Review: Computer Go 1984 - 2000

Martin Müller

Department of Computing Science
University of Alberta
Edmonton, Canada
mmueller@cs.ualberta.ca

Abstract. Computer Go is maybe the biggest challenge faced by game programmers. Despite considerable work and much progress in solving specific technical problems, overall playing strength of Go programs lags far behind most other games. This review summarizes the development of computer Go in recent years and points out some areas for future research.
Key words: Computer Go, Go programs

1 Introduction

The introduction briefly describes the rules of the game. Section 2 summarizes the history and current state of computer Go, and Section 3 contains three sample games to illustrate the progress made from 1988 until today. The final Section 4 poses some challenge problems for further research in the field.

1.1 The Game of Go

Go is played between two players Black and White, who alternatingly place a stone of their own color on an empty intersection on a Go board, with Black playing first. The standard board size is 19×19, but smaller sizes such as 9×9 and 13×13 are also used. The goal of the game is to control a larger area than the opponent. Figure 1 shows the opening phase of a typical game.

The capturing rule states that if stones of one color have been completely surrounded by the opponent, so that no adjacent empty point remains, they are removed from the board. Figure 2 shows two white stones with a single adjacent empty point (liberty) at 'a'. If Black plays there, the two white stones are captured and removed from the board. If White plays on the same point first, it will now require Black three moves at 'a', 'b' and 'c' to capture the three stones. Capturing and recapturing stones can potentially lead to the infinite repetition of positions. The *ko rule* forbids such a repetition. A basic ko is shown in Figure 3. After Black captures a single White stone, White cannot immediately take back at 'a' in the diagram on the right side. A large number of rule sets exist, but the variations between them rarely affect the outcome of games. The main differences occur in the evaluation of coexistence or *seki* positions and in the treatment of rare, complex cases of position repetition.

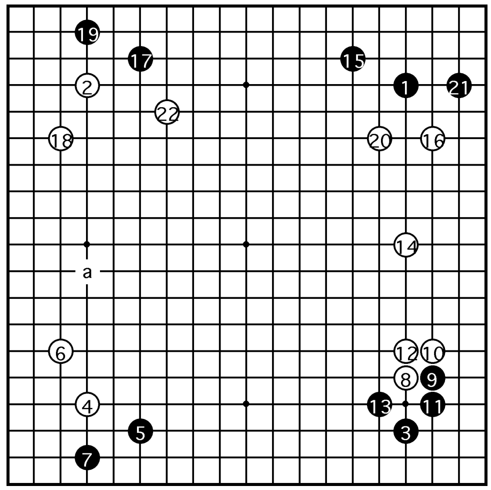


Fig. 1. The game of Go

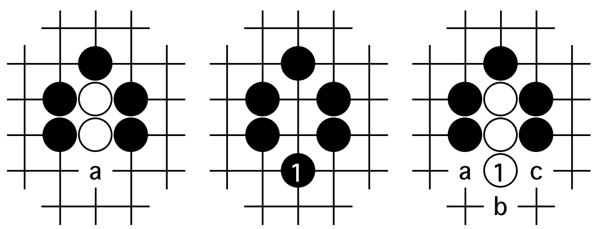


Fig. 2. The capturing rule

Players can pass at any time; consecutive passes end the game. Differences in playing strength can be balanced by a handicap system, which allows Black to place several stones in a row at the start of the game. For detailed information about rules and many other aspects of Go, see [1].

2 The Development and Current State of Computer Go

Of all games of skill, Go is second only to chess in terms of research and programming efforts spent. Yet in playing strength, Go programs lag far behind their counterparts in almost any other game. While Go programs have advanced considerably in the last 10-15 years, they can still be beaten easily by human players of moderate skill. An exact ranking of Go programs is difficult, but Figure 4 shows the rough interval on the human ranking scale in which current programs can be found.

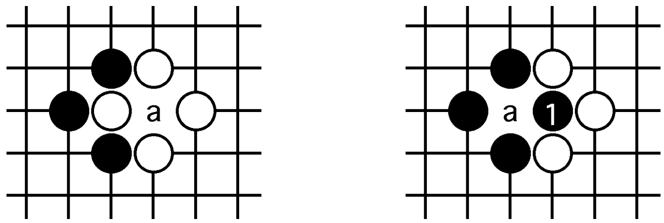


Fig. 3. Ko

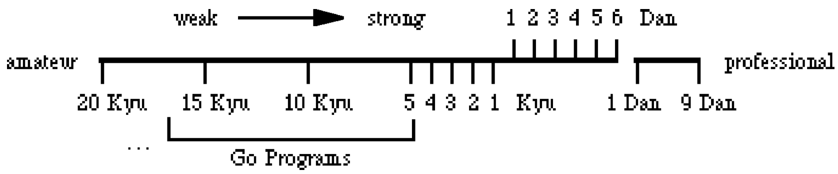


Fig. 4. Go programs on the human ranking scale

2.1 Overview Literature about Computer Go

The development of computer Go has been documented in a number of surveys. Wilcox [7] has written extensively about the early US-based Go programs in the seventies and eighties developed by Zobrist, Ryder, and by Reitman and Wilcox. Important early papers on computer Go are contained in Levy's collection [5]. Kierulf's Ph.D. thesis [4] contains references for most of the programs that participated in the computer Go tournaments of 1985-1989. Erbach [3] gives a good overview of the state of the art in the early nineties. Burmeister and Wiles have published detailed descriptions and comparisons of several modern Go programs [2].

Information about computer Go programs, tournaments and literature is available from many interconnected web sites. Ph.D. theses about computer Go started appearing 30 years ago, and are recently published at a rate of about one per year. See my forthcoming detailed survey [6] and its web companion mentioned at the end of this report for an extensive bibliography and web links.

2.2 Specialized Research in Subtopics of Computer Go

The most tangible progress in the field of computer Go has not been in programs that play a complete game, but rather in programs that address one specific problem. For many subproblems of Go, specialized methods have been developed which achieve a much greater heuristic accuracy than general methods, or can even solve a subproblem precisely. Thomas Wolf's *GoTools* has reached the level of strong amateur players [8] in solving Life and Death puzzles in a small, completely enclosed region. Methods have been developed for proving the safety of stones and territory, fighting semeai and solving difficult endgame puzzles [6].

2.3 Go Research in Related Fields

Go has been used as the topic for research in related fields such as cognitive science and machine learning. In cognitive science studies, human players with different levels of Go skill are tested in order to develop models of human perception and problem-solving behavior. In the field of machine learning, programs have been developed that can pick up basic Go principles, starting from only the rules of the game. A promising approach is the integration of a priori knowledge from expert Go modules into neural networks. There are many books, papers and theses in the field of combinatorial game theory related to computer Go, especially in the areas of endgame and *ko* evaluation. Again, please see [6] or its online companion for further references on these topics.

2.4 Recent Development of Computer Go Programs

While Go programming started in the late sixties, it got a big boost in the mid eighties, with the appearance of affordable PC's on one hand, and of tournament sponsors such as the Ing foundation on the other hand. In early tournaments, Taiwanese programs such as *Dragon* were successful. From 1989-91, Mark Boon's *Goliath* dominated all tournaments, followed by Ken Chen's *Go Intellect* and Chen Zhixing's programs *Handtalk* and *Goemate*. In recent years, *Go4++* by Michael Reiss, David Fotland's *Many Faces of Go* and the controversial North Korean program *KCC Igo* have also won major tournaments. In total there are about 10 top class programs, including *Haruka, Wulu, FunGo, Star of Poland* and *Jimmy*. *Goemate* and *Go4++* seem to be slightly ahead of the rest. A step behind the top 10 is a set of about 30 medium-strength programs. An interesting recent phenomenon is the appearance of good open source programs such as the new *GnuGo*. The total size of the computer Go community can be estimated at about 200 programmers, and is growing steadily.

Several milestones have been reached in the short history of computer Go: In 1991, *Goliath* won a yearly playoff with three strong young human players, taking a handicap of 17 stones. *Handtalk* won the 15 and 13 stone matches in 1995, and the 11 stone match in 1997. Programs such as *Handtalk* and *Go4++* have achieved some success in even games against human players close to *dan* level strength. However, experienced human players can still beat all current programs on much more than 11 stones. *Handtalk* was successively awarded 5, 4 and 3 *kyu* diplomas by the Japanese Go Association *Nihon Kiin* after winning the 1995-97 FOST cups, and *KCC Igo* received a 2 kyu diploma in 1999.

In Japan, in recent years there has been an enormous increase in the number of Go software packages. There are more than two dozen Go-related titles on the market, with prices ranging from $5 - 100$$.

At this time, the future of big, world championship caliber events is uncertain. The FOST cup has been cancelled for lack of funding this year, and the traditional Ing tournament will stop altogether. However, tournaments are likely to continue. Small-scale events continue to be held in Asia, Europe, North America and on the internet. The Computer Olympiad has recently been revived in the context of the Mind Sports Olympiad, and efforts are underway to organize a new large-scale tournament in Japan.

2.5 Computer Go Tournaments

This section lists computer Go tournaments in three separate tables. Table 1 contains major international tournaments, Table 2 North American championships, and Table 3 European championships. For a complete list of computer Go tournaments see *www.usgo.org/computer*.

Event	Location	NP	Winner	2nd place	3rd place
1985 ICGC	Taipei	?	Dragon, 3:0	?	?
1986 ICGC	Taipei	10	(Author: Du), 4:0	Dragon, 3:1	Nemesis, 3:1
1987 ICGC	Taipei	18	Friday, 4:0	Dragon, 3:1	Peanut, 3:1
1988 ICGC	Taipei	16	Codan, 4:0	Dragon, 3:1	Goliath, 3:1
1989 CO	London	10	Explorer, 8:1	Goliath, 7:2	SOP, 6:3
1989 ICGC	Taipei	14	Goliath, 4:0	Nemesis, 3:1	Go Intellect, 3:1
1990 CO	London	3	Go Intellect, 4:0	Explorer, 2:2	Go 4++, 0:4
1990 ICGC	Beijing	10	Goliath, 5:1	Go Intellect, 5:1	SOP, 4:2
1991 CO	Maastricht	?	Goliath	?	?
1991 ICGC	Singapore	15	Goliath, 6:0	Go Intellect, 5:1	Dragon, 4:2
1991 ICOT	Tokyo	8	Goliath, 5:0	Intellect, 4:1	SOP, 3:2
1992 CO	London	?	Go Intellect	?	?
1992 ICGC	Tokyo	10	Go Intellect, 5:1	Handtalk, 4:2	Goliath, 4:2
1993 ICGC	Chengdu	13	Handtalk, 6:0	SOP, 5:1	Go Intellect, 4:2
1994 ICGC	Taipei	9	Go Intellect, 5:1	Many Faces, 5:1	Handtalk, 5:1
1995 FOST	Tokyo	14	Handtalk, 7:0	Go 4++, 6:1	Many Faces, 5:2
1995 ICGC	Seoul	10	Handtalk, 5:0	Go 4++, 4:1	Go Intellect, 3:2
1996 FOST	Tokyo	19	Handtalk, 8:1	Go 4++, 7:2	Many Faces, 7:2
1996 ICGC	Guangzhou	12	Handtalk, 6:0	Go Intellect, 5:1	Stone, 4:2
1997 FOST	Nagoya	38	Handtalk, 9:1	Go 4++, 8:2	Go Intellect, 8:2
1997 ICGC	San Francisco	10	Handtalk, 8:1	Go 4++, 8:1	Go Intellect, 7:2
1998 FOST	Tokyo	38	Silver Igo, 6:0	Goemate, 4:2	Go 4++, 4:2
1998 ICGC	London	17	Many Faces, 6:1	Wulu, 6:1	Go 4++, 5:2
1999 CGF	Tsukuba	28	Go 4++, 8:1	Haruka, 7:2	Goemate, 7:2
1999 FOST	Tokyo	16	KCC Igo, 7:1	Go 4++, 7:1	Many Faces, 6:2
1999 ICGC	Shanghai	16	Go 4++, 6:0	Goemate, 5:1	KCC Igo, 4:2
2000 MSO	London	6	Goemate, 10:0	Go 4++, 8:2	Aya, 5:5

Table 1. Results of International Computer Go Tournaments. NP = number of participants, ICGC = International Computer Go Congress (Ing Cup), FOST = FOST Cup, CO = Computer Olympiad, MSO = Mind Sports Olympiad, ICOT = ICOT tournament, CGF = Computer Go Forum (CGF) Computer Go Tournament, SOP = Star of Poland.

Year	Location	NP	Winner	2nd place	3rd place
1984	USENIX	4	Nemesis, 4:1	Goanna, 3:2	Ogo, 2:3
1985	USENIX	?	Og	?	?
1986	USENIX	?	Og	?	?
1987	USENIX	4	Golem, 5:1	Og, 3:3	Codan, 2:4
1988	USENIX	5	G2*	Goo	Goanna
1988	Berkeley	5	Many Faces, 3:0	Nemesis, 1:2	Infinity Go, 1:2
1989	New Brunswick	10	Go Intellect, 4:0	MicroGo 2, 3:1	Many Faces, 3:1
1990	Denver	7	Goliath*, 6:0	Go Intellect, 5:1	Nemesis, 4:2
1991	Rochester	6	Many Faces, 5:0	Go Intellect, 4:1	Stone, 3:2
1992	Salem	7	Many Faces, 5:1	Go Intellect, 5:1	Nemesis, 4:2
1993	South Hadley	7	Stone, 6:0	Go Intellect, 5:1	Prototype, 4:2
1994	Arlington	4	Go Intellect, 6:0	Many Faces, 4:2	2 programs*, 1:5
1995	Seattle	5	Many Faces, 3:1	Explorer, 3:1	Poka, 3:1
1996	Cleveland	5	Many Faces, 4:0	Explorer, 3:1	Poka, 2:2
1997	Lancaster	2	Many Faces, 2:0	TeamGo, 0:2	-
1998	Santa Fe	5	Many Faces, 4:0	Smart Go, 3:1	Explorer, 2:2
1999	San Francisco	4	Many Faces, 3:0	Gnu Go, 2:1	Smart Go, 1:2
2000	Denver	3	Many Faces, 2:0	Smart Go, 1:1	Poka, 0:2

Table 2. North American Go Tournament. G2 was an early version of Many Faces. In 1990, Goliath was not eligible for the title because it is not an American program. The best American programs in 1990 were Go Intellect, Nemesis and Many Faces. In 1994, third place was shared by Contender and RisciGo.

3 Computer Go 1988, 1994 and 2000: Three Sample Games

The following three games illustrate the performance of the top Go programs in 1988, 1994 and 2000. The first game, shown in Figure 5, was the final of the International Computer Go Congress (Ing Cup), played in Taipei, Taiwan on November 11, 1988. The komi was 8 points, as in all Ing-sponsored competitions. *Codan* by Kazuyoshi Hayashi playing White won by 7 points against Liu Dong-Yue's *Dragon*. Overall, both programs play a very solid, territory-oriented game. *Codan* loses a group on the top edge, but gives it up early enough to avoid disaster, and wins the game by making slightly more efficient moves on average. Much of the game consists of simple, boundary-settling local sequences of play.

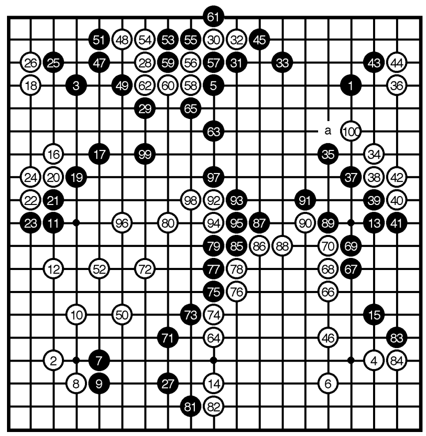
Six years later, Ken Chen's *Go Intellect* won the same event on a tiebreak with five wins to one loss (see Table 1). The game shown in Figure 6 was played in Taipei, Taiwan on November 17, 1994 between the two other programs with five wins, Chen Zhixing's *Handtalk* and David Fotland's *Many Faces of Go*. This game is one of *Handtalk's* rare losses in the time from 1993 to 1997, when it dominated the computer Go scene. In contrast to the 1988 game, this game is characterized by intense fighting, with the focus on the attack and defense of weak groups. The ability to cut and connect weak groups, and the amount of

Year	Location	NP	Winner	2nd place	3rd place
1987	Grenoble	7	SOP*, 6:0	Microgo 2, 5:1	Goliath, 4:2
1988	Hamburg	10	Goliath, 8:1	SOP, 7:2	Progo, 7:2
1989	Nis	6	Goliath, 5:0	Microgo 2, 4:1	SOP, 3:2
1990	Vienna	7	SOP, 6:0	Explorer, 5:1	Nemesis, 4:2
1991	Namur	7	Goliath, ?:?	SOP, ?:?	Progo, ?:?
1992	Canterbury	7	GO 4.3, 6:0	Modgo, 5:1	Progo, 4:2
1993	Prague	6	Progo, 5:0	3 programs*, 3:2	3 programs*, 3:2
1994	Maastricht	7	SOP, 6:0	Imago, 5:1	?
1995	Tuchola	6	SOP, 5:0	TurboGo, 3:2	Argus, 3:2
1997	Marseille	6	SOP, 5:0	GoAhead, 4:1	TurboGo, 3:2
1999	Podbanske	4	TurboGo, 3:0	The Turtle, 2:1	Alpha, 1:2
2000	Strausberg	5	GoAhead, 4:0	TS-Go, 3:1	TurboGo, 2:2

Table 3. European Computer Go Championship. SOP = Star of Poland. In 1993, there was a three-way tie for second to fourth place between Modgo, TurboGo and Gogelaar.

knowledge about Life and Death, play the dominant roles. In this game, *Handtalk* makes two decisive group-related mistakes: move 99 loses the black group at the bottom. It should be played below 96, after which Black could either capture the stones 94 and 96 by playing to the left of 96, or capture the other cutting stone 98 by a move at 100. The second mistake is 141 (41 in the diagram on the right side), which lets White lead out the previously dead group in the center. After these two reversals, White has a big lead. Finally, White blunders at 158, which should be at 160 immediately, but Black misses the chance to turn the corner into *ko* and lets White repair the damage one move later.

As an example of the current state of the art, Figure 7 shows the deciding game of the computer Go tournament at the Mind Sports Olympiad, played in London on August 22, 2000. Playing Black and giving a *komi* of 6.5 points, *Go4++* by Michael Reiss lost by just half a point to *Goemate*, developed by Chen Zhixing as the successor program of *Handtalk*. This game develops in a tight territorial fashion typical of most current top programs, with little fighting going on. Up to 26, standard opening sequences are played out to occupy all corners and sides. The moves from 27 to 33 are also a standard sequence. After that, both programs continue to surround territory, with Black simply giving up stones such as 25 and 35/47 rather than risking a big fight by running out. While the play of both programs in this game is rather simple and safe, their overall performance is very respectable. There may still a large number of less-than-optimal moves, but there are few really big mistakes. Both programs demonstrate an understanding of many aspects of Go. For example, they can build safe territory as well as large frameworks, and can react early to reduce an opponent's sphere of influence. Programs are careful to avoid getting weak groups, and play a reasonable endgame.



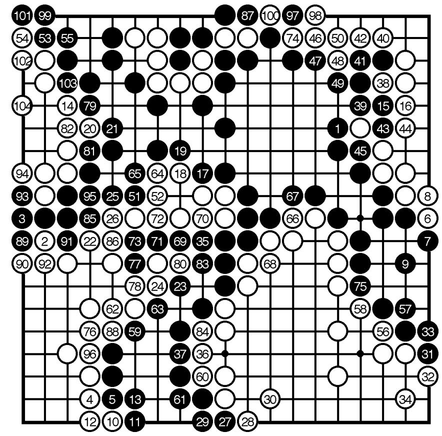


Fig. 5. ICGC 1988: Dragon (B) - Codan (W)

Comparing these three games, computer Go seems to have come full circle. Early programs knew little more than simple rules and patterns to surround territory and capture stones. The next generation, lead by *Goliath*, *Go Intellect* and *Handtalk*, dominated their opponents through a better knowledge of attack and defense. A typical game from this period was decided by a large margin, with the stronger program saving more of its own groups and killing more opponent groups. *Go4++* lead the revolution leading to the current state, by demonstrating that a program which is not so strong in fighting but very efficient in taking territory can win a large percentage of games, even when giving up a few small groups along the way. In recent years, these two extreme approaches have converged to the point where it is hard to distinguish between playing styles. Current programs are stronger than fifteen years ago in judging group strength, life and death, and tactics, but most prefer to play a peaceful game where these strengths are not so apparent. Their style of play hides much of the inherent complexity of Go.

4 Challenges for Computer Go Research

Develop a search-bound Go program In contrast to most other games, in Go there has not yet been a clear demonstration of correlation between deeper search and playing strength. Develop such a Go program that can automatically take advantage of greater processing power.

Comprehensive local analysis Develop a search architecture that can integrate all aspects of local fighting and evaluation.

Threats and forcing moves Develop a system that can systematically detect threats and use them for double threats, ko threats, or for forcing moves, while avoiding bad forcing moves, which have unexpected side effects.

Test suite Develop a comprehensive public domain computer Go test suite.

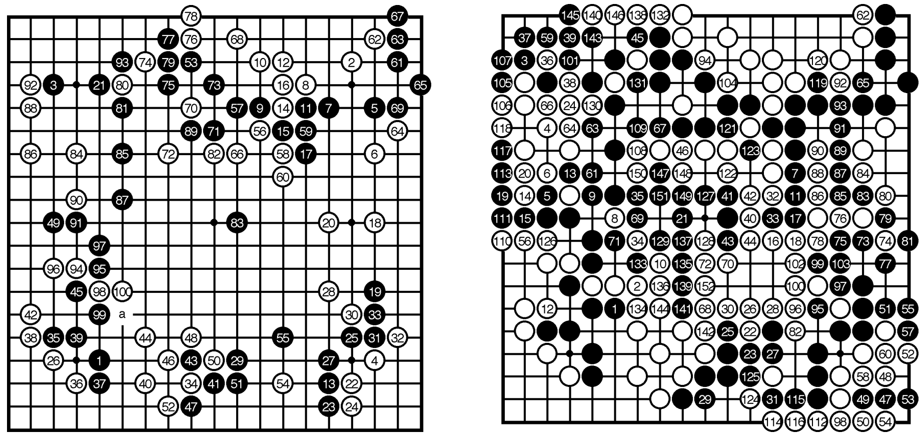


Fig. 6. ICGC 1994: Handtalk (B) - Many Faces of Go (W)

Computer Go source code library Provide a highly (re-)usable library of common functions.

Sure-win program for high handicaps Build a program that can demonstrably win all games on a high handicap, then successively reduce the handicap.

Integrate exact modules in heuristic program Solve the problems of interfacing specialized modules, which handle one aspect of the game very well, with a general Go program.

Solve Go on small boards Human players have analyzed Go on many small rectangular boards, but there are few exact proofs. Solve Go on small board sizes such as 5×5 or 7×7.

Acknowledgement and Web Reference

Parts of this review are based on a forthcoming detailed survey paper *Computer Go* [6], to appear in a special issue of Artificial Intelligence Journal on the state of the art in computer game-playing. For further information also see the web page http://web.cs.ualberta.ca/~mmueller/cgo/survey.

References

1. R. Bozulich. *The Go Players Almanac.* Ishi Press, Tokyo, 1992.
2. J. Burmeister and J. Wiles. AI techniques used in computer Go. In *Fourth Conference of the Australasian Cognitive Science Society*, Newcastle, 1997.
3. D. W. Erbach. Computers and Go. In Richard Bozulich, editor, *The Go Player's Almanac.* The Ishi Press, 1992. Chapter 11.
4. A. Kierulf. *Smart Game Board: a Workbench for Game-Playing Programs, with Go and Othello as Case Studies.* PhD thesis, ETH Zürich, 1990.

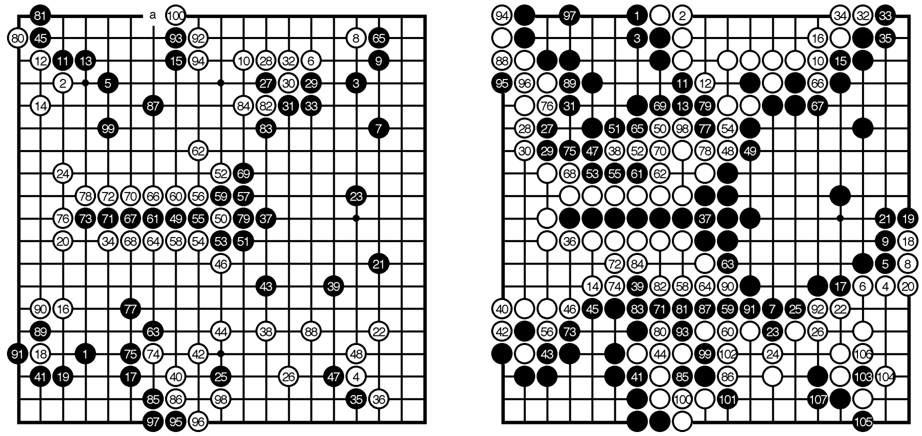


Fig. 7. Mind Sports Olympiad 2000: Go4++ (B) - Goemate (W)

5. D. Levy, editor. *Computer Games I+II*. Springer Verlag, 1988.
6. M. Müller. Computer Go, 2000. Survey paper, special issue on games of Artificial Intelligence Journal, to appear.
7. B. Wilcox. Computer Go. In D.N.L. Levy, editor, *Computer Games*, volume 2, pages 94–135. Springer-Verlag, 1988.
8. T. Wolf. Forward pruning and other heuristic search techniques in tsume go. *Information Sciences*, 122:59–76, 2000.