Design and Implementation of a Heuristic beginning Game System for Computer Go

Shi-Jim Yen, Wen-Jyh Chen, Shun-Chin Hsu

Department of Computer Science and Information Engineering of National Dong Hwa University, Taiwan

E-mail: sjyen@mail.ndhu.edu.tw

Abstract

There are roughly three stages in a Go game: the beginning game, the middle game, and the end game. This paper describes a computer Go beginning game system which includes occupying corners, joseki, extending edges, and dealing with Moyo. This beginning game system has been used in a computer Go program named Jimmy 4.0. Having been tested by professional Go players, this system is estimated at about 6 kyu in terms of its beginning game performance.

Keywords: computer Go, beginning game.

1. Generation and Quantifying of the Beginning Stage Moves

According to the conventions of the Go game, the beginning stage is divided into occupying corners, joseki, extending edges, and dealing with Moyo. In the following article, we will explain how to design Go programs based on the patterns and strategies of the beginning game. In addition to the research on how to determine good candidate moves for every stage of the game, we quantify the value of every move based on the experiences of Go experts. Next, the values of each move are integrated, in order to allow the program to accurately choose moves we need based on the established values.

This beginning system has been previously used in a computer Go program named Jimmy 4.0. After consultation with high dan Go players, it is estimated that this system is about 6 kyu in the beginning game stage.

2. Basic Principles of the Quantifying of the Go Moves

Finding legal candidate moves is very easy. However, in order to evaluate these moves effectively, the number of moves must be minimized. Because candidate moves are generated according to Go principles and shapes, the numbers of moves can be reduced to between fifteen and twenty-five.

The main idea of quantifying the moves is to apply end game theories to the beginning game. Because of fewer moves and easier positional judgment system in the end game stage, the best moves in a computer Go program can be judged by applying combinatorial game theory[1]. In fact, we discovered that the thinking styles of the combinatorial game theories that mathematicians explore is very similar to the end game strategies that human Go players utilize. Many Go experts such as Otake Hideo and Yoda Norimoto[2] have also used this thinking approach. And they have had success with this approach.

3. Generation and Quantifying of the moves of occupying corners and joseki

Because of the limited variations in the initial beginning moves of a Go game, we adopted pattern matching to generate the candidate corner occupying moves in this stage. The shapes of occupied corners generally consist of star point, 3-3 point, small point, high point and point-detached. We adopted pattern matching to quantify these moves. We stored the quantified estimated values of the corner moves for the shapes of occupied corners. When system generates a candidate move, it also knows the value of each move at the same time. The quantified and estimated values of the shapes of occupied corners are about 20 points[2].

Joseki is a series of standard moves in corners or sometimes on edges. We built a tree structure using a sequence of joseki moves. We can predict the location of the next move by searching the child nodes of a pointed node. The location of these child nodes becomes our joseki candidate moves. We input the move sequences of joseki and give values to them at the same time by referring to the information at hand. Therefore, we can obtain the relative quantified values by searching the nodes of joseki.

4. Generation and Quantifying of the moves of extending edges

We have two methods of generating the moves of extending edges. We used pattern matching to generate the candidate moves for the empty points near existing stones on the Go board. For example, The basic concepts for extending edges are "two in rows, extend three-point move", " three in rows, extend four-point move" and so on[4]. We generate candidate moves based on the "split" method of the concepts for beginning stage Go, when only two corners have Go stones on either side. The basic concepts of the "split" method are as follows. When only two corners have Go stones in either side, the empty point that is of equal distance from each stone in the corner is a good point[7]. Our system determines split-point on the edge in this way, then determine the play of a third-line or fourth-line move based on the location and color of the Go stones in the two corners.

In evaluation extending-edge moves, generally speaking, usually many stones have been placed on the board and discrimination between stronger stone groups and weaker groups is necessary in considering extending-edge moves. When the values of every extending-edge move is considered, the values of occupied territory and the influential values of both groups on each side are important factors[2]. We describe how to compute the two values as follows.

On evaluating the different values of occupied territory, we search candidate moves based on the above end-game-theory and use an accurate position judgment system to compute them. Figure 1 is an example for quantifying extending-edge moves. When Black plays 'A' move, Black can occupy 4-point lands and gain 2-point influential values. However, When White plays 'B' move, White can occupy 4-point lands and gain 2-point influence values. Therefore, on evaluating the different values of occupying lands here, the values of the Black's 'A' move are 12 points (4+2+4+2).

The unsafe stone groups of both sides are located based on the determination of both sides influential values by the position judgment system. Because the value of occupied territory will be increased through extending-edge moves, extending-edge moves have a definite effect on the support or attack on unsafe stone groups on both sides. When a candidate move is near the unsafe stone groups of both sides, we add the degree of influential value of the unsafe stone group to it's quantifying value. Take figure 1 for example. If stone group 'C' is captured by white stones, the white stones will occupy region 'X'(8 points) and stone 'C'(2 points). Therefore, the value of stone 'C' is about 10 points. The value of stone group 'D' is also 10 points. In this example, when stone group 'C' or stone group 'D' is attacked by the opponent, the opponent must play two sequential moves near that stone group to kill it. The value of 'A' move is about 10 points((10+10)/2) in consideration of the influential values of both stone groups. The quantifying values of the extending edges moves are 22 points, including the 12 points generated by the different values of the occupied territory of both sides.

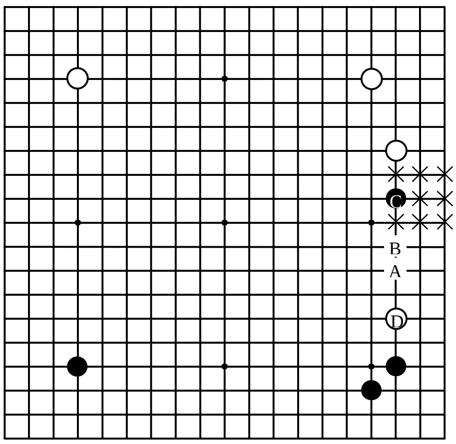

Fig. 1. An example of evaluation for occupy-edge move

5. Generation and Quantifying of the moves of Moyo

We use pattern matching to generate the candidate moves of Moyo. The main Moyo moves are one-point jump, two-point jump, small-horse fly , big-horse fly and so on. The generation of Moyo moves usually occurs in the center of the board. The most important Moyo moves are between Black groups and White groups(Go terms named "Super good points"). We can find candidate moves of Moyo based on the above Go principles.

We consider only the differences of occupied territory of both sides in evaluating Moyo moves. Here, the method for computing differences of occupied territory is the same as it in extending edges moves. We considered the differences of occupied territory of both sides and the differences of influence to compute the quantifying values.

6. Results and Performance

The beginning game system has been used in the computer Go program—Jimmy 4.0. In order to obtain the efficiency of the beginning game system accurately, we adopted tests used in Go books to test the rank of the beginning game system. The method for answering the questions was similar to the unique-option questions in tests. After finishing all of the questions, we computed the sum of the scores for every question. Finally the relative rank was obtained by referring to score-rank table. We choose one hundred questions for the beginning stage from some Go books[3][4][5][6]. The scores of this system determined that the beginning game has a rank of 4 kyu.

Besides analyzing the above tests, we must consider the degree of satisfaction this system provides to whole programs. In other words, we require more accurate quantifying values for the programs. A determination must be made if the quantifying values are accurate enough to improve the beginning game systems. Our method of estimation was to compare the opinion of high dan players with the beginning game system on identical board situations. First, the high dan players computed the values of a certain move. Then their computations were compared with the quantifying values of the moves generated by the beginning game system. Finally the error rate was computed. The results of the average error rate of every sub system are shown in table 1.

Table 1 The average error rate

	Average error rate
Occupying corners	8%
Joseki	15%
Extending edge	23%
Moyo	29%

From the above table, we know that the average error rate for occupying corners was lower than the rate of joseki. The main causes for this are that move generation and move quantifying are directly acquired from the shape database. The shape database was built by high dan players. If the shape database were more complete, the results would be more effective and the system rank would be nearer that of high dan players. However, the average error rate for extending edges and Moyo is relatively higher. The main causes are that move generation and move quantifying for both must refer to the information from stone group and influential estimating values provided by the position judgment system. In other words, the bottleneck is the position judgment system. The rank of the position judgment of our system is 6 kyu. From the above discussion and the results in table 1, the rank of this system is estimated at about 6 kyu in terms of beginning game performance.

References

[1] Martin Müller, “Computer Go as a Sum of Local Games: An Application of Combinatorial Game Theory,” Ph.D. Dissertation, Swiss Federal Institute of Technology Zurich, 1995.

[2] 大竹英雄九段，**大竹英雄圍棋系列４－圍棋大局觀**，6至17頁，理藝出版社，台灣新竹，1992。

[3] 大竹英雄九段，**大竹英雄圍棋系列４－圍棋大局觀**，6至17頁，理藝出版社，台灣新竹，1992。

[4] 張續儒，**圍棋段級測驗第一集**，**圍棋段級測驗第二集**，**圍棋段級測驗第三集**，王家出版社，台灣台南，1984。

[5] 林海峰九段，**初段棋力的測試**，大孚書局，台灣台南，1984。

[6] 陳憲輝，**圍棋實力測驗系列**，理藝出版社，台灣新竹，1992。

[7] 藤澤秀行，**序盤下法**，世界文物出版社，台灣台北，1981。