, or those exhibiting high levels of uncertainty? Unlike some problems, for chess there are well established standards against which to measure performance, not only through the Elo rating scale but also using standard tests [28] and relative performance measures [29]. The ACM sponsored competitions have provided nearly twenty years of continuing experimental data about the effective speed of computers and their operating system support. They have also afforded a public testing ground for new algorithms and data structures for speeding the traversal of search trees. These tests have provided growing proof of the increased understanding about chess by computers, and the encoding of a wealth of expert knowledge. Another potentially valuable aspect of computer chess is its usefulness in demonstrating the power of man-machine cooperation. One would hope, for instance, that a computer could be a useful adjunct to the decision-making process, providing perhaps a steadying influence, and protecting against errors introduced by impulsive short-cuts of the kind people might try in a careless or angry moment. In this and other respects it is easy to understand Donald Michie's view that computer chess is the "*Drosophila melanogaster* [fruit fly] of machine intelligence" [30].

References

1. A.G. Bell, *The Machine Plays Chess?*, Pergamon Press, Oxford, 1978.

2. D.N.L. Levy, *Chess and Computers*, Batsford Press, London, 1976.

3. T. Nemes, The Chess-Playing Machine, *Acta Technica*, Hungarian Academy of Sciences, Budapest, 1951, 215-239.

4. K. Zuse, Chess Programs, in *The Plankalkuel*, Rept. No. 106, Gesellschaft für Mathematik und Datenverarbeitung, Bonn, 1976, 201-244. (Translation of German original, 1945).

5. A.M. Turing, Digital Computers Applied to Games, in *Faster Than Thought*, B.V. Bowden (ed.), Pitman, London, 1953, 286-297.

6. A.D. de Groot, *Thought and Choice in Chess*, Mouton, The Hague, 1965 (2nd Edition 1978).

7. C.E. Shannon, Programming a Computer for Playing Chess, *Philosophical Magazine 41*, (1950), 256-275.

8. J. Kister, P. Stein, S. Ulam, W. Walden and M. Wells, Experiments in Chess, *J. of the ACM 4*, (1957), 174-177.

9. A. Bernstein, M. de V. Roberts, T. Arbuckle and M.A. Belsky, A Chess Playing Program for the IBM 704, *Western Joint Computer Conf. Procs.*, (New York: AIEE), Los Angeles, 1958, 157-159.

10. B. Mittman, A Brief History of Computer Chess Tournaments: 1970-1975, in *Chess Skill in Man and Machine*, P. Frey (ed.), Springer-Verlag, 1st edition 1977, 1-33.

11. G.M. Adelson-Velskii, V.L. Arlazarov, A.R. Bitman, A.A. Zhivotovskii and A.V. Uskov, Programming a Computer to Play Chess, *Russian Math. Surveys 25*, (Mar-Apr 1970), 221-262, Cleaver-Hume Press, London. (Translation of Proc. 1st summer school Math. Prog. v 2, 1969, 216-252).

12. J.E. Hayes and D.N.L. Levy, *The World Computer Chess Championship*, Edinburgh Univ. Press, Edinburgh, 1976.

13. D.E. Welsh and B. Baczynskyj, *Computer Chess II*, W.C. Brown Co., Dubuque, Iowa, 1985.

14. R.D. Greenblatt, D.E. Eastlake III and S.D. Crocker, The Greenblatt Chess Program, *Fall Joint Computing Conf. Procs. 31*, (New York: ACM), San Francisco, 1967, 801-810.

15. I.J. Good, A Five-Year Plan for Automatic Chess, in *Machine Intelligence 2*, E. Dale and D. Michie (ed.), Elsevier, New York, 1968, 89-118.

16. M.M. Newborn, *Computer Chess*, Academic Press, New York, 1975.

17. P.W. Frey (editor), *Chess Skill in Man and Machine*, Springer-Verlag, New York, 2nd Edition 1983.

18. R.M. Hyatt, A.E. Gower and H.L. Nelson, Cray Blitz, in *Advances in Computer Chess 4*, D. Beal (ed.), Pergamon Press, Oxford, 1985, 8-18.

19. D. Levy and M. Newborn, *More Chess and Computers*, Computer Science Press, Rockville MD, 2nd Edition 1981.

20. T.A. Marsland and F. Popowich, Parallel Game-Tree Search, *IEEE Trans. on Pattern Anal. and Mach. Intell. 7(4)*, (July 1985), 442-452.

21. M. Newborn, Unsynchronized Iteratively Deepening Parallel Alpha-Beta Search, *IEEE Trans. on Pattern Anal. and Mach. Intel. 10(5)*, (1988), 687-694.

22. T.A. Marsland and M. Campbell, Parallel Search of Strongly Ordered Game Trees, *Computing Surveys 14(4)*, (1982), 533-551.

23. J. Schaeffer, Distributed Game-Tree Searching, *Journal of Parallel and Distributed Computing 6*, (1989), 90-114, Academic Press.

24. M. Newborn, A Parallel Search Chess Program, *Procs. ACM Ann. Conf.*, (New York: ACM), Denver, Oct 1985, 272-277.

25. T.A. Marsland, A. Reinefeld and J. Schaeffer, Low Overhead Alternatives to SSS*, *Artificial Intelligence 31*, (Feb. 1987), 185-199..

26. J. Schaeffer, The History Heuristic, *ICCA Journal 6(3)*, (1983), 16-19.

27. M.S. Campbell and T.A. Marsland, A Comparison of Minimax Tree Search Algorithms, *Artificial Intelligence 20(4)*, (1983), 347-367.

28. D. Kopec and I. Bratko, The Bratko-Kopec Experiment: A Comparison of Human and Computer Performance in Chess, in *Advances in Computer Chess 3*, M. Clarke (ed.), Pergamon Press, Oxford, 1982, 57-72.

29. K. Thompson, Computer Chess Strength, in *Advances in Computer Chess 3*, M. Clarke (ed.), Pergamon Press, Oxford, 1982, 55-56.

30. D. Michie, Chess with Computers, *Interdisciplinary Science Reviews 5(3)*, (1980), 215-227.

AGT World Computer Chess Championship

	Participant		Round 1		Round 2		Round 3		Round 4		Round 5		Total Points	Final Place
1.	A.I. CHESS	USA												
2.	BEBE	USA												
3.	BP	USA												
4.	CENTAUR	USSR												
5.	CRAY BLITZ	USA												
6.	DAPPET	Netherlands												
7.	DEEP THOUGHT	USA												
8.	F/M CHALLENGER X	USA												
9.	HITECH	USA												
10.	KALLISTO	Netherlands												
11.	MEPHISTO X	Germany												
12.	MERLIN	Austria												
13.	MOBY	Britain												
14.	MUCH	Netherlands												
15.	NOVAG X	USA												
16.	PANDIX	Hungary												
17.	QUEST X	Netherlands												
18.	REBEL X	Germany												
19.	REX	USA												
20.	SHESS	Netherlands												
21.	SUN PHOENIX	Canada												
22.	WAYCOOL	USA												
23.	Y!89	Sweden												
24.	ZARKOV	USA												

Table of Participants

Program	Authors	Contact Address
A.I. CHESS	Martin Hirsch	P.O. Box 3535, Seal Beach, CA 90740-7535, USA
BEBE	Tony & Linda Scherzer	SYS-10 Inc., 2111 Stonington Ave, Hoffman Estates, IL 60195, USA
BP	Robert Cullum	P.O. Box 111, Prospect Heights, IL 60070, USA
CENTAUR	Victor Vikhrev	Fl.33, Kulakov Str. 27, Moscow, USSR
F/M CHALLENGER X	Dan & Kathe Spracklen, Ron Nelson	c/o Fidelity International, 13900 N.W. 58th Court, Miami, FL 33014, USA
CRAY BLITZ	Bob Hyatt, Albert Gower, Harry Nelson	c/o B.H., #105 Green Wing Circle, Pelham, AL 35124, USA
DAPPET	Dap Hartmann, Peter Kouwenhoven	c/o D.H., P.O. Box 9513, Leiden Univ., 2300 RA, Leiden, The Netherlands
DEEP THOUGHT	Feng-hsuing Hsu, Thomas Anantharaman, Mike Browne, Murray Campbell, Peter Jansen, Andreas Nowatzyk	c/o F.H., Computer Science Dept., Carnegie-Mellon Univ., Pittsburgh, PA 15213, USA
HITECH	Hans Berliner, Carl Ebeling, Murray Campbell, Gordon Goetsch, Andy Gruss, Andy Palay, Larry Slomer	c/o H.B., Computer Science Dept., Carnegie-Mellon Univ., Pittsburgh, PA 15213, USA
KALLISTO	Bart Weststrate	Pietheinstr. 46, Wormerveer 1521 KZ, The Netherlands
MEPHISTO X	Richard Lang	c/o Hegener+Glaser AG, Arnulfstrasse 2, D-8000 Munchen 2
MERLIN	Hermann Kaindl, Helmut Horacek, Marcus Wagner	c/o H.K., Marxergasse 18/2/1, 1030 Vienna, Austria
MOBY	Mark Taylor, Greg Wilson, David Levy	c/o M.T., 38 Cambridge Rd, W. Wimbeldon, London SW 20, England
MUCH	Jaap van den Herik, Roget Hünen, Henry Nefkens, Tom Pronk	c/o J.H., Computer Science Dept., P.O. Box 616, Maastricht 6200 MD, The Netherlands
NOVAG X	David Kittinger	5965 Arbon Ave, Mobile, AL 36608, USA
PANDIX	Gyula & Zsuzsa Horvath	Laborfalvi R. ul. 2, 1041 Budapest, Hungary
SUN PHOENIX	Jonathan Schaeffer	Computing Science Dept., University of Alberta, Edmonton, Alberta, T6G 2H1
QUEST X	Frans Morsch	Langenhorst 56, 2402 PX Alphen a/d Rijn, The Netherlands
REBEL X	Ed Schröder	c/o Hegener + Glaser AG, Arnulfstrasse 2, Munich 2 D-8000, W. GERMANY
REX	Don Dailey, Larry Kaufman	c/o D.D., 2002 Langdon Rd, Apt. 31, Roanoke, VA 24015, USA
SHESS	Ard van Bergen	T. à Kempisweg 92, Utrecht 3552 CD, The Netherlands
WAYCOOL	Ed Felton, Steve Otto, Rod Morison	c/o E.F., 206-49 Cal Tech, Pasadena, California, 91125
Y!89	Ulf Rathsman, Lars Hjorth, Sandro Necchi	c/o L.H., Vattugatan 3A, S-172 34 Sundbyberg, Sweden
ZARKOV	John Stanback	4237 Cape Cod Circle, Fort Collins, CO 80525, USA
AWIT'83*	Tony Marsland	Computing Science Dept., University of Alberta, Edmonton, Alberta, T6G 2H1
BELLE'81*	Ken Thompson, Joe Condon	c/o K.T., Room 2C519, Bell Labs, Murray Hill, NJ 07974, USA

* Alternate entries.

Participant Technical Table

Program	Country	Operators	Language	Code+Data K bytes	Book ×1000	Computer	Nodes/sec ×1000
A.I. CHESS	USA	Martin Hirsch	C,Assembler	200+64	10	Dyna 8086	2.5
BEBE	USA	Tony & Linda Scherzer	Assembler	16+40	4	Sys-10	45
BP	USA	Kevin O'Connell	C,Assembler	325	15	Unisys PW800*	0.6
CENTAUR	USSR	Victor Vikhrev	Pascal	120+30	16	IBM PS2/80*	1
CRAY BLITZ	USA	Bert Gower	Fortran, CAL	500+1M	5	Cray YMP	100
DAPPET	Netherlands	Dap Hartmann	Turbo Pascal	500	15	Toshiba	0.2-0.8
DEEP THOUGHT	USA	Feng-hsuing Hsu	C,Assembler	n/a	n/a	VLSI-Sys	1000
F/M CHALLENGER X	USA	Sid Samole	Assembler	32+16	32-64	MC68030	10
HITECH	USA	Murray Campbell	C	700+24M	5.8	VLSI-Sys	100
KALLISTO	Netherlands	Tom Pronk	Assembler	20+8	6	Apple II	3-6
MEPHISTO X	Germany	Richard Lang, Ossi Weiner	Assembler	128	60	MC 68020	2
MERLIN	Austria	Helmut Horacek	Pascal	n/a	n/a	IBM 3090	n/a
MOBY	Britain	Greg Wilson, Mark Taylor	Occam	n/a	8	Meikos Transp.	0.25
MUCH	Netherlands	Jos Uiterwijk, Harm Bakker	C	170-2.5M	4.5	Sun 4*	3
NOVAG X	USA	David Kittinger	Assembler	96+8	36	INTEL 6502	2.8
PANDIX	Hungary	Gyula & Zsuzsa Horvath	C, Assembler	50+10	7.4	Sanyo 386*	0.30
QUEST X	Netherlands	Frans Morsch	Assembler	32+8	8	INTEL 6502	8
REBEL X	Germany	Helmut Weigel	Assembler	32+8	16	INTEL 6502	2
REX	USA	Don Dailey, Larry Kaufman	Pascal	48	n/a	Unisys PW800*	n/a
SHESS	Netherlands	Ard van Bergen	Fortran	60+2.5M	2	Vax 8600*	0.4
SUN PHOENIX	Canada	Jonathan Schaeffer	C	200+2M	20	20×SUN 4*	10
WAYCOOL	USA	Murphy Jones	C	n/a	n/a	Intel 512-Cube	n/a
Y!89	Sweden	Lars Hjorth	Assembler	32+128	100	INTEL 6502	5
ZARKOV	USA	John Stanback	C	100+200	16	HP 9000/835	2.5
AWIT'83†	Canada	n/a	Algolw	750	10	Amdahl	0.1
BELLE'81†	USA	n/a	C	n/a	n/a	LSI Hardware	150

* We gratefully acknowledge the support of the following local companies for providing on-site equipment: AGT, ARC, Compu-ware, IBM, Sun Microsystems and Unisys.

† Alternate entries.

Program Descriptions

These descriptions are abstracted from summaries provided by the program authors.

A.I. Chess

A.I. Chess uses a fairly complicated algorithm combining full-width search, selective search, and a "layered" quiescence search which behaves differently at differing levels in the search tree. The program performs an iterative full-width search using a modified form of the Principal-Variation-Search (PVS) algorithm. On top of this, it does a combined selective/quiescence analysis. A. I. Chess has the unusual feature of sometimes re-searching a "quiescence node" with a full-width investigation.

The quiescence search incorporates a detailed "threat analysis" and therefore, the program spots many combinations long before a contrasting "brute force" approach would find them. The gain (from needing less full-width plys) seems to exceed the loss in speed by a significant amount.

Position evaluation starts by considering if the side to move is threatened with pawn promotion, check, or double attack, or has trapped, pinned, or skewered pieces. Penalties similar to swap-off scores are imposed if the position is too deep to merit a re-search. Scores are then added for other tactical patterns, pressure on pieces and pawns, development, King safety, passed pawns, pawn structure, outposts, and mobility.

Some types of endgame positions are scored differently, by pattern recognition processing. The program is alert to simplifications, and to tactics involving passed pawns.

BeBe

In early 1980 SYS-10 tried new hardware techniques needed for their mini/mainframe processor in co-processors for BeBe's CPU. Each co-processor takes over a specific function from the main CPU.

The first co-processor does the complete task of move list generation. The actual unit is divided into two processors which function in parallel: one that finds pieces and one that calculates and stores moves. This parallelism provides results more than 25 times faster than software.

A second co-processor performs the position scoring function. The scorer "looks at" the output of the move generator and uses the moves to calculate values for piece position mobility and co-operation. The scorer functions in parallel with the move generator.

BeBe operates at four distinct levels:

* Software does I/O, timekeeping, book lookup, search depth control, and overall system control.

* Special CPU instructions do move list sorting, internal board update for making and unmaking moves, the alpha-beta minimax control, keeping track of "killer moves", building bit maps of piece locations, and some board scoring functions.

* The co-processors perform move list generation, and some of the board scoring functions.

* The self-activated parallel processor determines if either king is in check and determines the attack-defender count for any square. Because it self-starts, the answers for both kings are ready before the software can ask the question.

BP

BP spends 95% of its time in board evaluation and the rest on move generation and search. Because of this, it must do a selective search. In fact, BP does move pruning at every level of the search tree.

Centaur

Centaur is a new chess-playing program with the heuristic search to consider the decisive series of moves. The algorithm is based on the probablic logic and uses the fuzzy value of positions. The depth of search is not limited. The whole information about all the series of moves is kept in RAM and is used to determine the decisive series of moves. Centaur features a low number of position analyzed. This is compensated with thoroughness of the position evaluation.

Cray Blitz

Cray Blitz owes its success to several features:

(1) It runs on the most powerful Cray computer system available, currently the Cray YMP with eight processors, 128 million words of memory, and a total speed of 1,333 mips.

(2) The program subscribes to the maxim that "pawns are the soul of chess" and has complex pawn evaluation procedures to guide the program. It understands passed pawns, pawn structure, passed pawn races, rook pawn endings and many other concepts that allow it to play many pawn endings perfectly without deep searches (although it can search incredibly deep anyway).

(3) The program currently searches about nine plies deep in the middle-game.

(4) The program has a sophisticated quiescence search, far beyond the usual "examine all captures." It pushes passed pawns, tries pawn levers, follows checks and captures near the king, and tries other moves that seem to affect the stability of the position, and it follows these moves to extreme depths.

(5) It uses a sophisticated parallel processing algorithm to take advantage of as many processors as are available (currently eight, but this will increase in the future). Current work is being done to allow Cray Blitz to search in parallel on machines with multiple processors and to spread the work over many Crays connected by some type of LAN, speeding up the search even further.

Dappet

Dappet uses the NegaScout algorithm enhanced with refutation tables, killer heuristic, history heuristic and transposition tables (700,000 entries) to search the game tree. The strategy used is basically brute force, with selective deepening of forced lines of play. The openings book consists of some 15,000 positions.

Deep Thought

Deep Thought uses two custom processors to achieve a speed of 720000 positions/sec. Each custom processor includes a VLSI chip that generates and executes legal chess moves at over 500000 moves/sec. The custom processor evaluates each position reached by adding up values stored in tables for material balance, piece placement and pawn & rook structure (viewed through 8 sliding windows 3 files wide), all in about 2 microseconds. The main program runs on a SUN workstation computer, and makes its move decision by exhaustively examining all possible move combinations upto a certain length, typically 10 plies (or half moves) in 3 minutes. The main program uses the custom processors to evaluate the last 3 plies. Because each additional ply involves examining about 6 times as many positions as before, this allows the main program to run at close to the speed of the custom processors, while retaining the flexibility of a general purpose computer for the remaining plies of the move combinations examined. This flexibility is used to examine forcing lines more deeply using a new technique called Singular Extension to identify forcing lines. During a typical 3 minute search, the deepest lines are 15 plies deep, and in some tactically complicated positions lines as deep as 40 plies have been observed. The main program is also responsible for updating the tables used by the custom processors for evaluating positions. The values in the tables are based on a number of positional features deemed important by human chess experts. Each feature is weighted by a polynomial of the material on the board, in order to smoothly transition between opening and endgame positions which may require different importance to be attached to each positional feature. The parameters of the polynomials are tuned automatically to a collection of about 900 games between human GMs and IMs, so as to make the program (handicapped to 5 plies of search) agree as often as possible with the actual move made by the human player.

Fidelity/Motorola Challenger X*

The Fidelity/Motorola Challenger relies for its strength on a combination of state-of-the-art, microcomputer hardware and a chess algorithm that has undergone continuous full-time development for over ten years.

The central feature of the hardware is a Motorola 68030 processor, hand-selected by Motorola engineers to run at the fastest possible clock speed. The exact speed will not be known

* An X after a program name means that it is an experimental version of a commercial chess program.

until just prior to tournament time. The system is completed by 32K of program ROM, 64K of opening book ROM, 16K of program RAM, one megabyte of dynamic RAM for transposition tables, and a special 16K of non-volatile RAM that supports the learning feature.

The learning feature is just one facet of a multi-faceted chess algorithm. The program is basically brute force in origin with evolution to incorporate extensive positional analysis and selective extensions during the quiescence search. The positional analysis incorporates extensive heuristics for king safety and pawn structure. Numerous end-game specific routines are incorporated, including mate with bishop and knight, complete evaluation of king and pawn vs. king, probably outcome of a pawn race, square of the pawn, bishop and rook pawn of the opposite color, the Philidor and Lucena positions and others. Dynamic recognition of minimum mating material, fifty move rule, and repetition of position assist in forestalling heartbreaking draws in otherwise won positions. The search algorithm uses a depth first, alpha-beta search with the zero width window technique (PVS). The search proceeds iteratively with a quiescence search incorporating captures and certain threats appended beyond the nominal depth. The program will not perform an evaluation on a position where either king is in check. The check must first be resolved by showing the existence of an escape move or mate. Iterations are finally halted under the direction of a time control algorithm which is dynamically incorporated for up to 40 moves in the root position. Two killer moves are stored at every ply. The program performs a preliminary sort on the ply above the quiescence search. The search is supported by extensive transposition tables incorporating random numbers selected using BCH theory.

Hitech

The key idea of Hitech is combining a fast search with pattern recognition capabilities. Hitech was the world's highest-rated program from 1985, the year it won the ACM North American Computer Chess Championship in Denver, through mid- 1988, when it became the first program to reach the Senior Master level (surpassing a rating of 2400). In addition, Hitech has won the Pennsylvania State Championship the last two years running.

Lachex

Lachex is specifically designed for the architecture of the Cray XMP and YMP series of machines. The highly repetitive parts of the program are written in assembly language, the rest in Fortran. Low level parallelism is achieved by extensive use of vector functional units and pipelining. High level parallelism is obtained by means of multiple independent processors splitting up the search using a self-scheduling algorithm and communicating with each other through a large common memory.

The search is basically alpha-beta with iterative deepening. In the initial depth one search each root move is actually scored and the list of moves ordered accordingly. Best moves at subsequent iterations are moved to the top of the list. Scouting is used at ply one only - the first move in the list is scored and the remaining moves are tested with a minimal window. Forward pruning is done with a positional estimator at nodes below the horizon and with the null move algorithm above. Moves out of check above the horizon extend the search depth for that path by one, but by two if the check is discovered or double. Selective searches below the horizon include captures, promotions, castling, and some checking moves.

Lachex spends 1/3 of its time generating moves, 1/3 doing bookkeeping, and 1/3 evaluating leaf nodes. The evaluation function is symmetric wherever possible. Mobility, pawn structure, king safety, piece placement and other features make up the evaluation function. Some strategy is incorporated at the root by shifting the minimal window to bias certain types of moves. There is a transposition table which can be as big as 32 million positions, on a 64 million word machine.

Mephisto X

Mephisto is a further development of the Mephisto Almeria program which won the World Micro Computer Chess Championship held in Almeria, Spain during September 1988.

The Almeria program is completely new and was started in December 1987. It is a selective program with a selection mechanism that prunes bad moves from the search tree and extends the depth of the search for interesting lines. The program contains a large amount of chess knowledge and uses hash tables which are of especially great benefit in the end game.

For Edmonton, extra chess knowledge has been added and the tactical strength has been improved with adjustments to the search extension algorithms.

Merlin

The primary goal was to combine the development of new methods with their actual use (together with conventional methods) in a competition chess program. While most of the better programs in this domain use very little domain knowledge, we tried to achieve improvements by incorporating strategic and positional (rather *static*) knowledge and by providing means for handling uncertainty using *meta*-knowledge.

However, the use of such knowledge did not only result in a better treatment and "understanding" of long-range aspects but also in slowing down the program with the implication of reducing its search. Unfortunately, the tactical abilities depend seriously on the depth of the search, and tactics are important in this highly dynamic domain. Therefore, we investigated methods for improving the tactical abilities including *dynamic* knowledge explicitly, for searching to variable depth were investigated.

The results have shown an interesting substitution of *knowledge for search*. Its performance on interesting positions is impressive. The knowledge selected (*a priori*) by humans showed its best in such positions, while in the general situation almost all knowledge is likely to be incomplete. This instance of "generality vs. power" is of special interest also for other domains, emphasizing the power of searches to *discover* detailed and dynamic issues.

Moby

This project is developing a multi-processor chess program to run on a large (200 to 400 processor) Meiko Computing Surface installed at the University of Edinburgh as part of the Edinburgh Concurrent Supercomputer Project (ECSP).

Meiko's Computing Surface range of parallel computers are based on INMOS T800 transputer chips. Each transputer contains a 10 MIPS CPU, four 20 MB/s inter-processor communications links, plus 4 KB of on-chip RAM and up to 16 MB external RAM. (The ECSP machine's processors currently have 4 MB RAM each.) The Computing Surface's processors are connected to one another through switching chips, which allows the machine to be electronically reconfigured to suit the needs of individual programs.

The Moby chess program is a descendant of Cyrus 68K, whose development was begun by Mark Taylor and David Levy in 1985. Moby uses conventional search techniques, but distributes the search across the available processors in a homogeneous fashion, i.e. all processors are carrying out the same type of operations, rather than some processors doing deep "scout" searches while others do more complete searches guided by the information returned by the scouts. Load balancing is achieved by processor overloading - each processor supports several search processes, time-slicing between them. In addition, each processor supports a hash table manager responsible for part of the global transposition table. One distinguished processor acts as a system master, interacting with the user and handling file i/o when the opening books are consulted.

Much

Much consists of several programs. The user-interface program accepts a move from the operator and subsequently generates evaluation tables for the search program. The user-interface program also handles time control, the opening library, and the endgame library. The search program receives the board position and evaluation tables from the user-interface program. The evaluation tables are tuned with the opening played. Before each move they are incrementally updated according to the board position (strategical evaluation of squares), but also bonus points are provided (to mention a few) to undeveloped pieces (opening), the pair of Bishops in open positions (midgame/endgame), the color of the Pawns and the Bishop on the board (endgame). Moreover, several plans are encouraged. The configuration belonging to the execution of a plan is supplied with bonus points such that every piece and pawn involved tries to reach the plan-ideal square. The plan as a whole, once started to be carried out, increases the bonus points for every piece/pawn to be played at each move. Much then searches until it is interrupted by the user-interface program. The search program, based on the alpha-beta algorithm and its refinements, uses PVS-search, killers and transposition tables. Move generation is done incrementally. Much uses specialized sub-programs to handle the KBBK, KBNK, KBPK and KNPK endgames. These programs use a goal-directed search.

Novag X

The Novag Super Expert program entered in the tournament is the new commercially available Super Expert 'B' program running at 9 mhz on a 6502. The program includes a user selectable selective search which has been shown to be effective against other computer programs although the gain against human opponents does not appear to be as great.

Recent work with selective searching has demonstrated a clear gain against other computer programs and some gains against human opponent's.

Earlier work involving one ply extensions to try and resolve tactically unclear positions has been retained and modified slightly. These extensions have proven very cost effective.

Pandix

The program runs on an IBM PC or any compatible. The main basic principle of the development is to use selective algorithms whenever it is possible. The algorithms implementing the selectivity are based, almost without exception, on origonal ideas. The development, so far, shows there is an unambiguous parallel between the progress of the program (the increase of its playing strength) and the degree of the selectivity implemented in it. It means that the increasing chess intelligence of the evaluation function can help the selective thinking algorithm to an greater degree. The upper limit of the development of this model is not visible yet.

Quest X

Quest is a classical brute-force Shannon-A program, using a full width search strategy with alpha-beta pruning and quiescence search. The evaluator function is dominated by piece activity. Positional heuristics include pins, X-ray attacks, king safety and pawn structure. A static exchange evaluator is used to filter the capture moves in the quiescence search. Quest does not use a planner or preprocessor stage to direct positional evaluation. Instead it relies on true endpoint evaluation to maximize the reliability of its positional judgement in turbulent lines.

Rebel X

The Mephisto Rebel has to be defined in between a Shannon A and a Shannon B type of chess program.

To all brute force calculations a fixed ply depth quiescence search is added. Capturing moves and checks are extended more deeply. The evaluation function integrated much chess-knowledge, so the program also finds good positional moves.

Rex

The Basic Structure of Rex is not unusual for it contains most of the elements of the well known 'generic' brute force chess programs. It employs the standard alpha/beta iterative search algorithm with certain tactical moves being extended on, such as checks and pawn moves to the 7th rank.

The previous version of Rex focused on positional knowledge by employing a sophisticated pre-processor which was driven by a programmable rulebase. The rulebase language was relatively simple for a human 'Expert' to learn and made it easy to modify Rex's evaluation function. Although Rex seemed to outplay most of its computer opponents positionally, it often lost games tactically due to it's inferior search speed.

Therefore it was decided to focus most of our attention on its tactical play. After much experimentation with various ideas including singular extensions and an assortment of move extensions, it was decided to use an 'extended quiescence search'. At the end of the main search, a few plies (3 or 4) of 'itermediate quiescence' are done. The moves included are decided upon by a brief 'static' analysis which determines if the candidate move is likely to affect the results of the search. After this stage is completed we do a restricted form of the standard capture search employed by most brute force programs is done.

Shess

It has a few features that are not found in most other chess programs. The first is that its evaluation contains the full databases for three piece endgames (KQK, KRK and KPK), allowing an exact evaluation for any line ending in such an endgame. The other one is its activity when the opponent is to move. Most programs prepare a reply to some expected move; Shess, on the other hand, always analyzes the current position and by updating its transposition table it effectively prepares a reply to any move of its opponent.

Sun Phoenix

Phoenix uses state-of-the-art search techniques including singular extensions, minimal window searching, transposition tables, and the history heuristic. The program has lots of chess knowledge, including an extensive long range planner.

Phoenix is capable of running in parallel on a network of machines. In tournament mode, Phoenix is actually two programs: ParaPhoenix and ParaMinix. ParaPhoenix uses 10 Sun 4s to build trees looking for the best positional move. ParaMinix uses 10 Sun 4s to build trees looking for the best tactical moves. Because of its specialized task, ParaMinix is capable of searching 1-2 ply deeper than ParaPhoenix. ParaMinix has the ability to veto ParaPhoenix's move choice, if a tactically superior move is found. The parallelism is achieved using the Dynamic Principal Variation Splitting Algorithm.

Waycool

Waycool is an experiment in parallel computing. The authors have taken the standard techniques of computer chess and modified them for distributed-memory computers with hundreds of processors. This has yielded insight into many issues surrounding algorithms, programming, debugging, and performance analysis for parallel computers.

Waycool runs on an NCUBE/10 system with 512 processors, and a total of 256 megabytes of memory. This configuration runs about 170 times faster than a single processor with 512k memory. In the future, Waycool will run on a variety of parallel computers, including the Meiko Computing Surface.

Y!89

Y!89 uses a full, partly extended, width iterative principal variation search with capture and promotion searches in terminal nodes. The program is designed to be used in a cheap commercial environment, thus the work memory is still just 4 kbytes of RAM, and the good old 6502 eight bit processor is used in tournaments emulated by the also commercially available Turbo kit. The search is fast for a micro, and includes detection of repeated positions (actual as well as potential), and performs extensions for check evasions, passed pawn moves and some king moves in pawn endgames.

Most of the material and positional evaluation is made incrementally by the means of "material value tables" and "positional score boards" for each piece type, created once for each position of the game with the computer to move. Some "absolute" evaluation is also done, e.g. for static evaluation of "unstopable passed pawns" and pawn structure.

The timing algorithm was recently changed to spend on the next few moves any time saved when using the opening library (or thinking on the opponent's time, etc.). This has several times helped to find critical moves in the early midgame.

Zarkov

Zarkov employs a full-width alpha-beta search with the standard techniques such as iterative deepening and a two pawn "window". The full-width search is extended an extra ply for check evasion moves, responses to pawn moves to the seventh rank, certain re-captures, and at horizon nodes when the side to move must defend threatened pieces. A quiescence search containing captures, pawn promotions and certain checking moves is conducted for up to 11 ply beyond the full-width portion of the search. The author is not a good chess player, but has attempted to instill Zarkov with a decent positional understanding by including about 50 simple but well-refined positional heuristics in the evaluator. The program can play well when it manages to achieve a good, open position, but it lacks the "human" ability to plan well and often gets into trouble in the opening against strong players.

THE 6TH WORLD COMPUTER CHESS CHAMPIONSHIP
1989 TOURNAMENT RULES

1. Each entry is a computing system with one or more human operators. An operator cannot simultaneously be responsible for two entries. A listing of all chess-related programs running on the system must be available on demand to the TD (Tournament Director).
2. Participants must attend the organisational meeting at 12 noon on Sunday 28 May 1989, to complete the official registration and to agree on a finalization of the tournament rules. The TD has the right to choose an alternative to replace an entrant who fails to appear.
3. The competition is a five round Swiss-style tournament. The first and second rounds are scheduled for 1 pm and 7 pm respectively on Sunday 28 May 1989. The remaining rounds will be on succeeding evenings at 7 pm. The organizers are permitted to make minor changes to the starting times to suit local conditions.
4. Trophies and cash prizes will be awarded to the first three finishers. The order of the finish will be determined by the number of points earned. Ties will be resolved by using the sum of the opponents' points in the last 4 rounds. Beyond that an agreed standard tie-breaking rule will be used.
5. The Shannon Perpetual trophy will be awarded to the champion. Should the 5-round tournament not yield a clear winner, a sudden death play-off for the title will take place as soon as practical after the final round. This might be on Thursday 1 June, or as part of another event.
6. Unless otherwise specified, rules of play are identical to those for "human" tournaments. If a point is in question the TD makes the final decision, possibly after consulting an ICCA appointed committee.
7. Games are played at a speed of 40 moves per player in the first two hours of play, and 20 moves per player per hour thereafter.
8. Each game is to be played on the official chess board provided by the organizers. The official clocks will be the ones designated by the organizers.
9. All offers for a draw must first be cleared with the TD. A computer generated draw offer can be communicated to the opposing program, which can then accept or reject it. The operator is purely passive and cannot initiate a draw offer, and cannot participate in the acceptance/rejection decision.
10. The TD has the right to adjudicate a game after five hours of total clock time. The adjudication will be based on perfect play by both sides. The TD will make every effort to avoid adjudication by requiring the programs to continue play until the situation on the board is clear. Before the start of each round the operator must declare to the tournament director and the opponent any computer availability constraints which may interfere with play beyond the minimum requirement.
11. Whenever it is detected that an operator has incorrectly entered a move, or played the wrong move on the official board, the TD must be notified immediately. The TD will back up the clocks to the point at which the last correct move was made, provided the clock times have been recorded. If the offending operator has not recorded the time, that team's clock will not be adjusted. The clock of the innocent side will be backed up by an amount equal to the average time taken per non-book move in the game so far.
12. The operating console must be at the tournament site and must communicate directly with the remote system. Thus a human intermediary at a remote location is not permitted.
13. Each team that uses a terminal must position the display on the game table in such a way that the opponent has a good view of it. An operator can only (1) type in moves and (2) respond to requests from the computer for clock information. If the operator must enter extra information, this must be

approved ahead of time by the TD. Entry of extra commands to display the board or the moves made so far is discouraged. The TD must be consulted before any special measures are taken to clear the communication line of noise or to interrogate the computer/communication to determine if they are still operational.

14. A team may ask that the TD stop the clock at most twice during the course of a game. The clock must be restarted each time after at most 15 minutes. If a team using a remote computer can establish that the problems are in the communication network and not in its own computing system, the TD can permit additional time-outs.

15. If a failure occurs during the course of a game (a program crash or communication failure leading to an equivalent situation) only the position, move number and clock time may be communicated to the computer. The program should be capable of automatically preserving and restoring from a "restart file" the current setting of the parameters. Other restart mechanisms such as re-accepting all the moves made since the start of the game (and hence re-computing the parameter settings) are permitted.

16. A team must receive approval of the TD to change from one computing system to another.

17. At the end of every game, each team must provide the TD with a copy of the offical score sheet (or computer generated equivalent), properly signed by both teams.

The Fifth World Computer Chess Championship* Cologne, 1986

Helmut Horacek

Research Unit for Information Science and Artificial Intelligence
University of Hamburg

This was clearly the most exciting finish of a World Computer-Chess Championship I have ever seen. In the last round four programs still had chances to become champion and three of them actually were tipped for champion in the last hours. In the end, four programs tied for first place and Cray Blitz maintained its title due to the Buchholz system. It was the agreed opinion of the audience that it was Hitech which had played the best chess in this tournament. The best microcomputer program and also the best European program was Rebel, the greatest surprise in Cologne. At a certain moment in the last round it even seemed the World Champion to be.

Several companies have sponsored this tournament. Compaq computer provided the terminals and Deutsche Mailbox GmbH took care of the communication facilities. Further sponsors were Hegener & Glaser and Deutsche Messe AG. The organizing committee was formed by Frederic Friedel, David Levy, Tony Marsland, Monty Newborn and Dieter Steinwender. Horst Lynsche took care of the computer-communication links. The games took place at the Cologne Messe, as part of the annual computer exhibition. Mike Valvo was the Tournament Director as he had been for the last five years at ACM tournaments. He also commented the games for the audience with the support of the Grandmasters Vlastimil Hort and dr. Helmut Pfleger. Participants in this tournament came from Canada, Germany, Great Britain, Hungary, the Netherlands, Sweden and the United States. Unfortunately the program from Hungary could not be installed completely on the machine available and it only played a few moves. A complete record of the games can be found somewhere else in this Journal.

The overall strength of the programs differed very much, which was proved by the absence of draws in the first two rounds. I feel that the average strength was below that of the last championship in New York and especially below that of the last ACM tournaments. This was mainly due to the fact that some of the top programs were missing: the former World Champions Belle (1980) and Nuchess (as a successor of the World Champion Chess 4.7 from 1977) and the vice world champion of 1980, Chaos. That the games were scheduled during the working hours caused the absence of some European programs that rely on mainframes (Chess 0.5X and Merlin were certainly among them). Others were not able to exploit their machines fully as they were not the only users.

CHESS QUALITY

The most convincing play was shown by Hitech, although some weak moves in the last round versus Cray Blitz caused its decisive defeat. The fast special- purpose hardware, coupled with the admittedly simple but extremely relevant knowledge has led to a very strong program. Especially the mating sequence against Schach 2.7 was a real nugget, that even Grandmaster Hort did not believe when Hitech had announced it. The positional play of Cray Blitz clearly was the weakness of this tactically very strong program. It also was rather fortunate that the game versus Schach 2.7 had to be adjudicated according to the rules of the tournament. Although the final position was actually won, the winning line is extremely difficult to find for a program. Bebe played most of its games rather solidly, but it sometimes was very

* Abridged from the ICCA Journal, Vol. 9, No. 2, pages 92-93.

careless about the protection of its King. This defect caused material loss against Cray Blitz and also against Rebel. But, nevertheless it had won the latter game from a very dynamic position, which was sufficient to reach the third place. After the first-round defeat (which was caused by a program bug according to its author) Phoenix had played four solid games, but the opponents were not as strong as those of the other programs that tied for the first place. The play of Rebel was very sound in all its games. The mistakes the program has made occurred in positions that are very hard to play for a program. Furthermore, it had the highest Buchholz score of all programs. Bobby succeeded in beating the World Champion in a beautiful, positional game. Theoretically, it could also become World Champion by a win against Phoenix in the last round, but it never had a favorable position in this game. It actually lost due to a trapped piece. Plymate found a beautiful many-ply mate against Lachex, which preferred not only in this game to centralize its King in the middle game. Mephisto lost two games with uneven material of about the same value, which is a situation that all the programs usually have difficulties with. Schach 2.7 is clearly stronger than its score indicates, it was the only program to play Cray Blitz and Hitech as well. BCP and Awit have some interesting features but these programs proved too unbalanced for the whole game of chess and scored only few points.

EVALUATION

As a final remark, I guess that a chess-player unfamiliar with computer chess will find the uneven capabilities of the programs very striking. On the one hand, a program's play can be very convincing if it discovers the essential issues in a position. For instance, Grandmaster Hort was very impressed by the mating sequence Hitech had found. In another situation, Hort had announced three different mate threats in the game Vax Chess against Lachex with no adequate defense against any of them. However, there existed a move warding off all mate threats by sacrificing a piece which would restore the material equilibrium. On the other hand, many tactical (some of them due to the well-known horizon effect) and considerable positional errors occurred in which the usual heuristics showed themselves somewhat inadequate. I am very curious to know how, and to what extent, this deficiency will be overcome in the near future. This is necessary if the programs are to compete successfully on the grandmaster level.

Table of Results for the 5th World Chess Championship

Participant		Perf	Round* 1		Round 2		Round 3		Round 4		Round 5		Total
1. CRAY BLITZ†	USA	2290	w20	1	b6	0	w3	1	b14	1	w2	1	4
2. HITECH	USA	2303	b18	1	w14	1	b7	1	w5	1	b1	0	4
3. BE-BE	USA	2215	b16	1	w15	1	b1	0	w11	1	b5	1	4
4. SUN PHOENIX	CDN	2318	b5	0	w11	1	b18	1	w7	1	b6	1	4
5. REBEL	NL	2235	w4	1	b12	1	w6	1	b2	0	w3	0	3
6. BOBBY	D	2188	b19	1	w1	1	b5	0	w8	1	w4	0	3
7. PLYMATE	S	2102	b21	1	w8	1	w2	0	b4	0	w12	1	3
8. MEPHISTO COLOGNE	D	1973	w9	1	b7	0	w17	1	b6	0	w14	1	3
9. DUTCH	NL	1828	b8	0	w19	1	b11	½	w15	½	w13	1	3
10. NONA	NL	1552	b14	0	w18	0	b21	1	b22	1	w15	1	3
11. ADVANCE 68	GB	1855	w17	1	b4	0	w9	½	b3	0	w19	1	2½
12. LACHEX	USA	1840	b13	1	w5	0	b16	½	w18	1	b7	0	2½
13. OSTRICH	CDN	1689	w12	0	b20	1	b15	½	w16	1	b9	0	2½
14. SCHACH 2.7	D	1716	w10	1	b2	0	w22	1	w1	0	b8	0	2
15. CYRUS 68K	GB	1572	w22	1	b3	0	w13	½	b9	½	b10	0	2
16. VAXCHESS	GB	1561	w3	0	b23	1	w12	½	b13	0	w17	½	2
17. CHAT	D	1533	b11	0	w21	1	b8	0	w19	½	b16	½	2
18. BCP	GB	1645	w2	0	b10	1	w4	0	b12	0	w20	½	1½
19. ENTERPRISE	DNM	1591	w6	0	b9	0	w20	1	b17	½	b11	0	1½
20. AWIT	CDN	1476	b1	0	w13	0	b19	0	w21	1	b18	½	1½
21. REX	USA	1157	w7	0	b17	0	w10	0	b20	0	b22	1	1
22. SHESS	NL	855	b15	0		1	b14	0	w10	0	w21	0	1
23. KEMPELEN ATARI	H	767		0	w16	0							0

* The letter indicates the player's colour (white or black). The number identifies the opponent that the participant faced. The next number shows whether the participant won (1), lost (0), or tied (½).

† Awarded 1st place title on tie break.

Scoresheet

White			Black
Date	Round	Board	Opening
1		26	
2		27	
3		28	
4		29	
5		30	
6		31	
7		32	
8		33	
9		34	
10		35	
11		36	
12		37	
13		38	
14		39	
15		40	
16		41	
17		42	
18		43	
19		44	
20		45	
21		46	
22		47	
23		48	
24		49	
25		50	

Scoresheet

White			Black
Date	Round	Board	Opening
1		26	
2		27	
3		28	
4		29	
5		30	
6		31	
7		32	
8		33	
9		34	
10		35	
11		36	
12		37	
13		38	
14		39	
15		40	
16		41	
17		42	
18		43	
19		44	
20		45	
21		46	
22		47	
23		48	
24		49	
25		50	


DAVID KITTINGER
YOU BETTER RE-PROGRAM THAT! THAT WAS A VERY SEALY THING TO DO!


STEVE OTTO


GLENN
GAME OVER
BP
ROBERT CULLUM


KEN THOMPSON
BELLE
AUSTRALIA


I WIN!
GNU
JOHN STANBECK


ED FELTEN
(WAYCOOL)
PIZZA
FRIES


DANNY KOPEC
"DAVID"


BURT WENDROFF

The International Computer Chess Association

Established at the Second World Computer Chess Championship in Toronto in 1977, this international association has about five hundred members from all over the world. It is published four times a year. The international Computer Chess Association (ICCA) is an international organization that represents the computer chess world, not only to the computer science community (such as ACM, IEEE, and IFIP), but also to the world chess federation (FIDE). The most visible benefit of membership is the quarterly ICCA Journal. Each issue contains roughly 60 pages outlining the latest in computer chess research, news, tournament results, book reviews, conferences, games, etc.: something for researchers, chess program hobbyists, and chess players.

****

ICCA Journal Subscription

Name: ______________________________

Address: ______________________________

____

____

____

____

Cost: $25 US per year.

Mail to:

Dr. Jonathan Schaeffer
Department of Computing Science
University of Alberta
Edmonton, AB
Canada T6G 2H1

Acknowledgements

Organizers

Kelly Aldridge	David Armstrong	Annette Chalifoux
Yian-Leng Chang	Shuang Deng	Darvin Heinemann
Ross Hewitt	David Hiebert	Mic Kohut
Rene Leiva	Jean Leroux	Elizabeth Marsland
Dan Melamed	Stephanie Miller	David Nemes
Adella Pchelnyk	Dan Perl	Stan Petrica
Hal Reitman	John Savard	Pavan Sikka
Steve Sutphen	Sally Trofanenko	

Stan Petrica, *Alberta Government Telephone*, access to a high performance computer system
Walter Neilson, *Alberta Research Council*, access to VAX/VMS computer
Gordon Stipdonk, *Heath-Zenith*, loan of a PC Viewer
John Abramson and Dave Wynn, *IBM*, for loan of two PS/2 high performance PCs
David Lepa, *Sun Micrsosystems*, loan of a SUN-4 workstation
Lloyd Osler, *Unisys*, loan of two PW800 25 MHz high performance PCs

Sincerest apologies to those people who donated their time, but were not acknowledged.

AGT

Alberta Government Telephones

Edmonton Branch of the Canadian Information Processing Society

Host and Sponsors of the 6th World Chess Championship, Edmonton, May 1989

NEW DIRECTIONS in GAME-TREE SEARCH

WORKSHOP

Edmonton, Canada

May 28- 31, 1989

Hosted by the Edmonton Section of the
Canadian Information Processing Society