

Effect of Potential Model Pruning on Different-Sized Boards in Monte-Carlo GO

Oshima Makoto[†], Koji Yamada^{††}, and Satoshi Endo^{††}

[†]Graduate School of Engineering and Science, University of the Ryukyus, Okinawa 903-0213, Japan

^{††}Faculty of Engineering, University of the Ryukyus, Okinawa 903-0213, Japan

Summary

Monte-Carlo GO is a computer GO program that is sufficiently competent without the knowledge expressions of IGO. Although it is computationally intensive, the computational complexity can be reduced by properly pruning the IGO game tree. In this study, we achieved this by using a potential model based on the knowledge expressions of IGO. The potential model treats GO stones as potentials. A specific potential distribution on the GO board results from a unique arrangement of stones on the board. Pruning with the potential model categorizes legal moves into effective and ineffective moves in accordance with the potential threshold. In this study, certain pruning strategies based on potentials and potential gradients were experimentally evaluated. In particular, for different-sized boards, the effects of pruning strategies were evaluated in terms of their robustness. We successfully demonstrated pruning with a potential model to reduce the computational complexity of the game of GO as well as the robustness of this effect across different-sized boards.

Key words:

Reducing Computational Complexity, Heuristic, Potential Filter, Geometric Information Systems, Potential Gradients

1. Introduction

In this study, we tackled the reduction of computational complexity by pruning the IGO game tree with a potential model based on the knowledge expression of IGO. Monte-Carlo GO [1], which is sufficiently competent without the knowledge expressions of IGO, was used as the computer GO program for this experiment. Monte-Carlo GO employs a randomized and computationally intensive algorithm. However, this computational complexity can be reduced by properly pruning the IGO game tree. Monte-Carlo GO shows no deviation in the sequence of moves for IGO. Therefore, the effects of the heuristics generated by a potential model are demonstrated correctly.

This study is the additional research of potential model pruning in Monte-Carlo Go [2]. In the previous experiment, effects of four kinds of potential model on Monte-Carlo Go were demonstrated on the 9×9 board. In this experiment, nine kinds of potential model were demonstrated on the 9×9 and 13×13 boards.

2. Proposed Method

The method proposed in this study consists of Monte-Carlo GO and a potential model.

2.1 Monte-Carlo GO

Monte-Carlo GO evaluates legal moves in each phase to choose the next move by a simulation based on a Monte-Carlo search process consisting of many moves. This simulation, called “Play Out,” involves both sides constantly choosing the next move alternately and randomly from the current phase until the end of the game. Play Out calculates an estimate $\overline{X_i}$ for each legal move i by using Eq. (1). Here, S_i is the number of times of Play Out and X_i is the total number of considerations. In Play Out, the consideration is +1 or 0 if an offensive move wins or loses, respectively. As a result, the move with the best estimate is selected as the next move.

$$\overline{X_i} = X_i / s_i \quad (1)$$

2.2 Potential Model

Stones influence the possibility that surrounding intersections become their territory. The potential model proposed here quantifies these influences by assuming GO stones as potentials, as shown in previous studies [3, 4].

2.2.1 Definition of Potential

The potential is defined in Eqs. (2–4) and Table 1. A calculation example is shown in Fig. 1. The sign of Eq. (3) is switched depending on the setting of the proposed method. The potential distribution on the GO board is calculated by these equations. If necessary, the potential gradient is subsequently calculated according to the gradient method by using geographical information systems [5] with the potential distribution. A schematic diagram of this process is shown in Fig. 2.

$$r = \sqrt{(X - x_i)^2 + (Y - y_i)^2} \quad (2)$$

$$P_i(X, Y) = \pm 1/m^r \quad (3)$$

$$P_{all} = \sum_{k=1}^n P_k(X, Y) \quad (4)$$

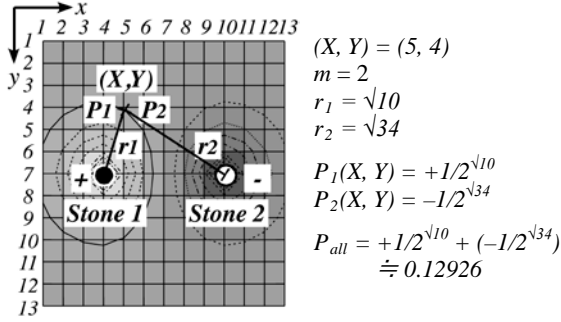


Fig. 1. Example of Potential Calculation

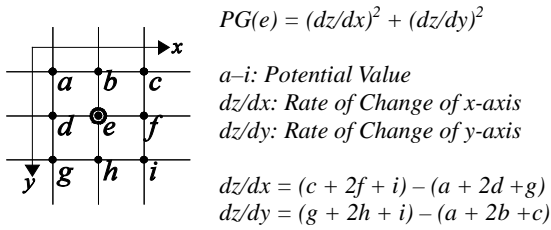


Fig. 2. Schematic of Potential Gradient

Table 1: Mathematical Expressions

r	:	Euclidean distance
m	:	Attenuation rate of potential, $m > 1$.
x_i, y_i	:	Intersection of $stone_i$
$P_k(X, Y)$:	Potential difference between intersection (X, Y) and $stone_k$
n	:	Total number of stones on the GO board
$P_{all}(X, Y)$:	Total potential difference between intersection (X, Y) and $stone_{1-n}$
$PG(X, Y)$:	Potential gradient at an intersection (X, Y)

2.2.2 Pruning by using Potential Model

Potential Filters (PFs):

Potential Filters (PFs) were used as the pruning instruments. In each phase of choosing the next move, these filters pruned legal moves according to the following procedures:

- Calculate the potential distribution resulting from the arrangement of GO stones on the GO board.
- Rank legal moves by each magnitude of potential (or potential gradient).

- Categorize ranked legal moves into effective and ineffective moves in accordance with thresholds for the ranking. (Each PF has a unique threshold level.)
- Eliminate ineffective moves from candidates for the next move. (Run Monte-Carlo search only on effective moves.)

In accordance with the number of eliminated legal moves, the computational load of the Monte-Carlo search is reduced; that is, PFs reduce the range of search spaces on the GO board.

Potential Filter Configurations:

Table 2 lists the configurations of the five filters (the Random Filter and PFs 1–4). These configurations include the ranking, attenuation rate of the potential m , polar characteristic of black and white stones, and threshold conditions. Each PF ranked legal moves in descending order of potential values (except for the Random Filter) and categorized them in accordance with each threshold condition for the ranking. PFs 1–3 are the same as PFs 1–3 in the previous experiment [2]. PF 4 is a new filter and both black and white stones have the same polarity.

Table 3 lists the configurations of five other filters (PFs 5–9). Each PF ranked legal moves in descending order of potential gradient values and categorized them in accordance with each threshold condition for the ranking. These fundamental configurations are the same as PF 4 in the previous experiment [2]. In PFs 5–9, m critically involved their filtering functions. According to the magnitude of m , potential gradient values of intersections surrounding each stone became higher. In contrast, the lower m became, there were higher potential gradients of intersections between black stones and white stones. For example in Table 3, PF 5, m is 4 and the intersection marked with an x, the midpoint between a black stone and a white stone, is ranked 41st in order of magnitude of potential gradient. In the case of PF 6, the intersection marked with an x is ranked 23rd. In the case of PF 7, the intersection marked with an x is ranked 9th. In the case of PF 8, the intersection marked with an x is ranked 3rd. And in the case of PF 9, the intersection marked with an x is ranked 1st.

All filters mutually reduced by half the number of legal moves. Thus, all filters reduced by half the computational load in each phase for choosing the next move.

On and Off Switch of Potential Filter:

Each PF had a point at which its state was switched on or off. This switching point took a number from among the number of all intersections on the GO board. Specifically, a switching point could be selected from numbers 2 to 169 when the board size was 13×13 (=169), or from 2 to 81 when the board size was 9×9 (=81).

During the course of a game, the PFs were on when the number of legal moves remaining on the *GO* board was above a switching point and off when it was below the switching point. The borders where effective PFs became ineffective were measured by changing the switching point one step at a time. The borders were the points where winning percentages exceeded the average winning percentage between two normal Monte-Carlo *GO* programs (51% with a board size of 13×13 or 57% with a board size of 9×9).

The performance of the Monte-Carlo search is higher when the game tree is small; performance deteriorates as the game tree becomes larger. Thus, pruning is effective in the opening game. However, pruning gradually becomes ineffective thereafter as legal moves on the *GO* board decrease.

3. Competence of Monte-Carlo *GO* with Potential Filters

Monte-Carlo *GO* with PFs was adopted for the initiative move, whereas normal Monte-Carlo *GO* was adopted for the passive move. The number of times of Play Out at each intersection was set to 100. In a match-up between two normal Monte-Carlo *GO* programs, the winning percentage of the initiative move was 51% when the board size was 13×13 , or 57% when board size was 9×9 . (The winning percentage of the initiative move exceeded 50% because this move was advantageous.) Therefore, 51% or 57% was considered the average level of normal competence.

4. Results and Observation

The winning percentages of Monte-Carlo *GO* with PFs are shown in Figs. 3 and 4 along the left-hand axis (upper graphs, board size of 9×9 ; lower graphs, board size of 13×13). The level of competence varied with the filter and switching point. The normal winning percentage of 57% or 51% and the calculated results of the Random Filter were important for comparing and evaluating the effects and tendencies of the PFs. The number of total Play Out times required for one game is shown in Figs. 3 and 4 for both board sizes along the right-hand scale. The number of total Play Out times varied with the filter and switching point.

4.1 Effects of Potential Filters

Random Filter prunes legal moves at random. Therefore, the winning percentage of the Random Filter decreased gradually with a reduction in the number of legal moves without exceeding the normal winning percentage.

PF 1 became the bias around which black stones gathered. These black stones effectively strengthened initiative territory. PF 2 became the bias where black stones were attracted around white stones. Black stones effectively suppressed white stones. PF 3 became the bias where black stones were scattered on the *GO* board. These black stones were removed easily by white stones. PF 4 became the bias where stones were attracted around black and white stones. PFs 5–9 became the bias where black stones were attracted around black and white stones, and areas between black and white stones were closed. The lower the value of m , the stronger was the bias and effect of pruning.

The characteristics of each PF were unique. However, they all showed the ability to properly prune ineffective moves that the Monte-Carlo search could not when the winning percentage exceeded the average (57% or 51%). Thereafter, the competence of each PF decreased gradually as the number of legal moves decreased and the precision of the Monte-Carlo search increased. In fact, pruning by each PF reduced the precision of the Monte-Carlo search.

4.2 Robustness of Potential Filter Effects

Concerning both the upper and the lower graphs in Figs. 3 and 4, if the x-axes are scaled to the same width, each of the winning percentage curves of PFs 1–9 in both graphs is similar and shares much in common with the others: the relative location, the proportion of the border where effective PFs became ineffective, the number of all intersections on the *GO* board, and the reduction rate of total Play Out numbers required for one game. This indicates that the PF effects are robust to the size of the *GO* board; they depend only on the ratio of the number of legal moves to the number of all intersections on the *GO* board.

5. Summary

In this study, we reduced computational complexity of Monte-Carlo *GO* by pruning the *IGO* game tree with a potential model based on the knowledge expression of *IGO*. In our experiments, the effects of 9 kinds of pruning strategies (PFs) were evaluated on different-sized boards. Each PF has a specific effect on *IGO*, which was maintained on the 9×9 and 13×13 boards.

We successfully demonstrated pruning by using the potential model to reduce the computational complexity of *GO*, as well as robustness of the PF effects to the size of the *GO* board. However, our experiments were limited as the Play Out number was set to 100 and the board size was set to 9×9 or 13×13 . For future research, we intend to expand the proposed strategy to address more complex games with larger Play Out numbers and *GO* board sizes.

References

- [1] B. Brüggmann B (1993), Monte Carlo Go. Physics Department, Syracuse University.
- [2] M. Oshima, K. Yamada and S. Endo, Probability of Potential Model Pruning in Monte-Carlo Go, Procedia Computer Science, Vol. 6, 2011, pp. 237-242.
- [3] A.L. Zobrist (1969), A model of visual organization for the game of GO. AFIPS '69 (Spring) Proceedings of the May 14-16, 1969, Spring Joint Computer Conference, pp. 103-112.
- [4] K. Nakamura and K. Syuhei, IPSJ Journal (The Information Processing Society of Japan) 43, no. 10 (10 2002) 3021-3029.
- [5] P.A. Burrough and R.A. McDonnell, Principles of Geographical Information Systems, Oxford University Press, 1998, p. 190.

Table 2. Types of Potential Filters (Random Filter and PFs 1-4)

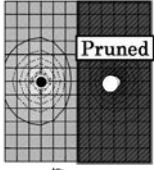
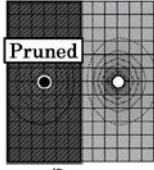
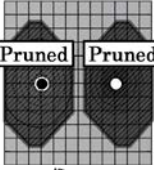
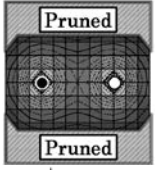
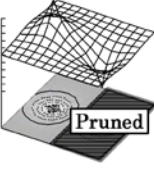
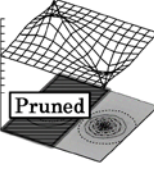
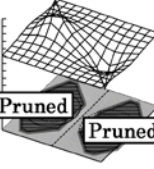

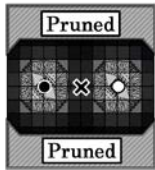
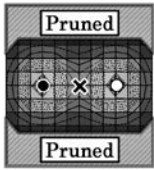
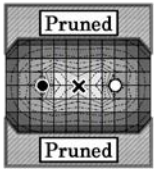
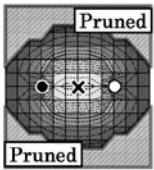
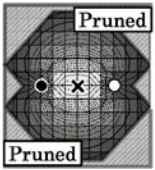
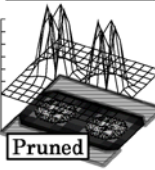

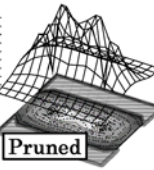


Method	Random Filter	PF 1	PF 2	PF 3	PF 4
Ranking	-	Potential	Potential	Potential	Potential
m	-	2	2	2	2
Black/White	-	+/-	+/-	+/-	+/+
Filtering	Random	Low 50%	Top 50%	Above 25% and below 75%	Low 50%
Overhead	-				
Landscape	-				

Table 3. Types of Potential Filters (PFs 5-9)

Method	PF 5	PF 6	PF 7	PF 8	PF 9
Ranking	Potential Gradient	Potential Gradient	Potential Gradient	Potential Gradient	Potential Gradient
m	4	2	1.5	1.25	1.15
Black/White	+/-	+/-	+/-	+/-	+/-
Filtering	Low 50%	Low 50%	Low 50%	Low 50%	Low 50%
Overhead					
Landscape					

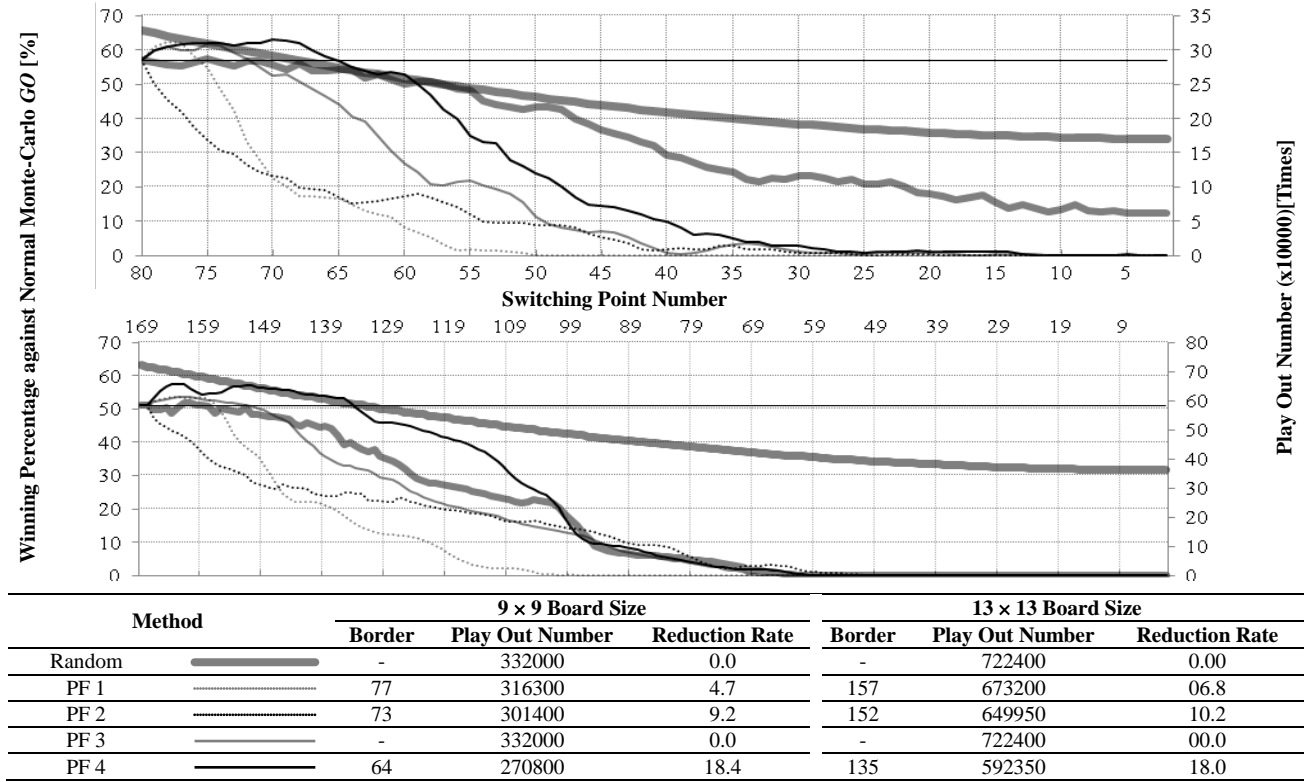


Fig. 3. Winning Percentages of Monte-Carlo GO with Potential Filters

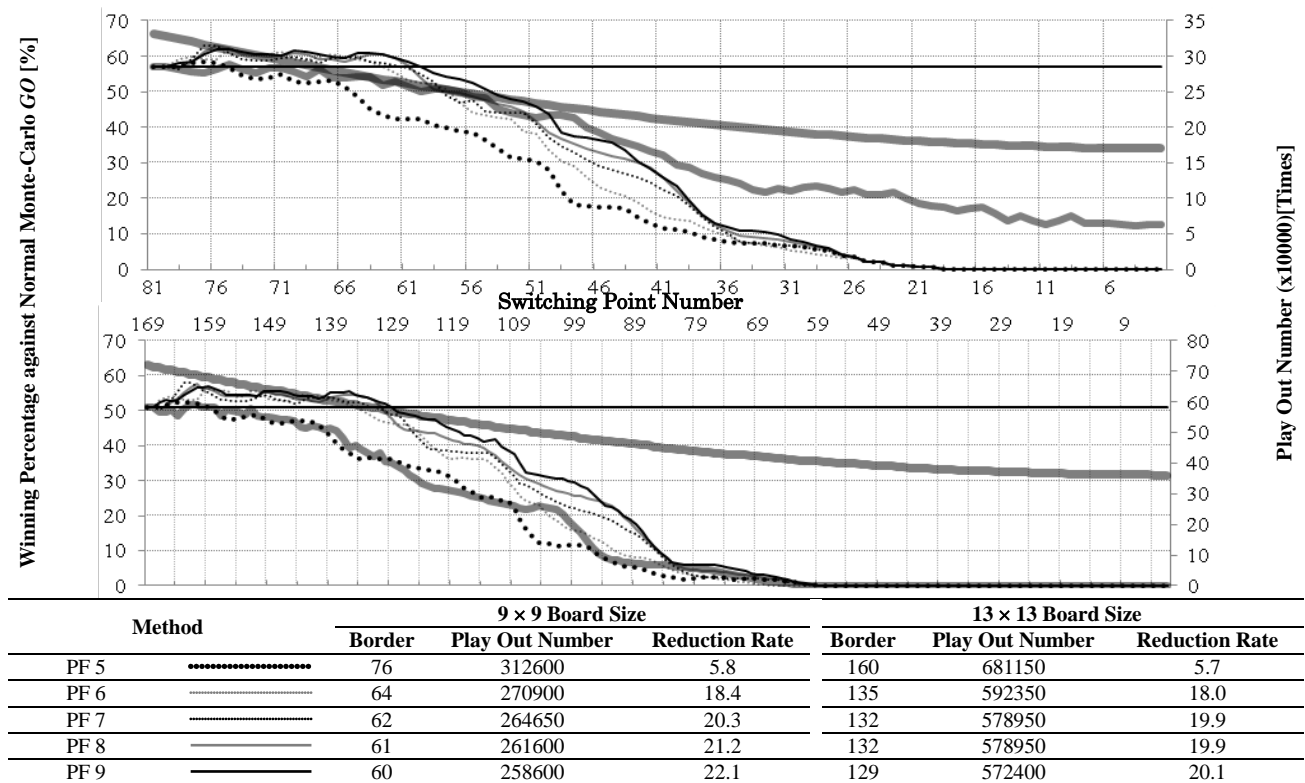


Fig. 4. Winning Percentages of Monte-Carlo GO with Potential Filters



of JSAI and IPSJ.

Makoto Oshima received the B.E. and M.E. degrees, from University of the Ryukyus. in 2003 and 2005, respectively. He has been a doctoral student at University of the Ryukyus since 2009. His research interest includes artificial intelligence, game programming and complex systems engineering. He is a member



intelligence and robotics. He is a member of JSAI, RSJ, IPSJ, JSME and SICE in Japan.

Koji Yamada received the B.E., M. E. and Dr. Eng. degrees from Hokkaido Univ. in 1990, 1992, and 1995, respectively. After working as an assistant professor (from 1995), he has been an associate professor at the Dept. of Info. Eng., Univ. of the Ryukyus since 1999. His research interest includes emergent intelligence, swarm



He is a member of JSAI, IPSJ, SOFT and SICE in Japan.

Satoshi Endo received the M.E. and D.E. degree from Hokkaido Univ. in 1990 and 1995, respectively. He had been a research assistant at Hokkaido Univ. from 1990 to 1995. He has been a professor at Univ. of the Ryukyus since 2005. His research interest includes intelligent informatics and sensitivity informatics/soft computing.